

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pada jaringan komputer saat ini, setiap fungsi jaringan biasanya dilakukan oleh *appliance* dengan perangkat keras *proprietary*. *Software* dan *hardware* dalam *appliance* sengaja dibuat tidak bisa dipisahkan dan tergantung satu sama lain. Contoh dari *appliance* yg dimaksud antara lain adalah: perangkat DPI (*Deep Packet Inspection*), CDN (*Content Distribution Network*) *Appliance*, *Router*, *Firewall*, *Load-Balancer*, NAT (*Network Address Translators*), *Session Border Controller*, *Mobile Base Station Controller*, *Mobile Packet Gateway*, DNS (*Domain Name System*) dll^[1].

Pesatnya pertumbuhan *cloud computing* dalam beberapa tahun terakhir terkait *load balancer as a service* pada *cloud computing environment* sangat menarik dan impresif untuk diimplementasikan. Tetapi teknologi yang dibutuhkan dalam membangun *load balancer as a service* masih dalam tahap penelitian^[12]. Saat ini *platform* bisnis *e-commerce* menjadi bagian penting dalam pembangunan bisnis *e-commerce*. Seiring berkembangnya bisnis *platform* saat ini, memberikan layanan yang lebih cepat dan lebih handal telah menjadi masalah yang perlu dipecahkan dengan segera^[14]. Meningkatnya pertumbuhan *cloud computing* sebagai *platform e-commerce* menyebabkan peningkatan jumlah trafik serta permintaan pada layanan *cloud server*, sehingga beban yang diterima oleh *server* menjadi semakin besar. Dengan kondisi seperti ini akan mengakibatkan *overload request* pada sebuah layanan *cloud server* karena meningkatnya CPU *usage* pada saat melayani *request* dari *client* yang berlebihan di waktu yang bersamaan. Di lain sisi, *client* menginginkan kecepatan akses yang maksimal. Ada beberapa solusi dalam permasalahan ini, diantaranya yaitu *load balancer*, *multipath routing* dan *bandwidth management*.

Salah satu solusi adalah penggunaan *load balancer*. *Load balancer* adalah sebuah perangkat yang digunakan sebagai pengatur beban trafik dalam suatu jaringan di sejumlah *server* menggunakan teknik *load balancing*. *Load balancing* adalah teknik untuk mendistribusikan beban trafik pada dua atau lebih jalur koneksi secara seimbang, agar trafik dapat berjalan optimal, memaksimalkan *throughput*, memperkecil waktu tanggap dan menghindari *overload* pada salah satu jalur koneksi. *Load balancing* digunakan pada saat sebuah *server* telah memiliki jumlah *request* yang telah melebihi maksimal kapasitasnya. *Load balancing* juga mendistribusikan beban kerja secara merata di dua atau lebih komputer,

link jaringan, CPU, *hard drive*, atau sumber daya lainnya, untuk mendapatkan pemanfaatan sumber daya yang optimal. *Load balancer* juga digunakan untuk meningkatkan kapasitas (pengguna bersamaan) dan reliabilitas suatu *server*.

Dalam tugas akhir ini telah diimplementasikan dan dianalisa kinerja *load balancer as a service* di dalam *cloud environment* berbasis *openstack* dengan menggunakan algoritma *round robin*, *least connection*, dan *source ip* dengan parameter yang akan diuji meliputi *throughput*, *cpu utilization*, *bandwidth usage* dan *frame drop*.

1.2 Tujuan

Adapun tujuan dari penyusunan Tugas Akhir ini antara lain :

1. Dapat mengimplementasikan *load balancer as a service* di dalam *cloud environment* berbasis *openstack*.
2. Dapat mengkonfigurasi beban kerja *server* agar beban kerja tidak terpusat oleh satu *server* saja.
3. Mengukur performansi *load balancer as a service* dengan parameter *throughput*, *cpu utilization*, *bandwidth usage* dan *frame drop*.
4. Dapat mengetahui performansi terbaik *load balancer as a service* dengan membandingkan algoritma *round robin*, *least connection* dan *source ip* dengan melihat parameter *throughput*, *cpu utilization*, *bandwidth usage* dan *frame drop*.

1.3 Rumusan Masalah

Rumusan masalah pada Tugas Akhir ini antara lain :

1. Bagaimana mengimplementasikan *load balancer as a service* dengan algoritma *round robin*, *least connection* dan *source ip* pada *openstack*?
2. Bagaimana mengukur performansi *load balancer as a service* dengan algoritma *round robin*, *least connection* dan *source ip* dengan parameter *throughput*, *cpu utilization*, *bandwidth usage* dan *frame drop*?
3. Bagaimana sistem kerja dari *load balancer as a service*?
4. Perangkat apa saja yang akan dibutuhkan untuk mengimplementasikan *load balancer as a service*?

1.4 Batasan Masalah

Adapun yang menjadi batasan masalah pada Tugas Akhir ini adalah :

1. Hanya mengimplementasikan *load balancer as a service* dengan algoritma *round robin*, *least connection*, dan *source ip*.
2. Parameter performansi yang diamati meliputi parameter *throughput*, *cpu utilization*, *bandwidth usage* dan *frame drop*.
3. Tidak membahas tentang keamanan jaringan.
4. Tidak membahas *Internet Protocol Version 6*.
5. Layanan yang digunakan adalah *streaming server*.
6. Link yang digunakan berkapasitas maksimal 100 Mbps.

1.5 Metode Penelitian

Metode penelitian yang digunakan dalam pengerjaan Tugas Akhir ini antara lain:

a. Studi literatur dan eksperimen

Mempelajari teori-teori *openstack mitaka* yang mendukung pelaksanaan tugas akhir ini dari beberapa referensi, literatur, serta jurnal yang terkait dalam penelitian ini.

b. Tahap perancangan dan desain

Setelah studi literatur dilaksanakan, maka dilanjutkan dengan melakukan perancangan dan konfigurasi *hardware* yang akan digunakan.

c. Tahap implementasi dan pengukuran

Setelah melakukan perancangan maka dilanjutkan dengan proses implementasi sesuai konfigurasi dan dilakukan pengukuran menggunakan *wireshark*, *htop*, *flazr*, *vnstat*, *vlc*.

d. Analisis

Analisis berdasarkan hasil perancangan dan konfigurasi sesuai dengan parameter-parameter yang diinginkan.

1.6 Sistematika Penulisan

Sistematika penulisan tugas akhir ini adalah sebagai berikut :

BAB I PENDAHULUAN

Bab ini mencakup latar belakang, tujuan, rumusan masalah, batasan masalah, metode penelitian dan sistematika penulisan.

BAB II DASAR TEORI

Bab ini mencakup teori tentang *video streaming*, *load balancing* serta algoritma *load balancing*, *virtualisasi*, *hypervisor*, *cloud computing* dan *openstack*.

BAB III PERANCANGAN DAN IMPLEMENTASI SISTEM

Bab ini membahas mengenai langkah – langkah pengerjaan tugas akhir ini yang direpresentasikan dalam bentuk *flowchart* dan penentuan skenario pengambilan data.

BAB IV PENGUJIAN DAN ANALISIS

Bab ini membahas analisis hasil pengujian sistem sesuai dengan skenario yang telah ditentukan.

BAB V KESIMPULAN DAN SARAN

Bab ini berisi kesimpulan dan saran yang berkaitan dengan tugas akhir ini yang dapat digunakan untuk perbaikan atau pengembangan lebih lanjut.