

1 Pendahuluan

1.1 Latar belakang

Meningkatnya pengguna internet pada saat ini, mendorong akan meningkatnya trafik internet. Internet sudah menjadi sebuah kebutuhan yang harus dipenuhi. Aplikasi internet seperti *web sites* sudah menjadi bagian dari kehidupan dengan kelebihanannya yang dapat diakses dari manapun. Konten web site semakin interaktif dan semakin beragam, bahkan saat ini sudah digunakan pada teknologi transaksi online, hal ini tentunya berdampak pada *request rate* dari sebuah web sites terlebih web sites populer yang semakin hari semakin meningkat.

Dengan kondisi seperti ini akan mengakibatkan *stress* pada sebuah web server karena meningkatnya *cpu usage* pada saat melayani request konten dari user. Di lain sisi, user mengharapkan *low latency*, *high throughput* dan *high availability*. Distribusi web server menyediakan solusi efektif untuk meningkatkan kualitas dari web server. Kumpulan dari beberapa web server digunakan sebagai kesatuan resource untuk menyediakan service ke client. *Load balancing* merupakan salah satu teknik untuk meningkatkan performansi dari distribusi web server. *Load balancing* dapat menyediakan *high availability* melalui *redundant server* dan mengurangi *latency* dengan membagi load diantara server yang dikenal dengan *load sharing*. Banyak algoritma dan teknik untuk mengimplementasikan *load balancing*. Pada umumnya strategi *load balancing* yang ada, request dialihkan ke server, dimana server dipaksa untuk menerima request dalam kondisi apapun.

Pada tugas akhir ini, mencoba melakukan pendekatan dengan memanfaatkan kelebihan dari metode *dispatcher* sebagai *full control* yang akan digantikan dengan sebuah proxy untuk menyimpan antrian request dari client. Namun pemilihan server untuk memproses request dilakukan oleh server-server itu sendiri dengan melihat kondisi resource server. Server sendiri yang nantinya akan menentukan apakah dirinya akan memproses request yang ada di antrian atau tidak. Dengan metode seperti ini diharapkan mendapatkan performansi yang lebih baik, terlebih dalam lingkungan hardware yang tidak sama

1.2 Perumusan masalah

Masalah yang akan diselesaikan dalam tugas akhir ini adalah

1. Bagaimana mengaplikasikan teknik yang berbeda dalam *load balancing* web server yaitu *dispatcher receiver pull based* dimana *dispatcher* akan digantikan dengan metode antrian
2. Bagaimana membangun sebuah *load balancing* yang *fair share* sesuai dengan *current load* diantara web server terlebih spesifikasi hardware diantara web server yang berbeda-beda sehingga mempengaruhi *response time* dan *request rate* server dalam memproses request dari client dan akhirnya meningkatkan performansi web server dalam hal *throughput*.

Ruang lingkup yang membatasi tugas akhir ini adalah

1. Metode antrian yang akan digunakan adalah *single central* antrian, yaitu hanya ada satu server antrian yang dijadikan sebagai *proxy* dari distribusi web server
2. Untuk menentukan apakah web server dapat melayani request dari client atau tidak akan digunakan sebuah parameter atau metrik yaitu nilai utilisasi cpu web server kurang dari batas maksimal *threshold* cpu utilisasi yang telah ditentukan sehingga dinyatakan dalam kondisi *health*. Jika utilisasi cpu web server melebihi batas maksimal *threshold* cpu utilisasi maka, web server tidak dapat melakukan pelayanan request dan harus menunggu sampai kondisinya dalam keadaan *health*.
3. Implementasi tugas akhir ini menggunakan sistem operasi linux

1.3 Tujuan

Tujuan dari penelitian Tugas Akhir ini adalah mengimplementasikan metode *dispatcher receiver pull based* dan membandingkan dengan metode *dispatcher load balancing* yang menggunakan algoritma *Round Robin* dan menganalisa performansi server dalam hal *throughput*.

1.4 Metodologi penyelesaian masalah

Dalam penelitian Tugas Akhir ini, metode yang digunakan untuk menyelesaikan Tugas Akhir sebagai berikut :

1. Studi literatur

Mengumpulkan bahan-bahan pustaka sebagai referensi tugas akhir dan mempelajari literatur-literatur yang relevan dengan permasalahan meliputi melakukan studi pustakan dan referensi mengenai konsep dari load balancing, algoritma load balancing dan penerapannya.

2. Pengembangan aplikasi yang meliputi :

1. Analisa dan perancangan

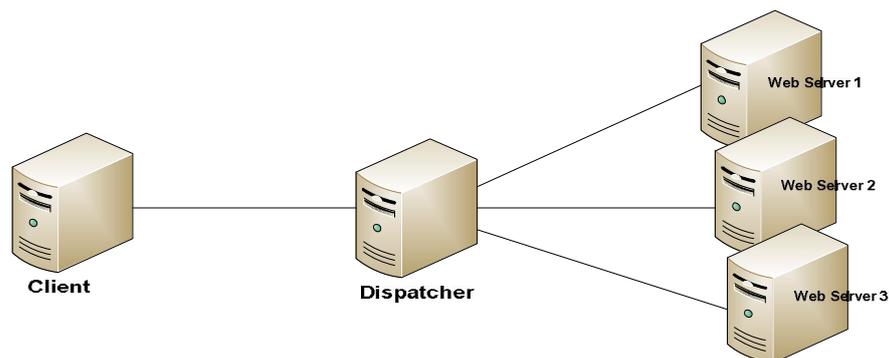
Memodelkan perancangan aplikasi ke dalam suatu diagram berdasarkan analisis dan desain

2. Pengkodean

Mengimplementasikan perancangan menjadi sebuah aplikasi.

3. Pengujian aplikasi

Melakukan pengujian dari sistem yang telah dibangun pada tahap implementasi kemudian menganalisa performansi dengan bantuan beberapa tool dan dibandingkan dengan sistem tradisional. Pengujian menggunakan sistem operasi linux yang terdiri dari beberapa web server dan satu sebagai dispatcher yang merupakan single central antrian. Topologi seperti gambar dibawah ini



4. Analisis hasil

Menganalisa hasil dari percobaan yang akan dibandingkan dengan sistem tradisional load balancing

5. Pembuatan laporan

Mendokumentasikan tahap-tahap yang dilakukan selama penelitian dan melaporkan hasil penelitian dalam tugas akhir ini