

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pada masa sekarang, internet merupakan salah satu kebutuhan untuk sebagian besar orang di dunia ini, baik itu bagi pelajar, guru, ilmuwan dan usahawan. Internet sangat sering dimanfaatkan oleh berbagai kalangan untuk mencari informasi secara cepat dan dalam waktu yang singkat. Hal tersebut menyebabkan permintaan informasi oleh pengguna internet semakin meningkat setiap harinya, sehingga data yang disimpan dan juga yang ditransmisikan dalam jaringan internet akan semakin besar. Sistem pengelolaan *database* atau *database management system* (DBMS) perlu dikelola dengan baik agar kuat dan handal dalam menangani jumlah data yang semakin meningkat.

Ketersediaan layanan akan data yang handal merupakan permasalahan yang dihadapi oleh perusahaan atau instansi yang mengelola data dalam jumlah yang banyak setiap harinya. *Server* merupakan salah satu infrastruktur yang berperan penting dalam mengelola data-data yang nantinya dibutuhkan oleh pengguna (*client*). *Server* merupakan salah satu perangkat lunak yang menerima suatu perintah untuk menyediakan jenis layanan tertentu dari *client*[1], seperti layanan aplikasi *e-learning* maupun aplikasi yang memiliki data-data informasi lainnya.

Pada masa sekarang ini, teknologi informasi juga digunakan dibidang pendidikan. Proses belajar mengajar menjadi lebih menarik dan interaktif dengan bantuan teknologi informasi. Beberapa contoh penerapan teknologi informasi di bidang pendidikan seperti penggunaan *Power Point* dan *Adobe Flash* sebagai aplikasi untuk mempresentasikan materi bagi pengajar. Seiring dengan perkembangan internet, pembelajaran juga memanfaatkan internet, diantaranya adalah penggunaan *e-learning*.

E-Learning berasal dari gabungan 2 kata, yakni *electronic* dan *learning*. Dalam arti lain, *e-learning* adalah proses belajar mengajar yang dilakukan secara *online*. Dalam pelaksanaannya, *e-learning* menggunakan teknologi informasi sebagai sarana pembelajarannya[2]. *E-Learning* digunakan sebagai salah satu cara untuk tetap melaksanakan kegiatan belajar mengajar tanpa harus tatap muka secara langsung.

Moodle merupakan salah satu *e-learning* yang banyak digunakan pada saat ini, *Moodle* mengimplementasikan *Learning Management System* (LMS) yang banyak digunakan di beberapa kampus di Indonesia, salah satu nya Universitas Andalas dengan penamaan iLearn (ilearn.unand.ac.id).

Seiring meningkatnya penggunaan iLearn di Universitas Andalas, *server down* sering terjadi ketika iLearn di akses oleh banyak pengguna di waktu yang sama. Dengan semakin banyaknya pengguna dan peningkatan permintaan dari kebutuhan pengguna maka kerja dari *server* tersebut menjadi bertambah berat. Oleh karena itu, diperlukan suatu metode yang dapat mengatasi pengontrolan beban tersebut agar mendapatkan hasil kinerja *server* yang maksimal.

Terdapat beberapa metode yang dapat digunakan untuk meningkatkan skalabilitas sebuah *server* agar dapat bekerja dengan baik seiring dengan jumlah pengguna yang terus bertambah. Metode yang pertama yaitu dengan menambah kapasitas *hardware* yang digunakan untuk *server* seperti menambah jumlah *server* ataupun meng-*upgrade* komponen yang terdapat pada *server*. Dengan menggunakan metode ini sementara dapat menyelesaikan masalah tersebut, akan tetapi masalah tersebut akan muncul kembali seiring dengan jumlah pengguna yang semakin tinggi lagi, sehingga memerlukan biaya yang besar. Metode kedua yang dapat digunakan yaitu *Load Balancing* dengan menggunakan perangkat lunak *Ipvsadm*. *Ipvsadm* digunakan untuk menerapkan *load balancing* yang digunakan untuk membagi atau memisahkan antara 2 atau lebih jalur koneksi secara seimbang, yang dapat membantu koneksi internet menjadi seimbang[3]. Dengan penggunaan metode ini dinilai sangat efektif dalam mengoptimalkan pembagian jalur koneksi dan *bandwidth* pada setiap user yang datang ke *server* secara sekaligus. Metode ini hanya membagi beban ke *server* tersebut bukan untuk menambah bandwith dari sebuah *server*. Dengan *load balancing* pada database *server* dapat mengurangi waktu respon yang diterima oleh user. Metoda *load balancing* dapat membagi beban yang akan di kirim ke *server* dengan berbagai macam algoritma.

Berdasarkan pada website *linuxvirtualserver.org*, terdapat beberapa algoritma pada *load balancing* yang berguna dalam mendistribusikan beban. Algoritma

penjadwal berfungsi untuk mengatur dan membagi beban trafik pada *server*. Pada penelitian ini algoritma yang digunakan pada *load balancing* yaitu algoritma *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)*.

Agar menghindari kegagalan pada ketersediaan *database*, digunakan replikasi *master to master* dengan menggunakan *MySQL* sebagai media manajemen basis data yang ada. Dari penelitian yang dilakukan, *Ipsadm* mampu membagi beban secara merata pada setiap *server*. Selain itu, penerapan replikasi bekerja dengan baik dimana *file* yang di *upload* pada ke 2 *database* akan saling sinkron karena sinkronisasi ini bersifat dua arah.

Dan juga seiring dengan perkembangan internet ini, banyak oknum yang mencoba memanfaatkan internet untuk mencuri data yang sangat penting seperti data perbankan, data rekam medis dan juga data akademik kampus. Oleh karena itu diperlukan sebuah sistem yang dapat mengatasi keamanan data pada saat proses pengiriman. Untuk mengatasi masalah keamanan dari komunikasi antara *user* dan *server* tersebut, maka topologi yang dapat diterapkan pada metoda *load balancing* salah satu nya yaitu topologi *IP Tunneling*. Pada *IP Tunneling*, terjadi suatu proses perubahan sebuah kode dari yang dapat dimengerti menjadi sebuah kode yang tidak dapat dimengerti atau disebut enkapsulasi sehingga tidak dapat dibaca dengan mudah oleh orang lain yang ingin mengambil data tersebut saat dikirim dari satu alamat IP ke alamat IP lainnya[4]. Dengan *IP Tunneling*, seolah-olah terbentuk suatu terowongan yang dapat mengamankan data yang dikirim sehingga akan sulit diketahui oleh pihak lain.

Pada penelitian [5], membahas mengenai kinerja dari metode *load balancing* dengan sinkronisasi *file web server*. Pada penelitian tersebut didapatkan, bahwa sistem *load balancing* yang dibangun dapat bekerja dengan baik pada saat datangnya 10.000 *request* dari *client* tanpa terjadi nya *error request*. Pada penelitian tersebut

menyerankan ada nya penerapan *load balancing* pada *cluster database server* agar performa dari *web server* dapat berjalan dengan baik.

Selanjutnya pada penelitian [6], membahas mengenai kinerja dari metode *load balancing* pada *MySQL Cluster* berdasarkan *Transaction per Second* (TPS) dan *response time* dengan menggunakan metode *direct routing*. Pada penelitian tersebut di dapatkan, bahwa *response time MySQL Cluster* dengan *load balancing* memiliki nilai lebih baik yaitu 854,04 ms pada algoritma penjadwalan *round robin* dengan beban *threads* 256 sedangkan *MySQL Cluster default* di dapatkan *response time* sebesar 1161,26 ms.

Penelitian lainnya [7], membahas mengenai performansi algoritma penjadwalan pada protokol *http* menggunakan metode *load balancing via NAT*. Pada penelitian tersebut di dapatkan, bahwa dari sepuluh algoritma penjadwalan yang diujikan, algoritma penjadwalan WRR dipilih sebagai algoritma penjadwalan terbaik karena hasil uji *throughput* yang didapatkan lebih baik dibandingkan dengan algoritma penjadwalan lainnya.

Dengan permasalahan yang sudah dijabarkan sebelumnya, karena itulah penulis melakukan penelitian ini yaitu “**Analisis Kinerja Algoritma Penjadwalan Sistem Load Balancing Pada Database Cluster Dengan Topologi Tunneling**”. Penulis memilih judul ini bertujuan untuk mengetahui kinerja dari sepuluh algoritma penjadwalan *load balancer* yang efektif untuk diterapkan pada topologi *tunneling* berdasarkan parameter *response time* dan *throughput* pada *database cluster*, sehingga mendapatkan hasil yang optimal pada kinerja *server*.

1.2 Perumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan di atas, rumusan masalah yang didapatkan adalah :

1. Bagaimana metode *load balancing* dapat menyelesaikan masalah pada koneksi jaringan yang tidak stabil?
2. Analisa data yang didapatkan menggunakan sepuluh algoritma penjadwalan dan topologi IP *tunneling*.

1.3 Tujuan Penelitian

Tujuan penelitian dari tugas akhir ini adalah :

1. Mengetahui proses mekanisme sistem *load balancing* dengan topologi *tunneling* untuk aplikasi *database*.
2. Menentukan dan mengetahui hasil algoritma penjadwalan terbaik dan efektif dari sepuluh algoritma penjadwalan pada *load balancing database cluster* dengan topologi *tunneling* berdasarkan parameter *throughput* dan *response time*.

1.4 Manfaat Penelitian

Hasil penelitian dan penulisan tugas akhir ini diharapkan dapat memberikan beberapa manfaat berupa :

1. Memberikan informasi perancangan *load balancing database cluster* menggunakan topologi *tunneling*.
2. Memberikan informasi tentang algoritma penjadwalan yang terbaik dan terefektif pada *load balancing database cluster* dengan topologi *tunneling*.

1.5 Batasan Masalah

Dalam penulisan tugas akhir ini terdapat beberapa batasan masalah sebagai berikut:

1. Perancangan sistem *cluster database* menggunakan *replikasi master to master*.
2. Perancangan sistem *load balancing* menggunakan *Ipvsadm*.
3. Algoritma yang digunakan pada *load balancing* 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)*.
4. Topologi yang digunakan pada sistem *load balancing* adalah *IP tunneling*.
5. Aplikasi yang diuji adalah *database MySQL*.
6. Parameter pengujian performa pada *load balancing* yang akan dianalisa adalah *throughput* dan *response time*.
7. *Database server* yang digunakan sebanyak dua buah dan *load balancer server* yang digunakan sebanyak satu buah.

1.6 Sistematika Penulisan

Pada laporan tugas akhir ini, disusun dalam beberapa bab dengan sistematika penulisan sebagai berikut :

BAB I PENDAHULUAN

Bab ini membahas mengenai latar belakang dari masalah dalam penelitian ini, tujuan yang ingin dicapai, manfaat yang akan didapatkan, batasan masalah, dan sistematika penulisan

BAB II TINJAUAN PUSTAKA

Bab ini membahas mengenai teori-teori pendukung yang digunakan dalam menyelesaikan masalah dalam tugas akhir ini.

BAB III METODOLOGI PENELITIAN

Bab ini memberikan informasi mengenai bagaimana langkah-langkah dalam menyelesaikan tugas akhir ini.

BAB IV DATA DAN PEMBAHASAN

Bab ini berisi hasil pengujian yang dilakukan dan analisa dari hasil pengujian yang telah dilakukan.

BAB V PENUTUP

Bab ini terdiri dari kesimpulan dan saran.



