

## BAB II

### TINJAUAN PUSTAKA

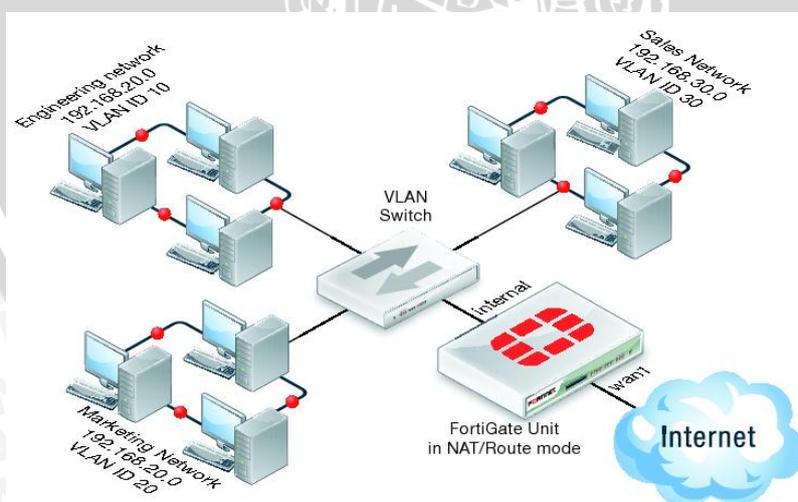
#### 2.1 Virtual Local Area Network (VLAN)

VLAN adalah suatu *broadcast domain* yang dibuat pada sebuah *managable switch* dan memakai sebuah *subset* dari port fisik pada *switch*. *Broadcast domain* adalah kumpulan alat jaringan dimana sebuah *broadcast frame* yang dikirim oleh satu alat diterima oleh seluruh alat yang terdapat dalam kumpulan alat jaringan tersebut, suatu LAN fisik dapat disamakan dengan suatu *broadcast domain* (Odom 2004).

Pada dasarnya, VLAN tidak berbeda jauh dengan LAN. Perbedaan besar antara keduanya adalah perbedaan konsep secara logika. Pada LAN, untuk satu peralatan jaringan seperti *switch*, hanya dapat dibuat satu LAN di dalamnya, sedangkan untuk VLAN tidak. Pada VLAN, untuk satu peralatan jaringan seperti *switch*, dapat dibuat lebih dari satu VLAN dimana setiap VLAN tersebut memiliki wilayah teritorial masing-masing.

Pengimplementasian VLAN pada suatu sistem jaringan memiliki beberapa keuntungan antara lain:

1. Penggunaan *bandwidth* yang lebih efisien.
2. Meningkatkan fleksibilitas jaringan.
3. Meningkatkan kinerja jaringan yang ada.
4. Menjamin tingkat keamanan jaringan yang lebih baik (Rajaravivarma 1997).



Gambar 2.1 Jaringan VLAN

(Sumber: <http://docs-legacy.fortinet.com>)

### 2.1.1 Klasifikasi *Virtual Local Area Network* (VLAN)

Berdasarkan perbedaan kriteria, VLAN dapat diklasifikasikan pada beberapa kategori. Ada beberapa tipe yang biasa digunakan yaitu:

#### 1. Port based

Pada port-based VLAN, setiap VLAN diberi nama pada satu atau lebih port VLAN yang ada pada *switch*. Pada kondisi awal setiap port VLAN dinamakan dengan (VLAN ID 1). Jika setiap *user* yang ingin berkomunikasi satu dengan yang lain walaupun dengan IP VLAN yang berbeda jalur harus dialihkan oleh *switch*, bahkan jika setiap *user* tersebut secara fisik pada bagian I/O yang sama. Hal ini menunjukkan setiap VLAN harus dikonfigurasi dengan alamat IP.

Untuk membuat VLAN dengan menggunakan dua *switch*, pengguna perlu menetapkan port pada setiap *switch* ke VLAN dan untuk menghubungkan dua *switch* atau lebih (menggunakan *Link Resilient* atau *Link Aggregation*) pada setiap *switch* per VLAN

#### 2. Tagged based

Tagged VLAN adalah proses menandai ke *ethernet frame*. Tag VLAN memuat nomor identifikasi dari VLAN, yang biasa disebut dengan VLAN ID. Setiap VLAN ditetapkan pada 802.1Q VLAN tag. Port *switch* dapat ditambahkan ke VLAN dengan 802.1Q yang telah ditetapkan.

Sebuah VLAN dapat ditandai pada beberapa port dan sebaliknya. *Switch* menentukan jalur dari port yang ditandai atau yang tidak ditandai secara *Real Time*. Salah satu keuntungan *Tag* VLAN adalah port yang digunakan bisa menjadi anggota beberapa VLAN. Hal ini berguna jika menggunakan *server* dari setiap VLAN.

#### 3. IP Subnet Based

*Interface* VLAN adalah komunikasi *interface* secara virtual yang digunakan antara VLAN yang berbeda. Setiap VLAN memiliki satu *interface* VLAN. Packet dari sebuah VLAN dapat ditujukan pada layer jaringan melalui *interface* VLAN yang sesuai. Sebagai setiap bentuk alamat *broadcast* dari VLAN, sebuah VLAN dapat menjadi segmen jaringan IP tersendiri dan *interface* VLAN akan menjadi gerbang yang mengizinkan IP *address* sesuai dengan layer tiga yang dituju. (Zeng Xiyang, 2009: 2)

### 2.2 Video Streaming

*Video streaming* adalah gambar yang bergerak serta memiliki urutan, dikompresi menjadi *file* yang lebih kecil kemudia dikirimkan melalui jaringan internet dan ditampilkan oleh *media player* ketika *video* tersebut telah diterima oleh pengguna. Pengguna

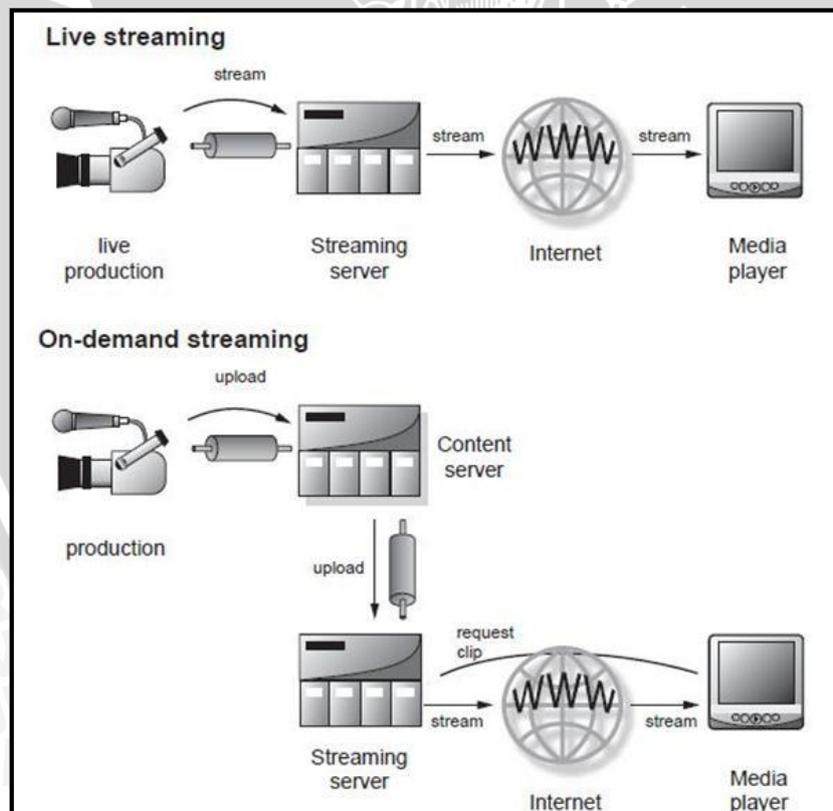
membutuhkan *media player*, yaitu aplikasi khusus yang melakukan proses mulai dari dekompresi, pengiriman data berupa *video* ke tampilan layar *monitor* dan suara ke bagian *speaker*.

Ada dua jenis tipe *video streaming*, yaitu:

1. *Live Streaming*, dimana tayangan yang ditampilkan merupakan siaran langsung.
2. *VOD (Video On Demand)* dimana *video* yang ditampilkan sudah terlebih dahulu direkam (*pre-encoded*) atau disimpan dalam *server*.

Faktor-faktor yang berpengaruh dalam distribusi *video streaming* melalui jaringan antara lain besar *bandwidth* tersedia yang bervariasi (terhadap waktu), *delay* (waktu tunda), *lost packet*, dan juga teknik mendistribusikan *video* tersebut ke beberapa tujuan secara merata dan efisien. (Apostolopoulos, 2002: 1)

Kemudian berdasarkan Austerberry dalam bukunya *The Technology of Audio and Video Streaming*, proses *live streaming* dan *on-demand streaming* dapat dibedakan dan ditunjukkan oleh Gambar 2.2,



**Gambar 2.2** *Live Streaming* dan *On-demand Streaming*  
(Sumber: Austerberry, 2004: 144)

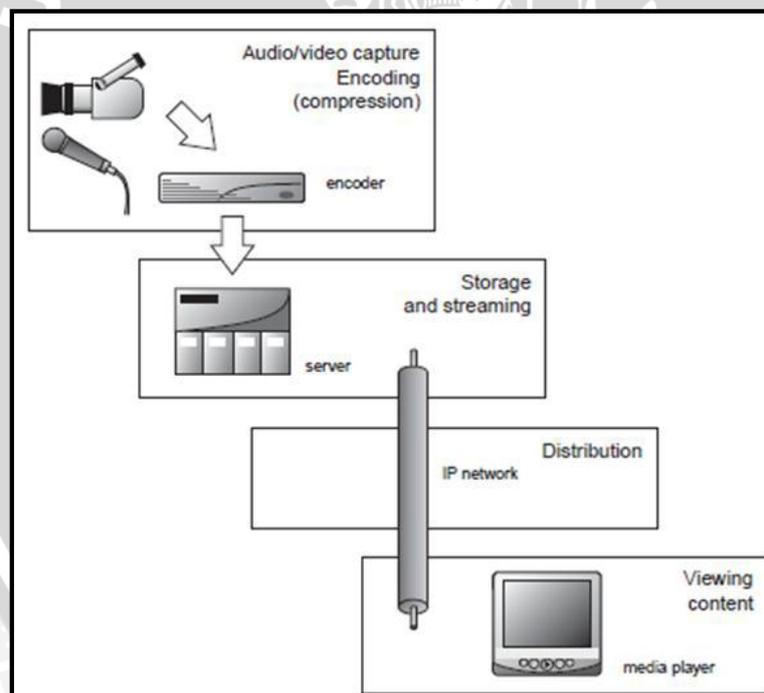
Perbedaan mendasar antara *streaming* dengan televisi pada umumnya adalah adanya interaksi. Pengguna dapat berinteraksi dengan konten *file*. Pengguna memiliki kontrol-kontrol VCR (PLAY, STOP, FF, REW) dan akses acak lainnya. Untuk mendukung fungsi ini tidak

hanya dibutuhkan sebuah *regular web server* tapi juga sebuah *streaming server*. (Austerberry, 2004: 144)

### 2.3 Video On Demand (VoD)

*Streaming* dapat dilakukan dengan dua cara. Media dapat menjadi saluran televisi pada umumnya. Ini disebut sebagai *live streaming*. Kemudian jika ada alternatif yang membuat pengguna dapat *me-request* media dari *server* konten seperti perpustakaan, ini disebut *on-demand*. *Video On Demand* merupakan kegiatan *streaming* aliran data *video* dan *audio* yang sudah terekam sebelumnya. Ada 4 komponen dalam arsitektur *streaming*, antara lain:

- *Capture dan Encoding*
- *Serving*
- *Distribution dan Delivery*
- *Media Player*



**Gambar 2.3** Komponen *Streaming*  
(Sumber: Austerberry, 2004: 139)

#### *Capture dan Encoding*

Proses *capture* dan *encoding* mengolah sinyal *audio* dan *video* dalam format televisi dan mengkonversi sinyal tersebut menjadi *file streaming*. Ada berbagai macam tingkatan proses, yakni:

- Proses mengambil format *file* komputer
- Proses kompresi data media
- Proses pemaketan data dengan indeks

### **Serving**

*File* yang telah dienkripsi kemudian diunggah menuju *server* agar bisa dikirim ke luar jaringan. *Server* adalah sebuah aplikasi perangkat lunak, seperti *server web*, namun lebih dari sekedar perangkat keras *server*. Sebuah *server streaming* bukan hanya sebuah *server file* atau data, *server* ini juga sebagai pengatur pengiriman aliran data secara *real time*. Kemampuan ini membedakan *server streaming* dengan *server web* pada umumnya.

### **Distribution**

Pada prinsipnya, distribusi itu sederhana. Selama ada koneksi IP antara *server* dengan pengguna, paket yang di-*request* akan sampai pada pengguna. Pada prakteknya, hal ini sangat sulit. Masalah yang terjadi adalah dahulu internet tidak didesain untuk mendukung aliran data secara terus-menerus dalam suatu koneksi yang tetap. Internet secara mendasar adalah sistem dengan *packet switch* untuk data asinkron. Sehingga dilakukan pengembangan meliputi pelebaran *bandwidth* baik melalui modem kabel/DSL sehingga kecepatan hingga 1 Mbps dapat diterapkan, dan berbagai instalasi *fiber* maupun protokol internet untuk meningkatkan QoS.

### **Player**

*Media Player* digunakan untuk mendekodekan *file streaming* dan biasanya tersedia gratis/*free download* maupun sudah terinstal bersamaan dengan *operating system* (OS). (Austerberry, 2004: 138).

## **2.4 Parameter Video**

Beberapa parameter yang mempengaruhi kualitas suatu *video*, antara lain: resolusi, *pixel* dan *frame rate*.

- Resolusi

Resolusi menyatakan ukuran panjang dan lebar sebuah *frame*/gambar. Jumlah resolusi yang digunakan semakin besar menyebabkan jumlah bit yang akan dikirim/diterima semakin besar namun ukuran gambar/*frame* akan semakin leluasa untuk ditayangkan.

- *Pixel*

*Pixel* adalah bagian terkecil yang menyusun suatu gambar. *Pixel* memiliki ukuran kedalaman. Semakin besar nilai kedalaman *pixel* menyebabkan jumlah bit yang digunakan

untuk merepresentasikan 1 *pixel* semakin banyak dan ketajaman serta kejernihan gambar semakin baik. Ukuran kedalaman *pixel* gambar, antara lain: 8 bit (256 warna), 16 bit (65.536 warna) dan 32 bit (16.777.216 warna),

- *Frame Rate*

*Frame Rate* menyatakan kelajuan *frame* tiap detik dalam suatu *video*. *Frame Rate* biasanya ditetapkan dengan satuan fps (*frame per second*). Semakin besar *frame rate* suatu *video* maka semakin banyak *frame* yang dikirim/diterima dan semakin besar jumlah bit yang dikirim/diterima dan semakin halus pula pergerakan *video*-nya.

## 2.5 Protokol Pendukung Video Streaming

Protokol adalah sebuah aturan yang mengatur komunikasi data. Protokol merepresentasikan sebuah persetujuan antara dua perangkat. Tanpa protokol, 2 perangkat mungkin tersambung namun tidak dapat berkomunikasi, sama halnya seperti seseorang yang berbahasa Prancis tidak dapat dipahami oleh orang yang berbahasa Jepang. (Forouzan, 2007: 5).

Beberapa protokol yang dapat digunakan dalam streaming, antara lain:

### a. Transfer Control Protocol (TCP)

TCP adalah protokol transportasi aliran data yang handal. Protokol ini bersifat *connection-oriented*: Sebuah koneksi harus sudah terpasang antara ujung-ujung dari transmisi sebelum keduanya bisa mentransmisikan data. TCP terletak pada *Transport Layer* dari protokol TCP/IP. Walaupun bersifat *connection-oriented*, TCP tidak cocok digunakan untuk trafik multimedia yang interaktif karena TCP tidak memiliki fitur untuk mentransmisikan ulang paket yang hilang/rusak.

### b. User Datagram Protocol (UDP)

UDP adalah protokol yang lebih sederhana dari dua standar protokol pada layer *transport* dari TCP/IP. Protokol ini adalah protokol proses ke proses yang hanya menambahkan alamat port, *error control checksum*, dan panjang informasi data dari layer lebih atas. Protokol ini bersifat *connection-less*. UDP lebih cocok digunakan untuk trafik multimedia yang interaktif. Namun UDP tetap membutuhkan protokol tambahan yakni RTP.

### c. Real-time Transport Protocol (RTP)

RTP adalah protokol yang didesain untuk menangani trafik *real-time* dari internet. RTP tidak memiliki mekanisme (*multicast*, *port numbers* dan lain-lain); RTP harus digunakan bersamaan dengan UDP. Kontribusi utama dari RTP adalah *time-stamping*, *sequencing* dan fasilitas *mixing*.

## 2.6 Perangkat Keras Pendukung Jaringan VLAN

Pada penerapannya, teknologi VLAN membutuhkan perangkat yang sama seperti LAN. Dalam pembuatan jaringan VLAN ada beberapa peralatan yang dibutuhkan, seperti *router*, *switch*, *media converter*, kabel UTP, dan *patchcord*. Perangkat tersebut masing-masing memiliki fungsi yang berbeda-beda.

### 2.6.1 Router

Router merupakan sebuah device atau alat yang dapat menghubungkan dua atau lebih jaringan komputer yang berbeda. Secara umum router adalah alat pada suatu jaringan komputer yang bekerja di network layer pada lapisan OSI. Dalam router ini terdapat routing table yaitu tabel yang berisi alamat-alamat jaringan yang dibutuhkan untuk menentukan tujuan dari paket-paket data yang akan dilewatkan pada suatu jaringan tersebut. (Yugianto, dkk. 2012)

Untuk membuat router, bisa memanfaatkan suatu jenis sistem operasi Windows, Unix, Linux atau jenis sistem operasi lain pada komputer PC dengan hanya menambahkan dua buah network interface card (NIC).

Di dalam PC router dapat di pasang beberapa aplikasi yang dibutuhkan untuk aplikasi webHTB supaya dapat berjalan dengan baik pada sistem. Salah satu diantaranya adalah Webservice Apache, MySQL dan *Network Address Translation* (NAT).



**Gambar 2.4** Router

(Sumber: [www.mikrotik.co.id](http://www.mikrotik.co.id))

### 2.6.2 Switch

*Switch* merupakan perangkat keras penghubung di dalam jaringan komputer yang lebih banyak digunakan saat ini dibandingkan *Hub*. Hal ini disebabkan karena dengan fungsi yang serupa dengan *Hub*, *switch* memiliki dua buah kelebihan utama dibandingkan *Hub*. Kelebihan-kelebihan yang dimiliki oleh *Switch*, yaitu:

1. *Switch* memiliki kemampuan untuk membaca alamat fisik (*MAC Address*) dari setiap komputer yang terhubung ke dalam *Switch* bersangkutan. *Switch* menyimpan alamat fisik (*MAC Address*) dari setiap komputer yang terhubung ke dalam *switch* tersebut beserta dengan nomor *port switch* yang digunakan oleh komputer bersangkutan.
2. *Switch* memiliki kemampuan untuk melakukan filter terhadap paketdata yang keluar masuk *switch*. Hal ini akan memberikan keamanan paket data (terkait dengan pengendalian paket data didalam jaringan komputer). Hal terpenting lainnya adalah memberikan kemudahan didalam memberikan informasi mengenai alamat tujuan untuk komputer penerima (*Destination Address*) serta kemampuan untuk keluar menuju ke komputer tujuan (*Outgoing Port*)

*Switch* bekerja di dua buah *layer* pada jaringan komputer, yaitu *Data Link layer* dan *Physical Layer*. Pada *Data Link Layer*, terjadi proses pengecekan terhadap alamat fisik jaringan (*MAC Address*) untuk otentifikasi alamat fisik komputer yang terhubung pada *switch*, untuk kemudian disesuaikan dengan alamat jaringan pada *Network Layer* (*IP Address*). Pada *Physical Layer* terjadi proses pengolahan sinyal digital. (Pratama, 2014:487)



**Gambar 2.5** *Switch*

(Sumber: [www.tp-link.com](http://www.tp-link.com))

*Switch* yang digunakan dalam perancangan ini adalah jenis *Switch Manageable*. Jenis *switch* ini dapat dikonfigurasi untuk beberapa layanan, seperti port VPN, bandwidth control, port trunking, port-based VLAN, dan lain-lain.

Konfigurasi *port* VLAN dilakukan dengan masuk ke dalam konfigurator *switch* Zyxel ES-2108. Pada *port* tersebut pembagian VLAN dilakukan berdasarkan *tagged* VLAN, dimana pengguna dapat menandakan *port* mana saja yang dijadikan *port* VLAN. Sistem kerja *port* tersebut adalah dengan memilih data yang sesuai dengan *port* VLAN yang dituju atau diteruskan.

### 2.6.3 Media Converter

*Media converter* adalah sebuah perangkat yang umumnya berfungsi mengubah sinyal listrik yang digunakan oleh tembaga pada kabel *Unshielded Twisted Pair* (UTP) ke dalam gelombang cahaya yang digunakan dalam kabel serat optik. Kabel serat optik digunakan apabila jarak antara dua perangkat melebihi jarak transmisi kabel tembaga. Konversi tembaga ke serat optik dengan menggunakan *media converter* memungkinkan perangkat dengan menggunakan kabel tembaga diperpanjang melalui kabel serat optik.

*Media converter* juga dapat digunakan sebagai konversi jenis fiber antara jenis single-mode dan multi-mode dengan menggunakan aliran data *Bi-Directional*. *Media converter* juga dapat mengkonversi antara panjang gelombang untuk aplikasi *Wavelength Division Multiplexing* (WDM). Protokol *media converter* biasanya khusus dan tersedia untuk mendukung berbagai jenis jaringan dan kecepatan data. (Irvine, 2008)



**Gambar 2.6** *Media Converter*

(Sumber: [www.tp-link.com](http://www.tp-link.com))

Dalam perancangan jaringan VLAN pada penulisan ini, penggunaan setiap *media converter* harus memiliki panjang gelombang yang berbeda yaitu 1310nm dan 1550nm. Penggunaan perbedaan panjang gelombang ini dimaksudkan karena fiber optik yang digunakan memiliki panjang gelombang yang berbeda untuk transmisinya (*receiver* dan *transmitter*).

#### 2.6.4 *Unshielded Twisted Pair (UTP)*

Kabel UTP (*Unshielded Twisted Pair*) adalah kabel jaringan yang paling umum dan paling banyak digunakan di dalam kehidupan sehari-hari untuk jenis jaringan komputer bermedia kabel (*wired network*). Jenis kabel ini berbahan tembaga dengan harga yang jauh lebih rendah dibandingkan dengan kabel *Coaxial*. Kabel jaringan UTP (*Unshielded Twisted Pair*) ini memiliki setidaknya delapan buah kategori, namun yang umum digunakan di dalam kehidupan sehari-hari Cat5e (*Enhanced Category 5*) dan Cat5 (*Category 5*). Hal ini disebabkan oleh arena kedua kategori dari UTP (*Unshielded Twisted Pair*) memiliki dukungan transmisi paket data di dalam jaringan komputer sebesar 100 Mbps (Mega bit per second).

Di dalam merangkai sebuah jaringan komputer memanfaatkan kabel UTP (*Unshielded Twisted Pair*), setidaknya perlu diketahui mengenai dua jenis pengkabelan yang umum digunakan. Kedua jenis tersebut meliputi *Straight Through* dan *Cross Over*.

1. *Straight Through* umum digunakan pada dua buah perangkat yang berbeda didalam jaringan komputer. Misalkan saja untuk menghubungkan komputer ke perangkat *switch* maupun *hub* atau menghubungkan perangkat *switch* ke router dan lain-lain.
2. *Cross Over* umum digunakan pada dua buah perangkat yang sama didalam jaringan komputer. Misalkan saja untuk menghubungkan dua buah komputer secara langsung (*point to point*) ataupun menghubungkan dua buah *switch* atau *hub* antarjaringan. (Pratama, 2014:526)



**Gambar 2.7** Kabel UTP

(Sumber: [www.guru.technosains.com](http://www.guru.technosains.com))

## 2.7 Komunikasi Serat Optik

Serat optik atau *Fiber Optic (optical fiber)* merupakan media transmisi pada jaringan komputer yang menggunakan teknologi berupa serat optik (serat gelas murni) berbentuk kabel, dengan kehalusan melebihi rambut manusia, namun secara teknis memiliki kemampuan pengiriman data yang sangat cepat. Dengan kemampuannya ini, Serat Optik banyak digunakan sebagai media transmisi untuk saluran telekomunikasi, termasuk juga untuk jaringan komputer saat ini. Misalkan saja untuk jaringan kabel internet internasional dibawah laut, jaringan internet dari penyedia layanan internet, dan lain-lain.

Sistem komunikasi serat optik merupakan sebuah sistem komunikasi untuk telekomunikasi dan jaringan komputer yang memanfaatkan serat optik sebagai media transmisi dalam bentuk gulungan kabel, yang sangat andal untuk proses komunikasi dan transmisi paket data dengan kecepatan tinggi. Hal ini menjadikan dunia internet makin baik, dengan adanya transmisi paket data yang makin tinggi. Sehingga kedepannya, layanan di internet akan terus bertambah dan berkembang ke arah yang lebih baik.



**Gambar 2.8** Patchcord Optical Fiber

(Sumber: <http://www.abblg.com>)

Perancangan jaringan VLAN ini menggunakan jenis patchcord optical fiber dengan connector SC (*Subscriber Connector*). *Patch cord* adalah kabel serat optik yang pada dua sisi ada konektor. Patch cord digunakan untuk menghubungkan *device* atau dikenal juga dengan *optic jumper*. Panjang *patch cord* yang digunakan sepanjang tiga meter dan memiliki jangkauan panjang gelombang 1310-1550 nm. Sementara penggunaan konektor dari *patch cord* tersebut disesuaikan dengan konektor yang ada pada *media converter*.

## 2.8 Perangkat Lunak Wireshark

Wireshark adalah perangkat lunak *open source* dan bersifat gratis yang banyak digunakan orang-orang di dunia untuk menghitung, menganalisis paket data yang melewati suatu perangkat, umumnya komputer atau laptop. Wireshark akan menganalisa paket data

melalui *interface* dari perangkat internet seperti *wireless adapter* maupun LAN *adapter*. Aplikasi ini mampu menganalisa parameter-parameter performansi jaringan seperti *delay*, *packet loss* dan *throughput*.



**Gambar 2.9** Logo Wireshark  
(Sumber: [www.wireshark.org](http://www.wireshark.org))

## 2.9 Perangkat Lunak VLC Media Player

VLC *Media Player* adalah perangkat lunak yang digunakan untuk memutar file berbentuk musik maupun *video*. VLC dapat menampilkan berbagai jenis codec *video* yang ada seperti halnya, .MPEG, .MP4, .mkv dan lain-lain. VLC juga dapat digunakan untuk memutar *video streaming* yang ada di server dengan memasukkan alamat url pada menu pilhan url jaringan.



**Gambar 2.10** Logo VLC Media Player  
(Sumber: <http://www.videolan.org/vlc>)

## 2.10 Parameter QoS

Parameter kinerja jaringan VLAN dengan fiber optik yang dianalisis pada penelitian adalah sebagai berikut:

### 2.10.1 Throughput

*Throughput* adalah parameter yang menunjukkan jumlah data yang diterima oleh pengguna dengan benar setelah melalui media transmisi. *Throughput* yang mungkin didapat dengan memperhatikan probabilitas paket diterima dalam keadaan salah ( $\rho$ ). Persamaan untuk *bitrate* atau *throughput* tanpa *losses* secara matematis adalah:

$$\lambda_{FL} = \lambda_{FLVideo} + \lambda_{FLAudio} \quad (2-1)$$

$$\lambda_{FL} = Width \times Height \times Frame Rate \times 0,07 \times Motion rank$$

Keterangan :

$\lambda_{FL}$  = *Throughput* tanpa *losses* (bps)

$\lambda_{FLVideo}$  = *Throughput video* (bps)

$\lambda_{FLAudio}$  = *Throughput audio* (bps)

*Width* = Lebar *video*

*Height* = Tinggi *video*

*Frame Rate* = Jumlah *frame* dalam *video* setiap detiknya

*Motion Rank* = Bernilai 1 hingga 4 tergantung gerakan yang ada pada *video*

Jumlah gerakan dalam *video* dikategorikan menjadi tiga urutan yang disebut *motion rank*. Tiga jenis tersebut adalah *low motion*, *medium motion* dan *high motion*.

- *Low motion* adalah *video* yang memiliki gerakan yang tidak terlalu banyak seperti saat melakukan wawancara atau melakukan *review* suatu produk. Nilai dari *low motion* = 1.
- *Medium motion* adalah *video* yang memiliki gerakan-gerakan yang dinamis dengan beberapa perubahan *scene* atau adanya perubahan *background* secara cukup drastis. Nilai dari *medium motion* = 2.
- *High motion* adalah *video* yang memiliki gerakan yang sangat dinamis seperti *action movies* dengan pergerakan dan perubahan *scene* yang sangat cepat. Nilai dari *high motion* = 4.

### 2.10.3 Delay

*Delay* adalah waktu yang dibutuhkan untuk mengirim sebuah paket dari sumber menuju ke tujuan (ujung ke ujung). Menurut ITU-T G.114, kualitas suatu jaringan jika dilihat dari delay-nya ditampilkan sebagai berikut:

Tabel 2.1 Pengelompokan *delay* Berdasarkan ITU-T G.114

<i>Delay</i> (ms)	Kualitas
0-150	Baik
150-400	Cukup, masih dapat diterima
>400	Buruk

Sumber: ITU-T G.114, 2000

*Delay* pada analisis yang dilakukan dalam penelitian ini, terdapat dua macam *delay*, yaitu *delay end-to-end* dan variasi *delay (jitter)*. *Delay end-to-end* adalah waktu yang dibutuhkan untuk mengirim sebuah paket dari sumber ke tujuan. Sedangkan variasi *delay*

(*jitter*) merupakan variasi waktu tunda, hal ini diakibatkan oleh variasi-variasi dalam panjang antrian pengolahan data, waktu penghimpunan ulang paket-paket di akhir perjalanan, waktu transmisi dan propagasi.

*Delay end-to-end* aplikasi *video on demand* atau *video streaming* merupakan jumlah *delay* CODEC aplikasi *video streaming* ditambah dengan *delay* jaringan dimana aplikasi itu berjalan. *Delay* CODEC aplikasi *video streaming* dipengaruhi berbagai macam faktor seperti algoritma kompresi, proses *sampling*, dan lain-lain. Setiap metode *encoding* menggunakan MPEG dengan *simple profiles* akan mengalami *delay* sebesar 50 ms (Vigato, 2005: 52). Hal ini cocok digunakan untuk aplikasi *video conference*, maupun *streaming*.

$$t_{\text{total}} = t_{\text{CODEC}} + t_{\text{net}} \quad (2-2)$$

Keterangan:

$t_{\text{total}}$  = *delay end-to-end* untuk aplikasi *video streaming* (detik)

$t_{\text{CODEC}}$  = *delay codec* (detik)

$t_{\text{net}}$  = *delay* jaringan (detik)

$$t_{\text{net}} = t_{\text{proses}} + t_{\text{transmisi}} + t_{\text{propagasi}} + t_{\text{antrian}} + t_{\text{dekapsulasi}} \quad (2-3)$$

$t_{\text{proses}}$  = *delay* proses (detik)

$t_{\text{transmisi}}$  = *delay* transmisi (detik)

$t_{\text{propagasi}}$  = *delay* propagasi (detik)

$t_{\text{antrian}}$  = *delay* antrian (detik)

$t_{\text{dekapsulasi}}$  = *delay* dekapsulasi (detik)

$$t_{\text{codec}} = t_{\text{video}} + t_{\text{audio}} \quad (2-4)$$

Keterangan:

$t_{\text{codec}}$  = *Delay codec* (detik)

$t_{\text{video}}$  = *Delay codec video* (detik)

$t_{\text{audio}}$  = *Delay codec audio* (detik)

Tabel 2.2 Spesifikasi Codec Video dan Audio

Video CODEC	Bit Rate (kbps)	MaximumPayload (byte)	DelayCODEC (ms)
AVC/H.264 MPEG4	96-384	254	16-50
Audio CODEC	Bit Rate (kbps)	MaximumPayload (byte)	DelayCODEC (ms)
AAC	128	63	24-60

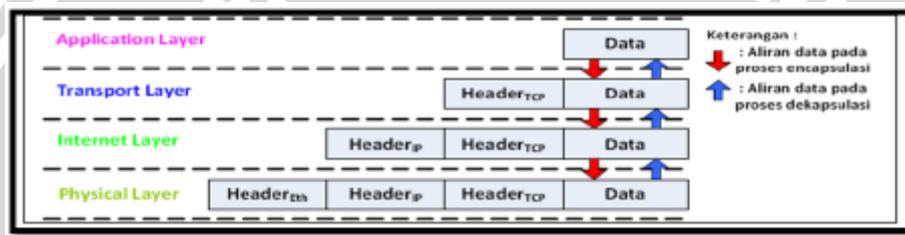
(Sumber: RFC 3640, 2003, dan RFC 3984, 2005)

### 2.10.2.1 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.

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. Perhitungan *delay* enkapsulasi dan *delay* dekapsulasi didapatkan menggunakan persamaan (2-5) dan (2-6)



**Gambar 2.11** Proses Enkapsulasi dan Dekapsulasi Paket Data.

(Sumber: <http://www.citap.com/document/tcp-ip/tcpip011.htm>)

Berikut adalah persamaan untuk *delay* enkapsulasi

$$t_{enc/dec} = \frac{L_{Header\ IPv4} + L_{Header\ NALU} + L_{Header\ UDP} + L_{Header\ RTP} + L_{Header\ Ethernet} + L_{Header\ FCS}}{C_{proses}} \times 8 \quad (2-5)$$

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

Keterangan:

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

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

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

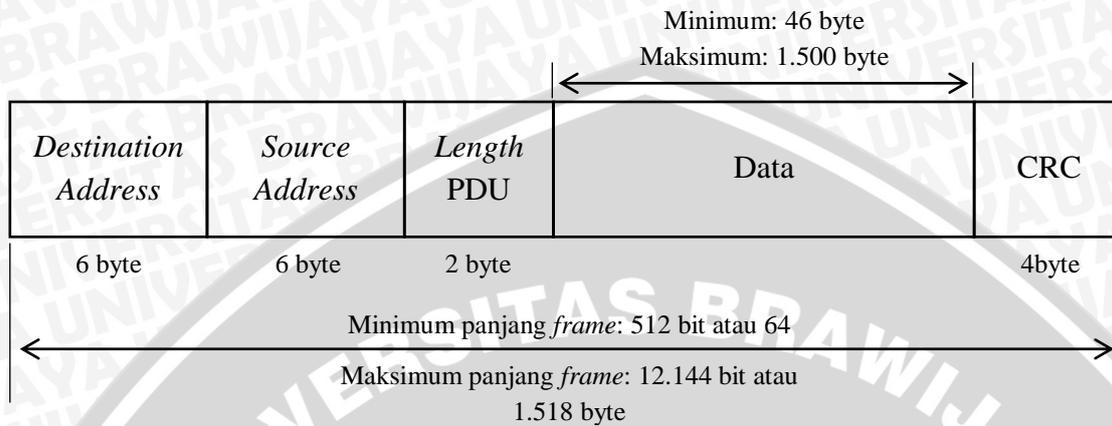
$T_k$  = waktu pengiriman total data (s)

Tabel 2.3 *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 Redundancy Check	4 byte	Forouzan, 2007
Ethernet Header	14 byte	Forouzan, 2007

### 2.10.2.2 Delay Transmisi

*Delay* transmisi adalah waktu yang dibutuhkan untuk meletakkan sebuah paket data ke media transmisi. Dipengaruhi ukuran paket data dan kecepatan transmisi. *Delay* transmisi dapat ditentukan dengan Persamaan 2.7 (Forouzan, 2007).



**Gambar 2.12** Minimum dan Maksimum Panjang *Frame*  
(Sumber: Forouzan, 2007)

$$t_{transmisi} = \frac{(L+L')}{B} \quad (2-7)$$

Keterangan:

$t_t$  = *delay* transmisi (s)

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

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

$B$  = kecepatan transmisi (bps)

### 2.10.2.3 Delay Propagasi

*Delay* propagasi adalah waktu yang dibutuhkan untuk mentransmisikan data dari *server* ke user. Nilai *delay* propagasi dirumuskan dalam Persamaan 2.8 (Behrouz A. Forouzan, 2007).

Tabel 2.4 Kecepatan Propagasi Media Transmisi

Jenis Media Transmisi	Kecepatan Propagasi ( $c = 3 \times 10^8$ m/s)	Sumber
Serat Optik	0,66c	Nicoletti, Pietro. 2004. <i>IEEE 802.3 - ISO 8802.3 CSMA/CD</i> . (www.studioreti.com)
UTP	0,64c	<a href="http://communications.draka.com/sites/eu/Datasheets/SuperCat5_24_U_UTP_Install.pdf">http://communications.draka.com/sites/eu/Datasheets/SuperCat5_24_U_UTP_Install.pdf</a>

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

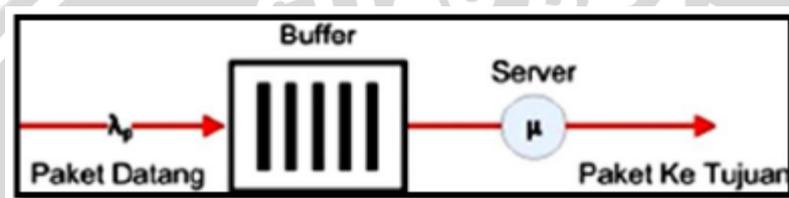
Keterangan :

$t_p$  = *delay* propagasi (s)

- $d_{\max}$  = Jumlah panjang media transmisi  
 $v$  = kecepatan propagasi media transmisi  
 $c$  = cepat rambat gelombang elektromagnetik ( $3 \times 10^8$  m/s)

#### 2.10.2.4 Delay Antrian

*Delay* antrian adalah waktu dimana paket data berada dalam antrian untuk diproses oleh *router*. *Delay* antrian dapat dihitung dengan menggunakan model antrian M/M.1. M pertama menunjukkan kedatangan *Poisson*, M kedua berarti distribusi waktu pelayanan eksponensial, dan 1 menunjukkan bahwa jumlah pelayannya adalah tunggal. Besarnya *delay* antrian yang terjadi pada *router* ditentukan dengan Persamaan 2.9 (Mischa Schwartz, 1987: 42).



**Gambar 2.13** Model Antrian M/M/S

(Sumber: Mischa Schwartz, 1987: 42)

Kecepatan kedatangan paket pada *server* ditentukan persamaan Little's Formula. Persamaannya adalah sebagai berikut:

$$\lambda p = \frac{N}{T} \quad (2-9)$$

Keterangan:

$\lambda p$  = kecepatan kedatangan paket pada *server* (paket/s)

$N$  = Total paket yang dikirim (paket)

$T$  = Waktu pengiriman paket total (s)

Kecepatan pelayanan ditentukan oleh persamaan di bawah ini (Mischa Schwartz):

$$\mu = \frac{C}{L_t} \quad (2-10)$$

Keterangan:

$\mu$  = kecepatan pelayanan *router* (paket/s)

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

$C$  = Kapasitas kanal (bps)

Sedangkan waktu tunggu paket pada *router* ( $t_{\text{queue}}$ ) dihitung dengan menggunakan Persamaan 2.11 (Mischa Schwartz, 1987: 42).

$$t_{\text{queue}} = 1/\mu/(1 - \rho) \quad (2-11)$$

kecepatan kedatangan paket pada *router* ditentukan dengan Persamaan 2.12:

$$\rho = \frac{\lambda}{\mu} \quad (2-12)$$

Keterangan:

$t_{\text{queue}}$  = waktu tunggu paket pada *router* (s)

$\lambda$  = kecepatan kedatangan paket pada *router* (paket/s)

$\rho$  = faktor utilitas ( $0 < \rho < 1$ )

$\mu$  = kecepatan pelayanan router (paket/s)

### 2.10.2 Packet Loss

*Packet Loss* adalah jumlah paket IP yang hilang selama prosestransmisi dari *source* menuju *destination*. Salah satu penyebab *packet loss* adalah antrian yang melebihi kapasitas *buffer* pada setiap *node*. Beberapa penyebab terjadinya *packet loss* yaitu *congestion*, *node* yang bekerja melebihi kapasitas *buffer*, *memory* yang terbatas pada *node* dan *policing*. Berdasarkan ITU-T G.1010, aplikasi streaming memiliki persyaratan *packet loss* yang berbeda-beda berdasarkan media yang dikirimkan. Berikut adalah tabel yang menunjukkan syarat *packet loss* berdasarkan ITU-T G.1010,

Tabel 2.5 Standar *Packet Loss* ITU-T G.1010 untuk Aplikasi *Streaming*

<i>Medium</i>	<i>Application</i>	<i>Degree of Symetry</i>	<i>Information Loss</i>
<i>Audio</i>	<i>Conversational Voice</i>	<i>Two-way</i>	<3% <i>Packet Loss Ratio</i>
<i>Audio</i>	<i>Voice Messaging</i>	<i>One-way</i>	<3% <i>PLR</i>
<i>Audio</i>	<i>High Quality Audio Streaming</i>	<i>One-way</i>	<1% <i>PLR</i>
<i>Video</i>	<i>Videophone</i>	<i>Two-way</i>	<1% <i>PLR</i>
<i>Video</i>	<i>Streaming</i>	<i>One-way</i>	<1% <i>PLR</i>

Sumber. ITU-T G.1010, 2002

Persamaan (2-13) untuk *packet loss* ditentukan sebagai berikut (Mischa Schwartz,1987),

$$\text{Packet Loss} = \frac{N_{\text{lost packet}}}{N_{\text{packet}} - N_{\text{lost packet}}} \times 100\% \quad (2-13)$$

Keterangan:

*Packet Loss* = Presentase *Packet Loss* (%)

$N_{\text{lost Packet}}$  = Jumlah paket data yang hilang (paket)

$N_{\text{Packet}}$  = Jumlah paket data yang diterima dengan benar (paket)

Semakin kecil paket data yang hilang dalam suatu pentransmisian, maka akan semakin kecil nilai *packet loss*-nya dan semakin baik kualitas jaringannya.