

ABSTRAK

Perkembangan aplikasi *website* berkembang dengan pesat. Hal ini membuat kebutuhan penggunaan akses suatu *website* meningkat. Dengan meningkatnya kebutuhan akan penggunaan suatu *website* maka membuat beban kerja *server* yang lebih pada suatu layanan *web server*, *server* menjadi tidak maksimal ketika permintaan dari pengguna meningkat, *server* akan terbebani karena harus melayani permintaan tersebut.

Metode yang dapat membagi trafik ke beberapa *server* sehingga tidak terjadi penumpukan beban pada sebuah *server* disebut *Load Balancing*. Perancangan *Load Balancing* pada penelitian ini bertujuan untuk meningkatkan kinerja pada sistem dan mendistribusikan beban yang tersedia pada *server*. Pertimbangan pembagian beban dapat dilakukan dengan menggunakan informasi perangkat pada *server* yang terdiri dari *CPU*, *memory* dan *disk*. Pada penelitian ini menggunakan algoritma *Dynamic Ratio* dan sebagai pembandingnya menggunakan algoritma *Round Robin*. Pengujian *Load Balancing* tersebut menggunakan *software* yang bernama *Httpperf*. *Httpperf* dapat menampilkan nilai parameter yang dibutuhkan seperti *Throughput*, *Response Time*, *Error* dan *CPU Utilization*.

Dari hasil pengujian menunjukkan dengan menggunakan 3 *server* tidak terjadi *overload* pada *server* pada saat pengujian algoritma *Dynamic Ratio* maupun *Round Robin*. nilai rata-rata pada algoritma *Dynamic Ratio* sebesar 57.83 KB/s dan pada *Round Robin* sebesar 55.27 KB/s, nilai rata-rata *Response Time* pada algoritma *Dynamic Ratio* sebesar 2.64 detik dan *Round Robin* sebesar 2.67 detik, nilai *Error* pada algoritma *Dynamic Ratio* sebesar 0.0% dan *Round Robin* sebesar 0.9% sedangkan nilai *CPU Utilization* pada algoritma *Dynamic Ratio* sebesar 93.5% dan algoritma *Round Robin* sebesar 93.0%. Nilai *Fairness Index* pada algoritma *Dynamic Ratio* tidak mencapai angka 1 sedangkan pada algoritma *Round Robin* dapat mencapai angka 1.

Kata kunci: *Load Balancing*, *Dynamic Ratio*, *Round Robin*.