

Analisis Kinerja Web Server Menggunakan Algoritma Round Robin dan Least Connection

Molavi Arman^{*1}, Novan Wijaya², Hafiz Irsyad³

^{1,2}Manajemen Informatika, AMIK MDP, ³Teknik Informatika, STMIK Global Informatika MDP
^{1,2,3}Jl. Rajawali, no. 14, Palembang, Indonesia

^{*1}molaviarman@mdp.ac.id, ²novan.wijaya@mdp.ac.id, ³hafizirsyad@mdp.ac.id

Abstract-- — In the industrialized world many companies are using web as a form of promotion or online transactions such as e-commerce. It handles millions of hit visits on the web server. There are many things that can cause the failure hit. One of them is the number of requests or transactions to the web server that is not able to be handled or the slowness of the response, which is very dangerous and detrimental for the company. The single web server with extremely expensive brands is a way to face the constraints overload, but only certain companies are able to have it. The issues faced, how to use some of the PCs as a web server with a reasonable economic value, could be empowered by implementing network load balancing technology. Network load balancing is a technology solution that is used, and expected to be able to handle the simultaneous load of web server with small output failure. Web server performance measurement at the round robin algorithm and least connection include parameters against components such as throughput, request loss, response time and cpu utilization, from the result of the measurement, it will be seen which is the best algorithm to use.

Keywords— web server, Network Load Balancing, Round Robin, Least Connection, throughput, request loss, respon time, cpu utilization.

I. PENDAHULUAN

Perkembangan internet di tanah air cukup mengalami peningkatan yang sangat signifikan, dari informasi KemKomInfo (Kementrian Komunikasi Dan Informatika Republik Indonesia) pada tahun 2014 pengguna internet di Indonesia telah mencapai 82 juta pengguna [1]. Peningkatan pengguna internet diiringi meningkatnya banyak konten yang biasa disajikan kedalam internet berupa teks, gambar, dan video streaming. Situs web yang memiliki traffic tinggi dikarenakan banyak pengguna internet mengaksesnya dengan kemampuan yang dimiliki web tersebut biasa dikatakan mempunyai kemampuan infrastruktur sumber daya yang baik.

Dalam dunia industri banyak perusahaan perusahaan menggunakan web sebagai bentuk promosi ataupun transaksi online seperti web-web e-commerce, web-web tersebut menangani jutaan hit pada web server mereka. Banyak hal yang menyebabkan kegagalan hit salah satunya adalah banyaknya permintaan ataupun transaksi ke web server yang tak mampu ditangani, hal ini sangat berbahaya dan merugikan perusahaan. Single Web Server dengan brand ternama

sangatlah mahal adalah sebuah cara dalam menghadapi kendala overload, akan tetapi hanya perusahaan tertentu yang mampu memilikinya. Masalah yang dihadapi bagaimana menggunakan beberapa PC sebagai web server dengan nilai ekonomi terjangkau mampu diberdayakan dengan menerapkan teknologi network load balancing. Teknologi network load balancing adalah solusi yang akan digunakan, diharapkan akan mampu menanggapi beban simultan web server yang besar dengan kegagalan output yang kecil.

Network Load Balancing adalah solusi sistem yang dapat membantu mengurangi beban simultan yang besar dengan memperkecil adanya kegagalan. Salah satu jenis sistem network load balancing yang bekerja pada layer empat yaitu Linux Virtual Server (LVS). Pada penelitian ini akan membandingkan dua kinerja algoritma round robin dan least connection sebagai load balancing.

Perfomansi sistem web server cluster yang akan dibangun, dengan membandingkan kinerja web server dengan algoritma round robin dan least connection berdasarkan parameter yang telah ditentukan guna mendapatkan hasil dari algoritma tersebut yang mempunyai perfoma paling baik.

II. DASAR TEORI

A. Web Cluster

Web Cluster adalah sekumpulan web server yang berkomunikasi dengan klien (browser) , menggunakan protokol yang sangat sederhana yang disebut hypertext transfer protocol . Klien membentuk koneksi pada port web server yaitu port 80, yang dibuka selama ada sesi. Pada web server non-cluster, hanya ada satu web server melayani permintaan yang ditujukan kepada satu host atau internet protokol (IP) address. Pada web server berbasis cluster, beberapa web server kooperatif melayani permintaan yang ditujukan kepada hostname atau IP address pada situs tersebut. Secara umum, semua web server ini menyediakan konten yang sama Konten tersebut baik direplikasi pada disk lokal setiap mesin atau bersama pada sistem file jaringan. Setiap permintaan ditujukan pada hostname atau IP address akan didistribusikan, berdasarkan beban sharing algoritma, untuk satu back-end server dalam cluster dan dilayani oleh server. Distribusi tersebut

direalisasikan dengan baik oleh modul perangkat lunak yang berjalan pada sistem operasi atau perangkat keras dengan tujuan terhubung ke jaringan.[2].

B. Algoritma Round Robin dan Algoritma Least Connection

Algoritma penjadwalan round-robin memberikan pekerjaan kepada semua peserta secara round robin. Misalnya, ada tiga server (server A, B dan C) dalam penjadwalan round robin, permintaan pertama akan pergi ke server A, permintaan kedua akan pergi ke server B, permintaan ketiga akan pergi ke server C, dan permintaan keempat akan pergi ke server A, kemudian ulangi dengan cara round robin. Dalam bentuk matematis, itu memilih server oleh $i = (i + 1) \text{ mod } n$ setiap kali, di mana n adalah jumlah server [3].

Dalam implementasinya IP Virtual Server kondisi tambahan yang diperkenalkan dalam round-robin penjadwalan untuk memeriksa apakah server tersedia atau tidak, karena server dapat diambil dari layanan baik untuk menutupi kegagalan atau sistem pemeliharaan server. Prosedur formal round robin penjadwalan adalah sebagai berikut:

Tabel 1 Algoritma Round Robin

```
Seandainya bahwa ada n server di set S = {S0, S1, ..., Sn-1}, mana n > 0;
"i" menunjukkan server dipilih terakhir kali, dan "i" diinisialisasi dengan -1;
j = i;
do {
    j = (j + 1) mod n;
    if (Available(Sj)) {
        i = j;
        return Sj;
    }
} while (j != i);
return NULL;
```

Algoritma penjadwalan least connection mengarahkan koneksi jaringan ke server dengan sedikitnya jumlah koneksi yang ada pada sistem. Ini adalah salah satu algoritma penjadwalan dinamis; karena itu perlu menghitung jumlah koneksi untuk setiap server secara dinamis untuk memperkirakan bebannya. Penyeimbang beban mencatat jumlah koneksi dari setiap server, meningkatkan jumlah koneksi dari server ketika koneksi baru dikirim untuk itu, dan menurunkan jumlah koneksi dari server ketika finish sambungan atau timeout [4].

Dalam implementasinya IP Virtual Server kondisi tambahan yang diperkenalkan pada setidaknya koneksi penjadwalan untuk memeriksa apakah server tersedia atau tidak, karena server dapat diambil dari layanan baik untuk menutupi kegagalan atau sistem pemeliharaan server. Server beban berat nol menunjukkan bahwa server ini tidak tersedia. Prosedur formal Least Connection penjadwalan adalah sebagai berikut:

Tabel 2 Algoritma Least Connection

```
Mengandaikan bahwa ada server set S = {S0, S1, ..., Sn-1},
W (Si) adalah berat server Si,
C (Si) adalah jumlah koneksi saat server Si,

for (m = 0; m < n; m++) {
    if (W(Sm) > 0) {
        for (i = m+1; i < n; i++) {
            if (W(Si) <= 0)
                continue;
            if (C(Si) < C(Sm))
                m = i;
        }
        return Sm;
    }
}
return NULL;
```

Untuk cluster IPVS yang memiliki koleksi server dengan kinerja yang sama, penjadwalan koneksi yang paling baik untuk kelancaran distribusi ketika beban permintaan bervariasi banyak. Penyeimbang beban akan mengarahkan permintaan ke server sebenarnya dengan koneksi paling sedikit.

C. Load Balancing

Load Balancing adalah proses pendistribusian beban terhadap sebuah servis yang ada pada sekumpulan server atau perangkat jaringan. Ketika ada permintaan dari pemakai ketika banyak permintaan dari pemakai maka server tersebut akan terbebani karena harus melakukan proses pelayanan terhadap permintaan pemakai. Solusi yang cukup bermanfaat adalah dengan membagi-bagi beban yang datang ke beberapa server, jadi tidak berpusat ke salah satu perangkat jaringan saja. Teknologi itulah yang disebut Teknologi Load Balancing. Teknologi Load Balancing dapat diperoleh keuntungan seperti menjamin reabilitas servis, availabilitas dan skalabilitas suatu jaringan. [5]

D. Linux Virtual Server (LVS)

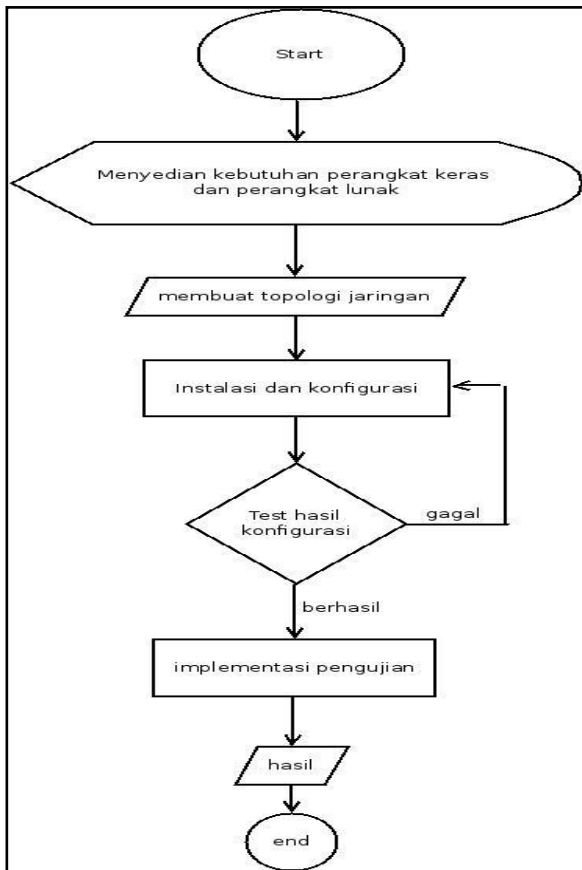
Linux Virtual Server adalah perangkat lunak yang mengarahkan koneksi jaringan ke beberapa server yang berbagi beban kerja mereka, yang dapat digunakan untuk membangun layanan yang sangat scalable dan sangat tersedia. Prototype dari Linux Virtual Server telah digunakan untuk membangun banyak situs beban berat di Internet, seperti Linux Portal www.linux.com, sourceforge.net dan Inggris National JANET Layanan Web Cache. Linux Virtual Server mengarahkan koneksi jaringan ke server yang berbeda sesuai dengan algoritma penjadwalan dan membuat layanan paralel dari cluster muncul sebagai layanan virtual pada satu alamat IP. Aplikasi client berinteraksi dengan cluster seolah-olah server tunggal. Klien tidak terpengaruh oleh interaksi dengan cluster dan tidak perlu modifikasi. Skalabilitas dicapai dengan transparan menambahkan atau menghapus node di cluster. Ketersediaan tinggi disediakan dengan mendeteksi node atau daemon kegagalan dan konfigurasi ulang sistem yang tepat. [6].

III. PEMBAHASAN

A. Alur Rancang Pengujian

Alur rancang pengujian dibuat didalam beberapa tahapan guna memudahkan dalam proses pengujian load balancing.

- a. Tahap menyediakan kebutuhan perangkat keras dan perangkat lunak untuk pengujian berupa 3 komputer, 1 switch, 1 komputer sebagai balancer, 1 komputer sebagai klien.
- b. Membuat topologi jaringan untuk pengujian.
- c. Melakukan instalasi dan konfigurasi.
- d. Melakukan test konfigurasi
- e. Melakukan implementasi pengujian / proses pengujian.



Gambar 1. Alur Rancang Pengujian

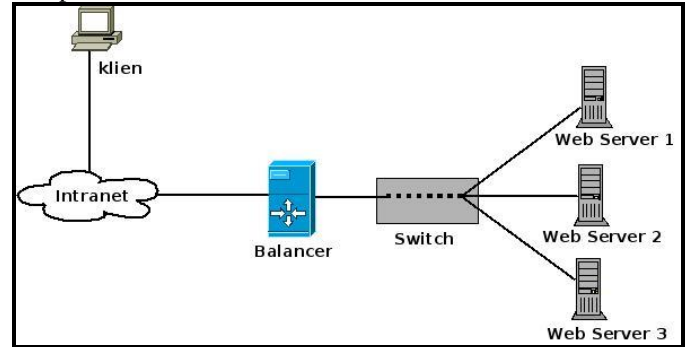
B. Perancangan Konfigurasi Sistem

Untuk menganalisis algoritma scheduling di linux virtual server (LVS) dilakukan perancangan, implementasi dan pengukuran beberapa parameter meliputi implementasi dan pengukuran beberapa parameter meliputi throughput, request loss, response time, dan cpu utilization [7].

Model LVS topologi dirancang seperti dibawah ini terlihat pada gambar 3.2. Pada topologi LVS ini terdapat 2 komponen yaitu :

a. Balancer sebagai router dan tiga web server. Komputer klien akan menganalogikan bahwa balancer / router seolah-olah sebagai server tunggal atau satu web server.

b. Web server, yang akan melayani permintaan komputer klien.



Gambar 2. Topologi Pengujian

Sistem LVS yang dirancang menggunakan metode Network Address Translation (NAT) dengan beban pada tiga buah web server diseimbangkan oleh load balancer/director. Terdapat sebuah director aktif. Balancer / Router dan tiga buah web server dihubungkan oleh switch dan memberikan layanan sama protokol http. Berikut keterangan tabel IP Address pada topologi LVS

Tabel 3 IP Address pada topologi LVS

Web Server 1	192.100.100.8
Web Server 2	192.100.100.9
Web Server 3	192.100.100.10
<i>Balancer</i>	192.100.100.1
	192.168.30.9
<i>Client</i>	192.168.30.2

C. Komponen Sistem

1) *Komponen Hardware*

Sistem LVS dilakukan uji dengan cara melakukan request / permintaan secara simultan dengan jumlah yang banyak. Untuk melakukan pengujian tersebut memerlukan jumlah klient yang banyak pula. Maka digunakan request generator agar cukup dengan satu buah komputer seolah dapat menghasilkan request yang dihasilkan oleh banyak klient secara simultan.

Spesifikasi komputer client:

a. Processor : AMD Bulldozer CPU 3,5 GHz

b. RAM : 2 GB

Spesifikasi komputer balancer:

a.Processor : : AMD Bulldozer CPU 3,5 GHz

b.RAM : 2 GB

c.Ethernet - 10/100 Mbps

2) *Komponen Software*

Perangkat software yang digunakan pada pengujian ini adalah sebagai berikut :

1. Sistem Operasi , Komputer sebagai router / balancer menggunakan sistem operasi Linux Slackware64-14.2.
2. Kernel Linux menggunakan versi 4.4.14. Kernel disini adalah sistem yang mengenali semua perangkat keras pada komputer.
3. Web Server, menggunakan apache versi 2.4.20.
4. Ipvssadm, IP Virtual Server Administration (Ipsadm) digunakan untuk mengatur kerja router / balance menambahkan layanan-layanan yang dapat diberikan, memilih algoritma penjadwalan yang digunakan, dan meneruskan request kepada web server yang sedang aktif. Ipsadm menggunakan versi 1.25.
5. Httpperf, sebagai aplikasi untuk melakukan pengujian membangkitkan request dalam satu waktu dengan secara simultan. Httpperf menggunakan versi 0.9.0.

Berikut adalah hasil dari workload httpperf :

Tabel 3.4 hasil dari workload pada algoritma round robin

```
httpperf --hog --timeout=5 --client=0/1 --server=192.168.30.9 --port=80 --uri=/wordpress --rate=5000 --send-buffer=4096 --recv-buffer=16384 --num-conns=20000 --num-calls=1
Maximum connect burst length: 58

Total: connections 13990 requests 13872 replies 13836 test-duration 6.712 s

Connection rate: 2084.2 conn/s (0.5 ms/conn, <=4094 concurrent connections)
Connection time [ms]: min 1.9 avg 1134.8 max 5061.4 median 774.5 stddev 935.1
Connection time [ms]: connect 549.3
Connection length [replies/conn]: 1.000

Request rate: 2066.6 req/s (0.5 ms/req)
Request size [B]: 74.0

Reply rate [replies/s]: min 2493.3 avg 2493.3 max 2493.3 stddev 0.0 (1 samples)
Reply time [ms]: response 584.7 transfer 0.0
Reply size [B]: header 220.0 content 238.0 footer 0.0 (total 458.0)
Reply status: 1xx=0 2xx=0 3xx=13836 4xx=0 5xx=0

CPU time [s]: user 0.27 system 6.38 (user 4.0% system 95.1% total 99.1%)
Net I/O: 1071.3 KB/s (8.8*10^6 bps)

Errors: total 6164 client-timo 154 socket-timo 0 connrefused 0 connreset 0
Errors: fd_unavail 6010 addrunavail 0 ftab-full 0 other 0
```

Tabel 3.5 hasil dari workload pada algoritma least connection.

```
httpperf --hog --timeout=5 --client=0/1 --server=192.168.30.9 --port=80 --uri=/wordpress --rate=5000 --send-buffer=4096 --recv-buffer=16384 --num-conns=20000 --num-calls=1
Maximum connect burst length: 28

Total: connections 12857 requests 12833 replies 12427 test-duration 6.679 s

Connection rate: 1925.1 conn/s (0.5 ms/conn, <=4094 concurrent connections)
Connection time [ms]: min 2.5 avg 1199.1 max 5199.6 median 934.5 stddev 1006.3
Connection time [ms]: connect 440.6
Connection length [replies/conn]: 1.000

Request rate: 1921.5 req/s (0.5 ms/req)
Request size [B]: 74.0

Reply rate [replies/s]: min 2205.3 avg 2205.3 max 2205.3 stddev 0.0 (1 samples)
Reply time [ms]: response 753.3 transfer 0.0
Reply size [B]: header 220.0 content 238.0 footer 0.0 (total 458.0)
Reply status: 1xx=0 2xx=0 3xx=12427 4xx=0 5xx=0

CPU time [s]: user 0.30 system 6.36 (user 4.4% system 95.2% total 99.6%)
Net I/O: 971.1 KB/s (8.0*10^6 bps)

Errors: total 7573 client-timo 430 socket-timo 0 connrefused 0 connreset 0
Errors: fd_unavail 7143 addrunavail 0 ftab-full 0 other 0
```

Berikut adalah hasil pengujian algoritma round robin dan least connection pada parameter throughput.

Tabel 3.6 Hasil throughput pengujian

Connection /s	Throughput (KB/s)	
	rr	lc
5000	1071.3	1044.8
10000	939.6	935.7
15000	1141.2	1042.0

Berikut adalah hasil pengujian algoritma round robin dan least connection pada parameter response time.

D. *Proses Pengujian*

Berdasarkan parameter pengujian terdapat bagian informasi penting sebagai berikut :

- a. Time Response didapatkan dari Reply Section.
- b. Throughput berdasarkan manual httpperf dapat diambil dari Miscellaneous Section, yaitu Net I/O yang merupakan rata-rata throughput jaringan yang mempunyai satuan kilobytes per detik dan megabits per detik.
- c. Request Lost didapatkan dari Error Section pada connrefused dan connreset.

1) *Pengujian Algoritma Round Robin*

Contoh pengujian dengan parameter perintah sebagai berikut :

Tabel 3.2 Perintah pengujian untuk algoritma round robin

```
httpperf --hog --server 192.168.30.9 --num-conn 20000 --ra 5000 --timeout 5
```

Perintah diatas akan menghasilkan 20000 dengan 5000 koneksi per detik. Pengujian disini menggunakan koneksi sebanyak, 5000,10000 dan 15000. Contoh penggunaan aplikasi Ipvadmin dalam pengujian algoritma round robin dan least connection adalah sebagai berikut.

Tabel 3.3 Contoh perintah Ipsvadmin

```
ipvsadm -A -t 192.168.30.9:80 -s rr
ipvsadm -a -t 192.168.30.9:80 -r 192.100.100.8:80 -m
ipvsadm -a -t 192.168.30.9:80 -r 192.100.100.9:80 -m
ipvsadm -a -t 192.168.30.9:80 -r 192.100.100.10:80 -m
ipvsadm -A -t 192.168.30.9:80 -s lc
```

Tabel 3.7 Hasil response time pengujian

Connection /s	Response Time (ms)	
	rr	lc
5000	584	753.3
10000	635.9	788.3
15000	495	647.6

Throughput dengan pengujian menggunakan algoritma load balancing round robin dan least connection bernilai 1071.3 KB/s dan 1044.8 KB/s, ini menandakan bahwa pengujian dengan menggunakan algoritma load balancing dengan algoritman load balancing round robin lebih baik dibandingkan dengan menggunakan least connection.

Respon time pada pengujian menggunakan algoritma load balancing round robin dan algoritma load balancing least connection bernilai 584 ms dan 753.3 ms ini menandakan bahwa pengujian dengan menggunakan load balancing dengan algoritma round robin lebih baik dibandingkan dengan menggunakan algoritma load balancing least connection.

Untuk CPU utilization dan Error tidak dijadikan parameter penentuan algoritma terbaik. Hasil CPU Utilization yang didapat dari pengujian adalah CPU utilization dari client machine terhadap kinerja saat menghidupkan permintaan secara simultan, jadi tidak berhubungan atau berkaitan dengan

kinerja dari suatu algoritma tersebut. Sedangkan untuk Error atau kesalahan yang dihasilkan dari pengujian yang dilakukan merupakan error / kesalahan pada client, error ini adalah penolakan dari server tidak ditemukan dan selalu bernilai 0 (lihat hasil uji), maka dari itu parameter error juga tidak layak diambil sebagai parameter yang menentukan algoritma terbaik. diambil sebagai parameter yang menentukan algoritma terbaik.

DAFTAR PUSTAKA

- [1] http://kominfo.go.id/index.php/content/detail/3980/Kemkominfo%3A+Pengguna+Internet+di+Indonesia+Capai+82+Juta/0/berita_satker#.VmZR23a_CsM (8 Desember 2015 10:43)
- [2] Xuehong, Trevor Schroeder, Steven Goddard, Byrav Ramamurthy Highly Available and Scalable Cluster Base Web Server, IEEE, Vol.8, 2000.
- [3] http://kb.linuxvirtualserver.org/wiki/RoundRobin_Scheduling (8 Desember 2015 11:50)
- [4] http://kb.linuxvirtualserver.org/wiki/Least-Connection_Scheduling (8 Desember 2015 13:37).
- [5] Teknologi Load Balancing Untuk Mengatasi Beban Server, Iwan Rijayana, Seminar Nasional Aplikasi Teknologi Informasi, 18 Juni 2005. Yogyakarta.
- [6] Implementasi Load Balancing Web Server Menggunakan Metode LVS-NAT, Jefry Alvonsius Rabu, Joko Purwadi, Willy S. Rahajo. Jurnal Informatika, Vol 8, No 2 (2012).
- [7] Metode Penjamin Kualitas Pada Teknologi Cloud Computing, Sinung Suakanto, Telematika, Vol 7, No 1 (2011).