

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi informasi yang pesat, otomatis seiring juga dengan meningkatnya para pengguna yang terhubung pada jaringan *internet*. Berawal dari sebuah *single server* yang selalu mendapatkan *request* dari banyak *user*, perlahan tapi pasti akan terjadi *overload* dan *crash* sehingga berdampak pada *request* yang tidak dapat dilayani oleh *single server*. *Desain arsitektur cluster* dapat di bangun dengan menggunakan konsep *network load balancing* yang memungkinkan proses pengolahan data di *share* ke dalam beberapa komputer.

Dalam penelitian ini menggunakan *algoritma* penjadwalan *Round Robin* sebagai solusi alternatif mengatasi permasalahan *overload* data pada *server* yang dapat mempengaruhi kinerja. *Algoritma* penjadwalan yang digunakan dalam teknik *load balancing* ini yaitu *Round Robin*. Maka *performansi algoritma* penjadwalan dapat diamati dengan menekankan pada parameter sebagai berikut yaitu *throughput*, *respon time*, *reply connection* dan *error connection* sehingga didapatkan *algoritma* penjadwalan terbaik. Secara otomatis dalam proses *load balancing* mampu mengurangi beban kerja setiap *server* sehingga tidak ada *server* yang *overload* dan memungkinkan *server* menggunakan *bandwidth* yang tersedia secara lebih efektif serta menyediakan akses yang cepat ke *web browser* yang di *hosting*.

Server adalah suatu sistem komputer yang menyediakan berbagai macam jenis – jenis layanan tertentu yang ditunjukkan untuk *client* dalam suatu sistem jaringan komputer. *Server* dilengkapi oleh sistem operasi (OS) yang khusus untuk mengontrol ataupun memonitor akses dan juga sumber daya yang terdapat di dalamnya, selain itu *server* juga didukung oleh *prosesor* yang bersifat *scalable* serta RAM yang kapasitas besar, dan dilengkapi oleh sistem operasi yang khusus, disebut sebagai sistem operasi jaringan komputer.

Server merupakan sebuah tempat yang dipenuhi dengan berbagai jenis informasi dimana *server* memiliki tugas utama untuk memberikan sebuah layanan bagi para *client* yang terhubung dengannya. Sebuah komputer dapat memiliki peran sebagai *server*, *client* atau bahkan keduanya. Misalkan saja, jika menggunakan sebuah komputer A untuk mengakses *website* milik pengguna B, maka kini berperan sebagai *client*. Sebaliknya, jika pengguna B menggunakan komputernya untuk mengakses *website*, maka anda kini berperan sebagai *server*. Konsep tersebut lebih dikenal dengan sebutan konsep *peer to peer*.

Dengan menggunakan sebuah komputer *server*, berbagai biaya dan waktu dapat dipangkas sehingga sebuah kegiatan menjadi lebih ekonomis. Misalnya saja, jika di sebuah perusahaan terdapat sebuah komputer *server* yang terhubung ke semua komputer lainnya sebagai *client*-nya, maka sebuah data dapat dibagikan kesesama *client* dalam jaringan perusahaan tersebut. Kemudian jika ada komputer *client* yang ingin mencetak sebuah *file*, maka bisa langsung melalui komputer *server*, sehingga hanya dibutuhkan satu buah *printer* saja. Tentu jauh lebih ekonomis dibandingkan dengan jika harus melakukan pemasangan *printer* untuk tiap komputer.

Cara kerja *server* sangatlah sederhana, *server* bekerja atas permintaan dari sebuah *client*. Misalnya saja, untuk kasus *web server*, ketika mengetikkan suatu alamat *website* menggunakan *browser*, maka artinya komputer sedang bertindak sebagai *client* yang meminta informasi kepada *web server*. *Web server* tersebut kemudian mengirimkan isi *website* ke komputer, sehingga dapat mengakses isi *website* tersebut.

1.2 Perumusan Masalah

Berdasarkan dari pemaparan latar belakang di atas, penulis dengan ini merumuskan rumusan masalah yang penulis akan kaji.

1. Bagaimana merancang *desain* sistem (*database*, *flowchart*, *activity diagram*), *algoritma* metode *Round Robin*, rancangan tampilan dan sebagainya dalam *Desain Monitoring Load Balancing Server* ?

2. Bagaimana hasil unjuk kerja *Desain Monitoring Load Balancing Server* dengan *Metode Round Robin* ?

1.3 Tujuan Penelitian

Membuat sistem yang dapat digunakan untuk mengumpulkan informasi sebagai pengontrol kelebihan beban (*overload*) yang terjadi pada *server* dengan menggunakan perhitungan algoritma *round robin* agar tidak terjadi masalah (*trouble*).

1.4 Manfaat Penelitian

Dengan penelitian ini diharapkan dapat Membantu pengguna untuk mengumpulkan informasi yang apabila terjadi *server* penuh atau *overload* sehingga dapat diketahui dengan cepat dalam kisaran waktu yang ditentukan.

