

BAB I PENDAHULUAN

1.1 Latar Belakang

Pada masa globalisasi seperti saat ini, seluruh sektor yang ada dalam kehidupan terus mengalami perkembangan, salah satunya dalam sektor teknologi dan komunikasi yang mengalami perkembangan pada jaringan internet. Dengan adanya jaringan internet dapat memungkinkan setiap orang di belahan dunia manapun terhubung dengan mudah dan dengan internet pula layanan informasi yang ingin diketahui bisa didapatkan dengan mudah. Sehingga kebanyakan sektor saat ini bergantung kepada peranan sebuah jaringan internet. Seperti halnya politik, bisnis, pendidikan dan social budaya [1]. Internet berkaitan erat dengan alat mediator yang dinamakan situs web.

Situs web dapat digunakan secara gratis dan bebas oleh semua orang untuk memperoleh berbagai macam informasi yang diinginkan oleh pengguna internet. Akan tetapi meningkatnya penggunaan kebutuhan internet juga dapat menyebabkan akses ke internet melalui jaringan tersebut semakin meningkat. Lalu lintas jaringan yang semakin padat dengan *bandwidth* yang terbatas dapat menyebabkan kinerja jaringan semakin menurun atau akses pada jaringan menjadi lambat bahkan dapat menyebabkan terjadinya gagal akses pada jaringan tersebut [2].

Dengan demikian pembangunan *proxy server* merupakan sebuah solusi yang efektif karena *proxy server* ini dapat menyimpan hasil permintaan *http* sehingga konten yang diminta lebih dekat dengan peminta. Hal ini membuat *proxy server* dapat mempercepat akses dan menghemat pemakaian *bandwidth* jaringan. Namun kinerja server *proxy* juga semakin menurun seiring dengan meningkatnya lalu lintas jaringan yang datang [3]. Oleh karena itu kinerja dari sever *proxy* perlu ditingkatkan agar dapat menjamin akses jaringan yang cepat.

Untuk mengatasi masalah tersebut dapat menggunakan metode *load balancing*. *Load balancing* merupakan metode yang digunakan untuk meningkatkan efektivitas kinerja suatu server dengan cara membagi beban antara server satu dengan server lainnya. Dengan demikian lalu lintas jaringan antar server tidak terlalu padat dan mencegah terjadinya penurunan kinerja pada server [4]. Penerapan *Load balancing* dapat dilakukan dengan menambah satu server baru atau lebih (*multiple server*) sehingga membuat suatu *cluster* atau kelompok server untuk menangani peningkatan beban dengan cara membagi request dari client ke server secara merata.

Penerapan metode *load balancing* sebagai pembagi beban dapat menggunakan salah satu perangkat lunak yaitu *Linux Virtual Server (LVS)*. LVS ini dapat diimplementasikan dengan menggunakan tiga cara, salah satunya adalah

virtual server dengan topologi NAT (*Network Address Translation*). Berdasarkan topologi NAT ini, server yang berperan sebagai pembagi beban akan meneruskan beban/*request* kepada masing-masing *real server* berdasarkan algoritma yang digunakan, lalu *request* diteruskan kembali kepada pembagi beban, dan terakhir pembagi beban meneruskan *request* kepada *user*. Walaupun skalabilitas server virtual melalui NAT terbatas, namun dengan menggunakan topologi NAT ini *real server* dapat menjalankan sistem operasi apa pun yang mendukung protokol TCP/IP (*Transmission Control Protocol* dan *Internet Protokol*). *Real server* dapat menggunakan alamat internet pribadi dan hanya alamat IP yang diperlukan untuk menyeimbangkan beban.

Pembagian tugas dari masing-masing virtual server dapat diatur dengan menggunakan salah satu dari sepuluh algoritma penjadwalan. Penggunaan algoritma penjadwalan ini bertujuan agar server-server dalam *cluster* bisa mendapatkan beban trafik yang seimbang untuk memaksimalkan *throughput*. LVS memiliki sepuluh algoritma penjadwalan yang berguna untuk mengirimkan permintaan (*request*) sesuai dengan kemampuan dari masing-masing server agar dapat melayani pengguna secara lebih optimal. Adapun algoritma yang dimaksud yaitu algoritma *Round Robin* (RR), *Weight Round Robin* (WRR), *Least Connection* (LC), *Weight 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* pada berbagai jenis aplikasi server telah banyak dilakukan, salah satunya yaitu penelitian yang dilakukan oleh Hariyon M. Sani [5]. Penelitian ini membahas dampak dari pengimplementasian *load balancing* pada *proxy-cache server* melalui parameter *response time* dan *throughput*. Algoritma penjadwalan yang digunakan penelitian ini ada tiga, yaitu *Destination Hashing*, *Locality Based Least Connection* dan *Locality Based Least Connection with Replication*. Dari hasil penelitian ini, disebutkan bahwa tingkat ketersediaan server menjadi lebih tinggi dikarenakan pengimplementasian dari *load balancing*. Ketika terdapat kerusakan pada salah satu server, maka proxy masih dapat beroperasi dengan menggunakan *real server* lain yang masih bekerja. Hasil lainnya yang diperoleh dari penelitian ini adalah algoritma penjadwalan yang paling efektif untuk *proxy server* diantara ketiga jenis algoritma penjadwalan yang diuji adalah *Destination Hashing*.

Pada penelitian lainnya yang dilakukan oleh Jefry Alvonsius Rabu, Joko Purwadi dan Willy S. Raharjo, terdapat pembahasan mengenai perbandingan *load balancing* web server menggunakan metode LVS-NAT dengan web server tunggal [6]. Dari hasil penelitian mereka, diperoleh kesimpulan bahwa *load balancing* web server dengan metode LVS-NAT mampu meningkatkan *throughput* hingga 2 kali lipat dari server tunggal dan juga mampu meningkatkan *response time* walau tidak terlalu signifikan.

Dari uraian di atas, penulis ingin melakukan penelitian terhadap kinerja *load balancing proxy server* dengan menggunakan topologi NAT dan menjadikan penelitian ini sebagai tugas akhir dengan judul “**IMPLEMENTASI LOAD BALANCING DAN ANALISA LIMA ALGORITMA PENJADWALAN PADA PROXY CLUSTER DENGAN TOPOLOGI NAT**”. tujuan dari pemilihan judul ini agar dapat mengetahui kinerja dari lima algoritma penjadwalan yang paling efektif untuk diterapkan pada topologi NAT berdasarkan parameter *response time* dan *throughput* pada proxy cluster, sehingga mendapatkan hasil yang paling optimal untuk kinerja server.

1.2 Rumusan Masalah

Perumusan masalah pada penelitian ini adalah sebagai berikut.

1. Bagaimana cara mengimplementasikan *load balancing* dengan menggunakan lima variasi dari algoritma penjadwalan yang digunakan pada aplikasi *proxy*?
2. Bagaimana cara mengetahui kinerja dari *load balancing* dengan menggunakan lima variasi dari algoritma penjadwalan yang digunakan pada aplikasi *proxy*?

1.3 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah.

1. Mengetahui proses mekanisme dari sistem *load balancing* dengan menggunakan topologi NAT pada aplikasi *proxy*.
2. Mengetahui dan menentukan hasil dari algoritma penjadwalan yang paling efektif dari kelima algoritma penjadwalan pada *load balancing proxy cluster* dengan topologi NAT berdasarkan parameter *throughput* dan *response time*.

1.4 Manfaat Penelitian

Manfaat yang dapat diambil dari penulisan Tugas Akhir ini adalah

1. Memberikan informasi tentang bagaimana cara mengimplementasikan *load balancing proxy cluster* dengan menggunakan topologi NAT.
2. Memberikan informasi tentang algoritma penjadwalan yang paling efektif pada *load balancing proxy cluster* dengan menggunakan topologi NAT.

1.5 Batasan Masalah

Batasan masalah yang diterapkan dalam penelitian ini adalah sebagai berikut.

1. Sistem *proxy cluster* dirancang menggunakan *transparent proxy*.
2. Sistem *load balancing* dirancang menggunakan *ipvsadm*.
3. Sistem *load balancing* menggunakan topologi NAT.
4. Memilih algoritma penjadwalan yang paling efektif untuk proxy cluster.

5. Algoritma yang akan diuji pada sistem load balancing yaitu *Locality Based Least Connection with Replication* (LBLCR), *Destination Hashing* (DH), *Source Hashing* (SH), *Shortest Expected Delay* (SED), *Never Queue* (NQ).
6. Parameter untuk pengujian performa pada *load balancing* yang akan dianalisa adalah *response time* dan *throughput*.
7. Aplikasi yang akan diuji adalah *squid*.
8. proxy server yang akan digunakan adalah sebanyak tiga buah dan *load balancing* sebanyak 1 buah.

1.6 Sistematika Penulisan

Draft tugas akhir ini disusun secara sistematis sebagai berikut:

BAB I PENDAHULUAN

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

BAB II TINJAUAN PUSTAKA

Bab ini berisi tentang teori dasar dan materi yang mendukung penelitian tugas akhir ini.

BAB III METODOLOGI PENELITIAN

Bab ini berisi tentang jenis penelitian, diagram alir penelitian, perancangan alat penelitian, alat dan bahan yang dibutuhkan serta tahapan penelitian.

BAB IV HASIL DAN ANALISA

Bab ini berisi tentang penjelasan hasil dan pembahasan dari perancangan alat yang diujikan. Analisa yang dilakukan meliputi analisa saat berbeban, analisa tanpa beban dan analisis perancangan kemampuan turbin angin.

BAB V PENUTUP

Bab penutup ini berisi tentang kesimpulan dari hasil penelitian yang dilakukan dan saran yang didapatkan berdasarkan hasil dan analisa dari penelitian ini.