

# BAB I PENDAHULUAN

## 1.1 Latar Belakang

Semakin berkembangnya zaman, penggunaan internet baik dalam hal mencari informasi maupun menyediakan informasi, membuat kebutuhan akan pengalamatan (*IP address*) di internet semakin membesar. Terutama penggunaan *World Wide Web* (WWW) menjadi infrastruktur dasar untuk layanan internet.

*Single web server* yang mayoritas digunakan perusahaan untuk layanan berbasis *web*, masih memiliki beberapa kekurangan yang tergolong fatal. Ketika mendapatkan *request* yang banyak dari *client*, *server* yang menerima permintaan (*request*) dari *user* seketika bisa mengalami *overload* serta *crash*. Melihat tingkat aktivasi dari *user* yang begitu tinggi, tentu saja hal ini berdampak pada perusahaan selaku penyedia informasi. Dan dalam kinerja jaringan, kualitas layanan (*Quality of Service*) juga menjadi acuan penting pada kemampuannya mengirim arus data dalam satu satuan waktu.

Ketika *server* sedang diakses oleh *user*, maka *server* tersebut sedang terbebani karena harus melayani proses permintaan dari *user*. Semakin banyak *user* yang mengakses *server* menyebabkan proses yang dilakukan oleh *server* itu semakin banyak. Apabila satu *server* terbebani lebih, maka *server* tersebut tidak bisa melayani banyak *user* karena kemampuan melakukan *processing* ada batasnya. Solusi yang paling ideal adalah membagi-bagi beban yang datang ke beberapa *server*, maka dikembangkanlah suatu teknik yang dapat mengatasi masalah tersebut. Teknik itu disebut dengan *load balancing* [1].

*Load balancing* merupakan salah satu teknik *routing* yang memanfaatkan beberapa sumber daya/trafik ke *server* untuk dapat digunakan secara bersamaan tanpa mengurangi kinerja *server*. Dan menggunakan algoritma penjadwalan untuk mengatur bagaimana *server-server* yang ada dalam *cluster* mendapatkan beban trafik yang seimbang sehingga memaksimalkan *throughput* dan *response time server* [2].

Sebagai sebuah sarana pengalamatan di internet, *IP address* sekarang ini semakin menjadi barang yang eksklusif dan tidak sembarang orang bisa mendapatkannya secara valid dengan mudah. Oleh karena itu, dibutuhkanlah suatu

mekanisme yang dapat memberikan solusi permasalahan dalam menghemat IP *address*. Logika sederhana untuk penghematan IP *address* adalah dengan meng-*share* suatu nomor IP *address* valid ke lebih dari satu komputer ke jaringan internet walaupun hanya memiliki satu IP *address* yang valid. Salah satu mekanisme itu disediakan oleh *Network Address Translation* (NAT).

*Network Address Translation* (NAT) merupakan suatu teknologi yang menggunakan jaringan IP *private* untuk menghubungkan lebih dari satu komputer ke jaringan internet dengan menyederhanakannya menjadi satu alamat IP *address* [3]. NAT berperan untuk melindungi jaringan seperti halnya *firewall* dengan hanya mengizinkan koneksi yang berasal dari dalam jaringan. Hal ini meningkatkan tingkat keamanan jaringan komputer pada jaringan *private* yang terkoneksi ke internet, serta fleksibilitas dalam administrasi jaringan.

LVS (*Linux Virtual Server*) merupakan sebuah perangkat lunak yang menerapkan metode *load balancing* sebagai pembagi kerja bagi beberapa *server*. LVS memiliki sepuluh algoritma penjadwalan yang berguna untuk mengirimkan permintaan (*request*) sesuai dengan kemampuan masing-masing *server* agar dapat melayani pengguna secara optimal. Algoritma tersebut yaitu *Round Robin* (RR), *Weighted Round Robin* (WRR), *Least Connection* (LC), *Weighted Least Connection* (WLC), *Locality Based Least Connection* (LBLC), *Locality Based Least Connection with Replication* (LBLCR), *Destination Hashing* (DH), *Source Hashing* (SH), *Shortest Expected Delay* (SED), dan *Never Queue* (NQ).

Penelitian mengenai *load balancing* sudah banyak dilakukan pada berbagai jenis *server*, seperti penelitian yang dilakukan oleh Sampurna Dadi Riskiono, Selo Sulistyono, dan Teguh Bharata Adji [4] dengan judul “**KINERJA METODE LOAD BALANCING DAN FAULT TOLERANCE PADA SERVER APLIKASI CHAT**”, yang membahas kinerja dari metode *load balancing* dengan replikasi *database* (*load balancer master* dan *slave*) dalam mengatasi beban *server* pada layanan aplikasi *chat* berdasarkan *response time* dan *throughput* serta *fault tolerance* sebagai pendeteksi kerusakan secara *real-time* pada sistem (*load balancer*) ketika terjadi kondisi yang tidak mendukung. Pada penelitian tersebut di dapatkan, bahwa penerapan metode *load balancing* mampu mengatasi permintaan 3000 koneksi dibandingkan tanpa metode *load balancing* dan penggunaan *fault*

*tolerance* mampu mengatasi kegagalan dengan menerapkan perpindahan *load balancer master* (gagal) ke *load balancer slave* selama 2 detik dan kondisi sebaliknya sehingga ketersediaan sistem terjaga.

Selanjutnya penelitian yang dilakukan oleh Yoppi Lisyadi O [5] dengan judul **“MEMBANGUN SISTEM CLOUD COMPUTING DENGAN IMPLEMENTASI LOAD BALANCING DAN PENGUJIAN ALGORITMA PENJADWALAN LINUX VIRTUAL SERVER PADA FTP SERVER”**, yang membahas perbandingan performansi *server* tunggal konvensional dengan sistem *cloud computing* dan pengujian algoritma penjadwalan *load balancing* berdasarkan *response time* dan *throughput* pada FTP dengan menggunakan metode *direct routing*. Pada penelitian tersebut didapatkan, bahwa sistem *cloud computing* performansinya lebih baik dari sistem konvensional dan algoritma *never queue* (nq) di pilih sebagai algoritma terbaik untuk aplikasi FTP dibandingkan 4 algoritma lainnya.

Penelitian lainnya, dilakukan oleh Suryayusra [6] dengan judul **“MENGATASI OVERLOAD MENGGUNAKAN LINUX VIRTUAL SERVER SEBAGAI LOAD BALANCING PADA SERVER E-LEARNING UNIVERSITAS BINA DARMA”**, membahas performansi dari server *e-learning* yang harus melayani 4.437 pengguna dengan bermacam kegiatan berdasarkan *response time* dan *throughput* yang di dapatkan dengan menggunakan metode topologi LVS-NAT. Pada penelitian tersebut di dapatkan, bahwa penggunaan algoritma penjadwalan *Round Robin* mampu menunjang beban besar dalam mengatasi kegiatan *e-learning* dengan cara membagi deklarasi penjadwalan berdasarkan dua kelompok (IP *Private* dan IP *Global*) dengan *network* yang berbeda.

Dari uraian di atas, maka penulis ingin mengembangkan penelitian mengenai algoritma penjadwalan *load balancer* pada penggunaan layanan *Network Address Translation* (NAT) dan menjadikannya sebagai tugas akhir dengan judul **“ANALISA PERBANDINGAN PERFORMANSI ALGORITMA PENJADWALAN PADA TOPOLOGI NAT”**. Penulis memilih judul ini karena ingin mengetahui kinerja performansi dari sepuluh algoritma penjadwalan LVS

atau *load balancer* yang efektif untuk diterapkan pada topologi *Network Address Translation* (NAT) berdasarkan parameter *response time* dan *throughput* pada *FTP server*, sehingga dapat memberikan hasil yang optimal terhadap kinerja *server*.

## 1.2 Rumusan Masalah

Permasalahan yang akan dibahas pada tugas akhir ini, yaitu:

1. Apa parameter performansi *load balancing* pada *FTP server* dengan topologi NAT yang akan digunakan dalam penelitian ini.
2. Bagaimana cara memilih di antara sepuluh algoritma penjadwalan LVS yang akan diterapkan pada *FTP server*.

## 1.3 Tujuan Penelitian

1. Menentukan algoritma penjadwalan LVS terbaik dan efektif dari sepuluh algoritma penjadwalan LVS pada *FTP server* dengan topologi NAT.
2. Mengetahui hasil perbandingan algoritma penjadwalan *load balancing* LVS pada *FTP server* dengan topologi NAT berdasarkan parameter *response time* dan *throughput*.

## 1.4 Batasan Masalah

Penelitian pada tugas akhir ini akan dibatasi dengan permasalahan berikut:

1. *FTP server* dibangun pada sistem operasi Debian *server 9* dengan menggunakan perangkat lunak vsFTPd.
2. Menggunakan perangkat lunak *ipvsadm* LVS untuk menjalankan *load balancing*.
3. Implementasi *load balancing* menggunakan metode *LVS Network Address Translation* (LVS-NAT).
4. Parameter performansi pada *load balancing* yang akan dianalisa, yaitu *response time* dan *throughput*.
5. Algoritma penjadwalan yang akan diuji adalah *Round Robin* (RR), *Weighted Round Robin* (WRR), *Least Connection* (LC), *Weighted Least Connection* (WLC), *Locality Based Least Connection* (LBLC), *Locality Based Least Connection with Replication* (LBLCR), *Destination Hashing* (DH), *Source Hashing* (SH), *Shortest Expected Delay* (SED) dan *Never Queue* (NQ).

6. Aplikasi yang akan diuji adalah FTP.
7. *Load balancer server* yang digunakan sebanyak satu buah dan jumlah *real server* atau *FTP server* yang digunakan adalah sebanyak dua buah.
8. Pengujian dilakukan sebanyak dua kali terhadap sepuluh algoritma penjadwalan LVS.

### **1.5 Manfaat Penelitian**

Tugas akhir ini diharapkan dapat memberi informasi dan gambaran dari topologi NAT dalam pemanfaatannya untuk layanan pentransferan ataupun jaringan internet dan juga mengetahui algoritma penjadwalan LVS yang terbaik dan efektif untuk diterapkan pada *FTP server* dengan penggunaan topologi NAT, sehingga dapat diimplementasikan dan memberikan hasil yang optimal untuk kinerja *server*.

### **1.6 Sistematika Penelitian**

Adapun sistematika dalam penulisan tugas akhir ini adalah

#### **BAB I PENDAHULUAN**

Bab ini berisi latar belakang penelitian, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah, dan sistematika penulisan.

#### **BAB II TINJAUAN PUSTAKA**

Bab ini berisi teori dasar yang mendukung penelitian tugas akhir ini.

#### **BAB III METODOLOGI PENELITIAN**

Bab ini berisi metodologi, prosedur penelitian, skema jaringan, perangkat lunak, dan perangkat keras yang akan digunakan dalam tugas akhir ini.

#### **BAB IV PEMBAHASAN DAN ANALISIS SISTEM**

Bab ini berisi tentang hasil penelitian dan analisis serta pembahasan dari tugas akhir ini.

## **BAB V      PENUTUP**

Bab ini berisi tentang kesimpulan dan saran.

