

BAB 1 PENDAHULUAN

1.1 Latar Belakang

DNS merupakan singkatan dari *Domain Name Server* yang merupakan sebuah sistem yang berfungsi sebagai identitas berupa nama pada *Internet Protocol Address* (IP Address). DNS adalah sebuah sistem yang menghubungkan *Uniform Resource Locator* (URL) dengan *IP Address* [1]. *User* harus mengingat nama alamat (DNS) dari suatu *website* dan menggunakan DNS tersebut untuk mencari halaman web yang diinginkan. Dari pada harus mengingat deretan angka *IP Address*, penerapan DNS ini dirasa lebih memudahkan *user* untuk mencari halaman web yang dibutuhkan. Setelah mengetik DNS dari suatu *website* pada *address bar*, maka komputer akan menerjemahkan domain tersebut ke dalam *IP Address* yang dipahami komputer. Dengan kata lain, DNS ini bisa disebut sistem yang menjembatani *user* dengan komputer.

Seiring dengan perkembangan ilmu pengetahuan di bidang teknologi dan komunikasi (IPTEK), kebutuhan terhadap sistem informasi dan data semakin penting. Penggunaan internet sebagai sumber informasi telah menjadi prioritas bagi umat manusia. Kemudahan dalam penggunaan internet merupakan salah satu penyebab dari semakin tingginya minat dalam penggunaan internet. Sebuah penelitian menyebutkan bahwa pengguna internet di seluruh dunia telah mencapai angka 4,5 miliar orang. Angka ini menunjukkan bahwa lebih dari 60 persen penduduk dunia telah menjadi pengguna internet [2]. Lalu pada penelitian lain, pengguna internet di Indonesia itu sendiri saat ini telah mencapai 175,4 juta dengan penetrasi mencapai 64 persen. Angka ini meningkat dari tahun lalu sebesar 17 persen, atau sekitar 25 juta [3]. Data pertama merupakan hasil riset dari layanan manajemen konten *HootSuite* dan agensi pemasaran media sosial *We Are Social* pada tahun 2019, dan data kedua adalah riset terbaru dari layanan manajemen konten *HootSuite* dan agensi pemasaran media sosial *We Are Social* dalam laporan yang bertajuk “Digital 2020”.

Meningkatnya pengguna internet ini memiliki dampak yang besar terhadap protokol-protokol komunikasi data internet, termasuk DNS server. DNS server

merupakan salah satu protokol *transport* UDP (*User Datagram Protocol*) sebagai pengirim informasi dalam lalu lintas internet yang mengandalkan kecepatan di atas keandalan/keakuratan [4]. Konsekuensi dari hal ini adalah semakin banyaknya beban/*request* yang diterima oleh DNS server dari suatu website. Jika hal ini tidak ditanggulangi, maka bisa saja suatu ketika DNS server mengalami kelebihan beban (*overload*) yang akan merugikan pengguna saat berselancar di internet. Oleh karena itu, diperlukan perancangan yang tepat dalam membangun DNS server. Salah satu solusi yang dapat diterapkan untuk mengatasi permasalahan ini adalah dengan menerapkan teknologi *load balancing* terhadap DNS server.

Load balancing merupakan salah satu cara yang digunakan untuk meningkatkan kinerja dan tingkat ketersediaan server. *Load balancing* dapat diterapkan dengan menambah satu server baru atau lebih (*multiple server*) dan membuat suatu *cluster* atau kelompok untuk menangani peningkatan beban. Ketika beban meningkat, maka beban yang datang akan dibagi ke beberapa server sekaligus, sehingga beban yang ditanggung oleh masing-masing server menjadi lebih ringan. Dengan kata lain, jika salah satu server tidak dapat melayani permintaan pengguna (*server down*), maka secara otomatis server yang lain langsung menggantikannya, sehingga pengguna seakan-akan tidak mengetahui bahwa server tersebut *down*. Tingkat ketersediaan server bisa tetap terjaga dengan penggunaan *load balancing* ini.

Salah satu perangkat lunak yang dapat menerapkan metode *load balancing* sebagai pembagi beban ke beberapa server adalah *Linux Virtual Server* (LVS). LVS ini dapat diimplementasikan dalam 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 tertentu, 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 Protocol*). *Real server* dapat

menggunakan alamat internet pribadi dan hanya alamat IP yang diperlukan untuk penyeimbang beban.

Pembagian tugas masing-masing virtual server diatur dengan suatu algoritma penjadwalan. Tujuan dari penggunaan algoritma penjadwalan ini adalah agar server-server dalam *cluster* mendapatkan beban trafik yang seimbang untuk memaksimalkan *throughput*. LVS memiliki sepuluh algoritma penjadwalan yang berguna untuk mengirimkan permintaan (*request*) sesuai dengan kemampuan masing-masing server agar dapat melayani pengguna secara 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* telah banyak dilakukan pada berbagai jenis aplikasi server, salah satunya yaitu penelitian yang dilakukan oleh Zafrullah Ahmad Shaddiq [5]. Penelitian ini membahas dampak dari implementasi *load balancing* pada DNS server dan perbandingannya dengan DNS tunggal melalui parameter *response time* dan *throughput*. Algoritma penjadwalan yang digunakan ada empat, yaitu *Weighted Round Robin*, *Least Connection*, *Shortest Expected Delay* dan *Never Queue*. Dari hasil penelitian, disebutkan bahwa dengan mengaplikasikan *load balancing*, tingkat ketersediaan server menjadi lebih tinggi. Ketika terdapat kerusakan pada salah satu server, maka DNS masih bisa beroperasi dengan *real server* lain yang masih bekerja. Hasil lainnya yang diperoleh adalah algoritma penjadwalan terbaik untuk DNS diantara empat jenis algoritma penjadwalan yang diuji adalah *Shortest Expected Delay*, sedangkan yang terburuk adalah *Weighted Round Robin*.

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* DNS server menggunakan topologi NAT dan menjadikan penelitian ini sebagai tugas akhir dengan judul “**Analisa Kinerja Algoritma Penjadwalan Load Balancing untuk Aplikasi DNS dengan Topologi NAT**”. Penulis memilih judul ini karena ingin mengetahui efektifitas dari penggunaan sepuluh algoritma penjadwalan *Load Balancing* terhadap kinerja layanan DNS server berdasarkan parameter *response time* dan *throughput* sehingga dapat memberikan hasil yang optimal terhadap kinerja server. Penulis menggunakan aplikasi BIND9 (*Berkeley Internet Name Domain* versi 9) sebagai sistem yang menyimpan, mengontrol, dan menerjemahkan nama domain menjadi *IP Address* atau sebaliknya.

1.2 Rumusan Masalah

Permasalahan yang akan diselesaikan dalam pembuatan Tugas Akhir ini adalah sebagai berikut.

1. Kinerja *load balancing* DNS server terhadap penggunaan sepuluh algoritma penjadwalan berdasarkan parameter *response time* dan *throughput*.
2. Analisis algoritma penjadwalan yang paling efektif diterapkan pada DNS server.

1.3 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah untuk menganalisis hasil perbandingan sepuluh algoritma penjadwalan *load balancing* LVS berdasarkan parameter *response time* dan *throughput* serta menentukan algoritma penjadwalan yang paling efektif untuk diterapkan pada *cluster* DNS dengan topologi NAT.

1.4 Batasan Masalah

Batasan masalah yang diterapkan dalam penelitian ini adalah sebagai berikut.

1. Server yang diuji adalah DNS server yang dibangun pada sistem operasi Debian versi 10.

2. Menggunakan IPVSADM untuk menjalankan *load balancing*.
3. Implementasi *load balancing* menggunakan teknik virtual server dengan topologi NAT.
4. Memilih algoritma penjadwalan yang cocok untuk DNS server.
5. Algoritma yang diuji yaitu *Round Robin* (RR), *Weighted 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), *Never Queue* (NQ).
6. Parameter yang diuji yaitu *response time* dan *throughput*.
7. *Load balancer* server yang digunakan sebanyak satu buah dan jumlah DNS server yang digunakan adalah sebanyak dua buah.
8. Spesifikasi komputer DNS server yang digunakan adalah sama.

1.5 Manfaat Penelitian

Tugas akhir ini diharapkan dapat menambah pengetahuan tentang cara melakukan *load balancing* DNS server dengan topologi NAT serta mengetahui algoritma penjadwalan LVS yang paling efektif untuk diterapkan pada DNS server.

1.6 Sistematika Penulisan

Sistematika penulisan adalah langkah-langkah dalam pembuatan tugas akhir. Tujuannya adalah untuk mempermudah dan memperjelas penyampaian informasi dan pembahasan masalah, dengan susunan sebagai berikut:

BAB I : PENDAHULUAN

Pendahuluan berisi tentang latar belakang penelitian, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah, dan sistematika penulisan.

BAB II : TINJAUAN PUSTAKA

Tinjauan pustaka yang berisi teori dasar yang mendukung penelitian.

BAB III : METODOLOGI PENELITIAN

Metodologi penelitian berisikan tentang langkah-langkah beserta penjelasan mengenai penelitian yang dilakukan.

BAB IV : HASIL DAN PEMBAHASAN

Hasil dan pembahasan ini berisikan analisa dan penelitian ini.

BAB V : PENUTUP

Penutup berisikan beberapa kesimpulan dan saran yang bisa ditarik dan disampaikan yang didasari dari hasil dan pembahasan penelitian ini.

