

BAB I

PENDAHULUAN

1.1. Latar Belakang

Menonton video atau film merupakan salah satu hiburan yang dapat dilakukan oleh banyak orang untuk saat ini, mulai dari anak-anak sampai orang dewasa. Dengan berkembangnya teknologi memudahkan kita untuk mengakses video dari mana saja. Namun terkadang video yang diakses tidak berjalan dengan lancar atau terjadi *buffering*. Hal ini dapat terjadi dikarenakan banyak faktor, dari sisi pengguna salah satu penyebabnya karena koneksi yang lambat saat memutar video dengan resolusi tinggi. Sedangkan dari sisi penyedia layanan atau server dapat disebabkan oleh karakteristik data media, aplikasi, dan *hardware* (CPU, *memory*, dan *Storage*) [1].

Berdasarkan permasalahan yang terjadi [1] dan pada percobaan sebelumnya [2], penggunaan virtualisasi dapat meningkatkan performansi pada media server. Seiring berkembangnya teknologi virtualisasi, serta pengembangan server berbasis virtual menjadi solusi untuk meningkatkan kinerja server dalam memproses banyak data dengan sumber daya terbatas. Docker merupakan salah satu *platform container* yang saat ini banyak digunakan oleh perusahaan-perusahaan dibidang *Computer Software* dan *IT Service*. Penggunaan Docker dinilai lebih baik dari VM (*Virtual Machine*) karena memiliki skalabilitas yang lebih cepat serta penyimpanan pada *images* yang lebih kecil[2]. Selain itu pemilihan protokol yang digunakan nantinya akan mempengaruhi server dalam mengolah data yang akan dikirimkan kepada penggunanya dan pengaturan *bitrate* agar dapat menyesuaikan dengan jaringan yang disediakan oleh penggunanya, sehingga pengguna dapat menonton video tanpa menunggu proses *buffring* yang terlalu lama.

Pada tulisan ini, penulis akan mengimplementasikan media server menggunakan NGINX ke dalam virtualisasi berbasis Docker untuk meningkatkan *performance* pada server. Protokol HLS (*HTTP Live Streaming*) dan MPEG-DASH (*Dynamic Adaptive Streaming over HTTP*) digunakan untuk perbandingan. Selama server berjalan, status CPU dan *memory* akan dipantau oleh sebuah *scripts* menggunakan perintah Bash (Unix Shell) yang dituliskan ke dalam file dengan

format csv. Kualitas layanan yang diuji menggunakan parameter QoS (*Quality of Service*).

1.2. Rumusan Masalah

Berdasarkan latar belakang di atas, permasalahan yang akan dibahas adalah sebagai berikut:

- Bagaimana menerapkan media server menggunakan Docker ?
- Bagaimana performansi CPU dan *memory* media server saat *client* mengakses *video stream* ?
- Apakah penggunaan CPU dan *memory* berkurang dibandingkan menggunakan *virtual machines* ?
- Bagaimana performansi protokol HLS dan MPEG-DASH terhadap perubahan *bitrate* berdasarkan penggunaan *bandwidth* ?
- Apakah perubahan *bitrate* dapat memperlancar *streaming video* ?
- Bagaimana hasil kualitas jaringan atau QoS saat menggunakan *bandwidth* yang berbeda ?

1.3. Tujuan dan Manfaat

Tujuan Tugas Akhir ini adalah untuk meningkatkan kualitas jaringan dengan menggunakan pengaturan *bitrate* video yang berbeda pada protokol HLS dan MPEG-DASH serta memperbaiki *performance* media server menggunakan Docker dan melakukan analisis pada QoS dan performansi server.

Tugas Akhir ini memiliki manfaat sebagai media pembelajaran yang diharapkan dapat menjadi pembelajaran dan pengembangan media server untuk ke depannya.

1.4. Batasan Masalah

Batasan masalah dari Tugas Akhir ini adalah :

- Implementasi menggunakan *container* Docker.
- Implementasi menggunakan NGINX sebagai media server.
- Tidak membahas keamanan jaringan.
- Menggunakan sistem operasi ubuntu untuk instalasi Docker.

- Menggunakan aplikasi OBS Stream untuk melakukan *video stream* dengan format video .mp4 menggunakan codec H.254 resolusi 720 *pixel*.
- Menggunakan VLC dan *web browser* Google Chrome untuk analisis.
- Menggunakan Bash *scripts* untuk mengambil data penggunaan CPU dan *memory*.
- Tugas akhir ini melakukan analisis meliputi QoS, performansi server (CPU dan *memory*), dan tidak melakukan analisis QoE.
- Pengujian CPU dan *memory* menggunakan jmeter dilakukan dengan mengkonfigurasi *client* sebanyak 50, 100, dan 150.

1.5. Metode Pengerjaan

Dalam Tugas Akhir ini berikut langkah – langkah yang dilakukan

1. Studi Literatur

Dalam Studi Literatur dilakukan dengan mempelajari beberapa *referensi* yang nantinya mampu menunjang untuk melakukan penelitian dan pengerjaan Tugas Akhir. Referensi yang dapat digunakan adalah bersumber dari buku – buku, paper, *white paper*, jurnal, serta sumber – sumber lain yang berhubungan dengan penelitian dan pengerjaan Tugas Akhir.

2. Perancangan Sistem

Tahap Perancangan Sitem dilakukan perancangan dan percobaan sistem yang akan dibuat, seperti alur instalasi *tools* yang digunakan, membangun server di *computer server*, serta *konfigurasi* di *computer server*.

3. Implementasi dan Pengujian

Memimplementasikan rancangan sistem yang telah dibuat. Mebuat server yang kemudian akan diakses oleh *client* sehingga *client* bisa melakukan *streaming* melalui web kemudian melakukan pengujian perfomansi pada *computer server*.

4. Analisa Sistem

Sitem yang telah diimplementasi dan dilakukan pengujian akan dianalisi perfomasansinya pada *computer server* dan *client* berdasarkan penggunaan CPU dan *memory*.

5. Dokumentasi

Penyusunan laporan untuk mendokumentasikan seluruh kegiatan Tugas Akhir.

1.6. Jadwal Pelaksanaan

Tabel 1.1 Jadwal pelaksanaan.

No.	Deskripsi Tahapan	Durasi	Tanggal Selesai	<i>Milestone</i>
1	Desain Sistem	2 minggu	22 Feb 2020	Diagram Blok dan spesifikasi <i>Input-Output</i>
2	Implementasi Media Server, dll	1 bulan	5 Maret 2020	Server dapat berjalan dan diakses
3	Pengujian QoS dan pengelolaan data	2 minggu	4 April 2020	Data setiap parameter
4	Penyusunan laporan/buku TA	2 minggu	13 Juni 2020	Buku TA selesai