

## Abstrak

Meningkatnya penggunaan internet pada saat ini mengakibatkan *stress* pada web server sehingga performansi web server menjadi menurun. Di lain sisi, user mengharapkan *low latency*, *high throughput* dan *high availability*. *Load balancing* merupakan salah satu teknik untuk meningkatkan performansi distribusi web server. *Load balancing* dapat menyediakan *high availability* melalui *redundant* server dan mengurangi *latency* dengan membagi *load* diantara *back end* server.

Banyak algoritma dan teknik untuk mengimplementasikan *load balancing*. Salah satu teknik adalah penerapan *dispatcher* yang difungsikan sebagai *controlling*. Pada umumnya *dispatcher* ini menjadi penentu pemilihan *back end* server yang akan melayani request sesuai algoritma yang diterapkan. *Dispatcher* ini mendorong *request* ke *back end* server. Pada tugas akhir ini, bertujuan untuk mengimplementasikan *load balancing* dengan teknik *dispatcher*, namun *dispatcher* pada tugas akhir ini berfungsi sebagai *central queue* untuk menyimpan request dari client. Keputusan siapa yang akan melayani request ditentukan sendiri oleh *back end* server berdasarkan utilitas *cpu* yang dimiliki.

Sistem *dispatcher* yang dihasilkan kemudian di uji dengan parameter nilai *throughput* yang hasilnya dibandingkan dengan sistem *dispatcher* menggunakan algoritma *round robin*. Hasil pengujian menunjukkan bahwa, *overhead* yang dialami pada sistem *receiver pull based* menyebabkan *response rate* yang dihasilkan tidak maksimal sehingga nilai *throughput* yang didapatkan lebih kecil dibandingkan sistem *dispatcher round robin*.

**Kata kunci:** load balancing, dispatcher, central queue, pull based