

# MENGATASI *OVERLOAD* MENGGUNAKAN *LINUX VIRTUAL SERVER* SEBAGAI *LOAD BALANCING* PADA *SERVER E-LEARNING* UNIVERSITAS BINA DARMA

Suryayusra

Dosen Universitas Bina Darma

Jalan Jenderal Ahmad Yani No.12, Palembang

Pos-el : suryayusra@mail.binadarma.ac.id, suryayusra@yahoo.co.id

---

**Abstract:** Provision of information services on a system relies heavily on existing infrastructure. At Bina Darma University server has the functions of the application and hold a very important role in carrying out teaching and learning activities. One function of these servers are running the e-learning applications to support teaching and learning or supplements that are given regardless of the regular meeting (in class). In daily operations, (MIS-CUTS Bina Darma) states that e-learning server must serve 4437 users with a variety of activities. Many activities that must be completed within a time resulted in Apache connection flooded with requests that exceed the ability of so many processes that are not resolved. LVS can be applied to generate a load balancing. By providing a computer unit that acts as a director and two or more computers that will do the (real server) will improve performance so that availability can be generated by the method of load balancing.

**Keywords:** Load Balancing, Linux Virtual Server, Network Address Translation, E-learning

**Abstrak:** Penyediaan layanan informasi pada sebuah sistem sangat bergantung pada infrastruktur yang ada. Pada Universitas Bina Darma server memiliki fungsi-fungsi aplikasi dan memegang peranan yang sangat penting dalam menjalankan kegiatan belajar mengajar. Salah satu fungsi server tersebut adalah menjalankan aplikasi e-learning sebagai penunjang belajar mengajar atau suplemen yang diberikan terlepas dari pertemuan reguler (di dalam kelas). Dalam operasional sehari-hari, (UPT-SIM Bina Darma) menyatakan bahwa server e-learning harus melayani 4.437 pengguna dengan bermacam kegiatan. Banyaknya kegiatan yang harus diselesaikan dalam satu waktu mengakibatkan koneksi Apache dibanjiri permintaan yang melebihi kemampuan sehingga banyak proses yang tidak terselesaikan. LVS dapat diterapkan untuk menghasilkan sebuah load balancing. Dengan menyediakan satu unit komputer yang bertindak sebagai director dan dua atau lebih komputer yang akan melakukan proses (real server) akan meningkatkan kinerja sehingga availability dapat dibangkitkan dengan metode load balancing.

**Kata kunci:** Load Balancing, Linux Virtual Server, Network Address Translation, E-learning

---

## 1. PENDAHULUAN

Sejalan dengan visi dan misi Kementerian Pendidikan dan Kebudayaan Republik Indonesia yaitu terselenggaranya layanan pendidikan dan kebudayaan nasional untuk membentuk insan Indonesia yang cerdas dan berkarakter kuat serta meningkatkan ketersediaan dan memperluas jangkauan informasi dalam dunia pendidikan. (<http://www.kemdiknas.go.id>). Media *e-learning*

merupakan suatu jenis belajar mengajar yang memungkinkan tersampainya bahan ajar ke siswa dengan menggunakan media *internet*, *intranet* atau media jaringan komputer lain (Romi, 2008).

Pada Universitas Bina Darma *e-learning* diterapkan sebagai suplemen penunjang kegiatan belajar di kelas (reguler). Dalam operasional setiap hari, *Network Operation Center* (NOC UBD) menyatakan ”bahwa server *e-learning*

harus melayani 4.437 pengguna dengan bermacam kegiatan seperti unggah dan unduh materi, diskusi (forum), mengobrol (*chatting*) dan ujian secara langsung". (<http://binadarma.ac.id/content/120/0/miscuts.html>). Layanan *e-learning* tersebut berbasis web dan menggunakan *Apache* sebagai *web server*. Menurut (Ashari, 2008) salah satu perangkat lunak yang dipergunakan secara luas pada sistem operasi Linux adalah *Apache web server*. Fitur *Apache* mendukung HTTP 1.1, CGI atau FastCGI, Virtual Host, dan *Secure Socket Layer* (Aidil, 2008).

Masalah yang muncul pada *e-learning* Universitas Bina Darma adalah *overload* yang ditunjukkan dari *log server e-learning* "Jun2:12:09-SERVERelearning kernel: [770645.647316] possible SYN flooding on port 80. Sending cookies". Secara standar *Apache* tidak mempunyai kemampuan untuk mengatur *load* seperti pada IIS/NT (Kresno, 2002).

Tindakan penyelamatan adalah dengan membagi proses menggunakan *Linux Virtual Server* sebagai penyeimbang beban. (<http://linuxvirtualserver.org>). *Load balancing* merupakan cara untuk mengatasi masalah dengan membagi beban *e-learning* ke beberapa komputer server. Sekumpulan komputer yang terhubung di dalam suatu jaringan komputer dan bekerja secara bersama kemudian saling berhubungan satu sama lain untuk mendukung suatu kerja yang biasa ditangani oleh sebuah komputer tunggal disebut *cluster* (Ferrianto, 2002). Beban server yang tinggi merupakan masalah yang kompleks dimana hanya dapat

terselesaikan oleh sebuah komputer super yang mahal harganya (Heriyadi, 2002).

## 2. METODOLOGI PENELITIAN

Metodologi yang digunakan adalah menggunakan metode penelitian tindakan (*action research*). Tahapan penelitian yang merupakan siklus dari *action research* ini yaitu: 1) melakukan diagnosa dengan melakukan identifikasi masalah pokok yang ada pada objek penelitian; 2) membuat rencana tindakan yaitu memahami pokok masalah yang ditemukan dan menyusun rencana tindakan yang tepat; 3) melakukan tindakan disertai dengan implementasi rencana yang telah dibuat dan mengamati kinerja *load balancing*; 4) melakukan evaluasi hasil temuan setelah proses implementasi; 5) pembelajaran yaitu mengulas tahapan yang telah dilakukan dan mempelajari prinsip kerja *load balancing*.

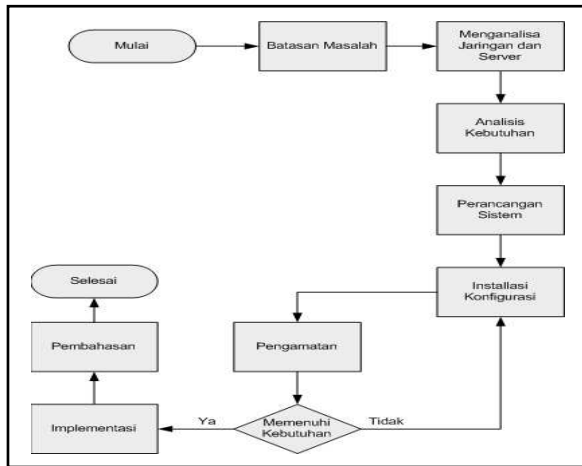
### 2.1 Lokasi Penelitian

Penelitian ini dilakukan pada bulan Januari sampai dengan bulan Desember 2011 bertempat di Unit Pelayanan Teknis-Sistem Informasi Manajemen (UPT-SIM) Universitas Bina Darma.

### 2.2 Kerangka Penelitian

Kerangka penelitian yang dituangkan dalam diagram alir dibawah ini menggambarkan proses penelitian yang akan ditempuh sekaligus

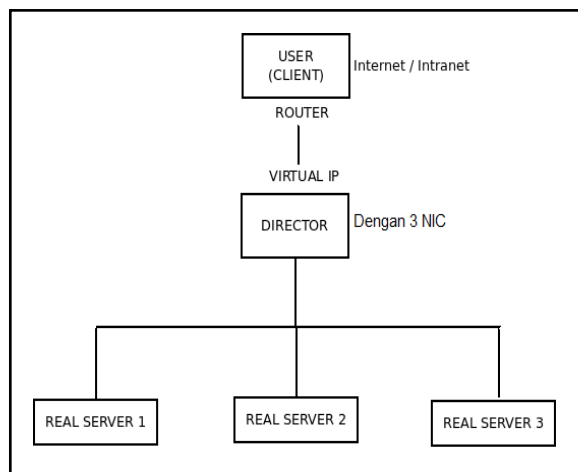
menggambarkan penelitian secara keseluruhan. Diagram alir ini memperlihatkan tahapan-tahapan proses penulisan yang akan dilakukan dari tahap awal sampai akhir.



**Gambar 1. Kerangka Penelitian**

### 2.3 Desain Penelitian

Berisi perancangan (desain) dari perangkat keras maupun perangkat lunak (*software*) yang akan digunakan dalam melakukan simulasi LVS NAT (*Linux Virtual Server Network Address Translation*), meliputi penentuan perangkat keras (*hardware*) digunakan, topologi yang akan digunakan, algoritma penjadwalan pada *director*, dan pengujian terhadap LVS NAT.



**Gambar 2. Skema LVS NAT**

### 2.4 Desain Proses Analisis

Tujuan dalam analisa kebutuhan ini adalah untuk mendapatkan informasi tentang apa yang dibutuhkan dalam perancangan berdasarkan pada aspek kebutuhan pengguna dan pengelola yaitu sebagai berikut :

- 1) Kebutuhan Pengguna: Kemudahan dan ketersediaan konektivitas jaringan dan modul *e-learning* setiap saat baik dari *intranet* maupun *internet*. Sumber daya *e-learning* diperuntukkan mahasiswa untuk mengunduh dan mengunggah konten berupa materi kuliah dimulai dari ukuran 1 MB hingga 32 MB. Kegiatan tersebut dilakukan menggunakan aplikasi *browser* pada masing-masing komputer mahasiswa. Saat melakukan koneksi untuk melakukan unduh dan unggah *e-learning* harus konsisten dengan *session* yang masih terbuka. Begitu juga dengan kegiatan mahasiswa dan dosen seperti mengerjakan kuis dan menjawab soal-soal ujian secara langsung maka *session* tidak boleh terputus. Apabila *session* terputus maka *history* dari semua jawaban dan pertanyaan terhapus. Menjaga konsistensi layanan HTTP merupakan masalah utama yang harus terselesaikan dari tingkat pengguna.
- 2) Kebutuhan Pengelola: Beberapa analisa tentang kebutuhan yang diperlu dicapai terhadap pihak pengelola yaitu : (a) keseimbangan antara beban *real server* yang menjalankan layanan HTTP dengan menjalankan layanan *Apache* secara konsisten sehingga *session* yang diberikan pada pengguna melalui *e-learning* dapat

terus terjaga keberadaannya; (b) ketersediaan *database* MySQL dalam merespon dua unit *real server* setiap ada konektivitas dari layanan HTTP secara berkala dan mampu menyajikan informasi terbaru dari setiap kegiatan pengguna dalam setiap *session* yang ditanggapi oleh koneksi PHP; (c) kemampuan *director* dalam membagi *request* yang datang kemudian mendistribusikannya kepada *real server* secara merata untuk membagi beban berdasarkan penjadwalan yang telah dirancang sebelumnya; (d) menyediakan data sharing yang akan digunakan *real server* untuk melayani pengguna dalam kegiatan unduh dan unggah dalam ukuran yang besar.

## 2.5 Variabel dan Data Penelitian

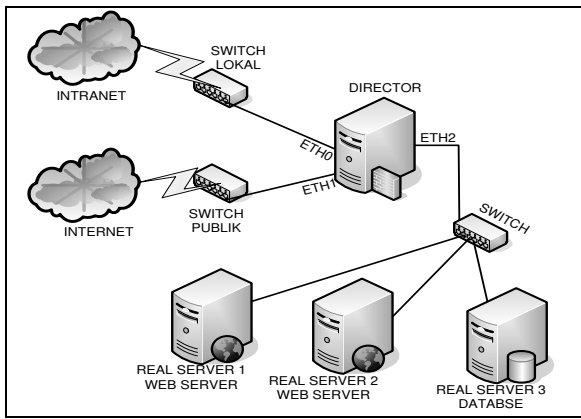
Dalam penelitian ini variabel dan data yang digunakan untuk kemudian diolah menjadi sebuah acuan adalah: (a) jumlah akses yang dapat diselesaikan pada satu waktu dalam satuan detik; (b) kecepatan dalam menyelesaikan proses dalam satuan detik; (c) jumlah memori yang digunakan pada setiap proses dalam satuan *Kilo byte*; (d) waktu ketahanan *server* saat menyediakan layanan dalam satuan detik.

## 3. PEMBAHASAN

### 3.1. Topologi *E-learning* LVS NAT

Perancangan LVS NAT didasarkan pada aspek kebutuhan serta permasalahan yang sering terjadi secara berulang-ulang. Ketidak mampuan sebuah layanan dalam menyelesaikan proses yang sedang dilakukan dan mengabaikan permintaan yang datang mengakibatkan tingkat ketersediaan yang rendah. LVS NAT dirancang dengan beberapa komponen pendukung sehingga dianggap mampu untuk menyelesaikan masalah dengan topologi yang dirancang berlapis.

Topologi LVS NAT lebih mirip dengan sebuah topologi LAN yang tertutup kemudian berhubungan dengan jaringan luar dengan perantara yang disebut *director*. *Real server* bekerja secara tertutup dan mengerjakan tugas-tugas pemroses berdasarkan *request* yang diberikan oleh *director*. Sebuah *director* mengumpulkan semua *request* yang datang kemudian mendistribusikannya ke *real server* berdasarkan algoritma penjadwalan yang diterapkan untuk melakukan distribusi *request* yang datang. Pada algoritma penjadwalan akan disebutkan atau dideklarasikan *port* yang dibuka untuk kemudian diteruskan. Tujuan dari mendeklarasikan *port* pada *director* agar permintaan yang datang lebih fokus untuk diteruskan ke *real server*. Topologi serta mekanisme kerja ditunjukkan pada gambar di bawah ini.



**Gambar 3. Topologi LVS NAT**

### 3.2 Karakteristik Beban *E-learning* Universitas Bina Darma

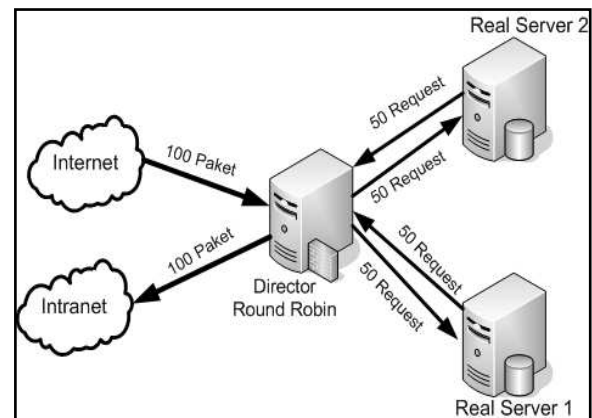
Dalam operasional sehari-hari banyak kegiatan yang harus dilayani server *e-learning* Universitas Bina Darma. Dosen selaku bagian dari pemain utama dalam operasionalnya menggunakan *e-learning* sebagai media untuk meletakkan materi kuliah dalam berbagai tipe berkas, diantaranya doc, pdf dan ppt, png, jpg, mp3, mp4, dan flv. Kegiatan tersebut dinamakan dengan unggah materi dan tidak terlalu membebani kinerja server karena tidak dilakukan secara bersamaan oleh para dosen. Dari hasil pengamatan peneliti melihat beban yang berat terjadi jika para mahasiswa mengerjakan soal ujian secara *online*. Sebagai contoh beban yang terlihat apabila seorang dosen melakukan ujian tengah semester secara *online* dengan memberikan 50 pertanyaan yang harus dijawab 30 mahasiswa. Beban akan memuncak ketika para mahasiswa mulai mengerjakan tugas secara bersamaan karena server harus mempertahankan dan membuka *session* setiap kali mahasiswa menjawab pertanyaan secara langsung.

Beban puncak terjadi pada minggu-minggu pelaksanaan ujian kuis, ujian tengah

semester dan ujian akhir semester. Para dosen menggunakan fasilitas kuis serta memberikan soal-soal ujian dalam bentuk pilihan berganda atau *essay*. Bayangkan jika kegiatan ini dilakukan pada waktu yang bersamaan dan diproses secara langsung untuk memperlihatkan nilai hasil ujian. Jumlah *session* yang harus dipertahankan oleh Apache terlampaui dengan banyaknya jumlah koneksi yang harus dibuka secara bersamaan.

### 3.3 Mekanisme Pembagian Beban

*Director* berfungsi sebagai perantara antara jaringan luar dan jaringan dalam, dimana paket yang datang menuju *e-learning* Bina Darma akan dibagi berdasarkan penjadwalan yang telah dideklarasikan. Alur paket yang datang dari *intranet* maupun *internet* secara konsisten dibagi menjadi dua arah menuju *real server*.



**Gambar 4. Pembagian Paket LVS NAT**

### 3.4 Implementasi Dan Konfigurasi

Pada penelitian ini digunakan tiga kartu jaringan yang digunakan pada mesin *director*. Setiap kartu jaringan terkoneksi dengan *network*

yang berbeda berdasarkan konfigurasi yang dibuat pada `/etc/network/interfaces`.

```

root@director-server: ~
File Edit View Terminal Help
GNU nano 2.2.2 File: ...network/interfaces
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
address 192.168.1.1
netmask 255.255.255.0
gateway 192.168.1.1

auto eth1
iface eth1 inet static
address 222.124.3.141
netmask 255.255.255.248
gateway 222.124.3.137

auto eth2
iface eth2 inet static
address 10.237.2.9
netmask 255.255.255.0
gateway 10.237.2.1
  
```

**Gambar 5. IP Address Director**

Langkah konfigurasi *director* selanjutnya adalah melakukan instalasi *tools ipvsadm* yang akan digunakan untuk mendeklarasikan penjadwalan paket yang datang untuk kemudian diteruskan kepada *real server* yang berada dalam lingkungan LVS NAT. Selain itu *director* juga akan dikonfigurasi menjadi sebuah *gateway* dengan melakukan editing pada berkas `sysctl.conf` dilanjutkan dengan membuat perintah *iptables* untuk melakukan NAT. Merujuk pada topologi yang telah di rancang sebelumnya maka *real server 1* dan *real server 2* berfungsi sebagai *web server*. Pada dasarnya konfigurasi dan tools yang ada pada *real server 1* dan *real server 2* adalah sama. Perbedaan dari kedua *real server* ini hanya pada konfigurasi *IP Address*. Selanjutnya melakukan instalasi paket LAMP dengan menjalankan perintah `sudo apt-get install Apache2 PHP5-MySQL libApache2-mod-PHP5` dan menyalin *CMS moodle* pada folder `/var/www` kemudian melakukan perubahan pada berkas `config.PHP`. *Real server 3* merupakan komputer yang difungsikan sebagai *database* dan *file server* yang menggunakan *IP Address* 192.168.1.4.

*Database* yang digunakan adalah MySQL versi 5 yang dianggap cukup tangguh selain kelebihanannya sebagai perangkat lunak dengan kode terbuka (*open source*).

```

root@elearning.binadarr
File Edit View Terminal Help
GNU nano File: /etc/network/interfaces

auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
address 192.168.1.2
netmask 255.255.255.0
gateway 192.168.1.1
  
```

```

root@elearning.binadarr
File Edit View Terminal Help
GNU nano File: /etc/network/interfaces

auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
address 192.168.1.3
netmask 255.255.255.0
gateway 192.168.1.1
  
```

**Gambar 6. IP Address Real Server**

### 3.5 Penjadwalan Round Robin

Deklarasi penjadwalan dibuat dalam dua kelompok berdasarkan *network* yang ditanganinya. Pada akses lokal dan akses publik terlihat bahwa pendeklarasian secara berulang, tetapi dengan *network* yang berbeda. Baris pertama untuk akses lokal menyatakan bahwa *IP Address* sebagai *virtual service* menggunakan penjadwalan *round robin*. Baris ke dua dan tiga mendefinisikan setiap paket yang datang menggunakan *port 80* akan diteruskan menuju *real server 1* dan *real server 2* dengan masing-masing 5 beban untuk setiap *real server*. Begitu pula pada akses publik, pada baris pertama menyatakan *Ip address* sebagai *virtual service*. Baris kedua dan ketiga menyatakan bahwa setiap paket yang datang menggunakan *port 80* akan

diteruskan menuju *real server* 1 dan *real server* 2 dengan masing-masing 5 beban.

```

root@director-server: ~
File Edit View Terminal Help
GNU nano 2.2.2 File: /etc/rc.local Modified
# In order to enable or disable this script just change the execut
# bits.
#
# By default this script does nothing.
#
#AKSES LOKAL
ipvsadm -A -t 10.237.2.9:80 -s rr
ipvsadm -a -t 10.237.2.9:80 -r 192.168.1.2:80 -m -w 5
ipvsadm -a -t 10.237.2.9:80 -r 192.168.1.3:80 -m -w 5
#akses PUBLIK
ipvsadm -A -t 222.124.3.141:80 -s rr
ipvsadm -a -t 222.124.3.141:80 -r 192.168.1.2:80 -m -w 5
ipvsadm -a -t 222.124.3.141:80 -r 192.168.1.3:80 -m -w 5

```

**Gambar 7. Penjadwalan Round Robin**

### 3.6 Mengamati Kinerja Director

Penjadwalan yang telah dibuat dapat dilihat dengan menguji apakah paket yang datang menuju *virtual server* dapat diteruskan menuju *real server*. Pengamatan dilakukan dengan mengamati *ipvsadm* melalui perintah `watch -n1 ipvsadm -ln`. Dengan perintah tersebut maka setiap paket yang datang menuju *director* akan ditampilkan secara real time seperti yang di tunjukkan pada gambar di bawah ini.

```

root@director-server: ~
File Edit View Terminal Help
Every 1.0s: ipvsadm -ln Wed Jul 13 09:19:14 2011
IP Virtual Server version 1.2.1 (size=4096)
Prot LocalAddress:Port Scheduler Flags
-> RemoteAddress:Port Forward Weight ActiveConn InActConn
TCP 222.124.3.141:80 rr
-> 192.168.1.2:80 Masq 5 0 0
-> 192.168.1.3:80 Masq 5 0 0
TCP 10.237.2.9:80 rr
-> 192.168.1.2:80 Masq 5 0 0
-> 192.168.1.3:80 Masq 5 0 0

```

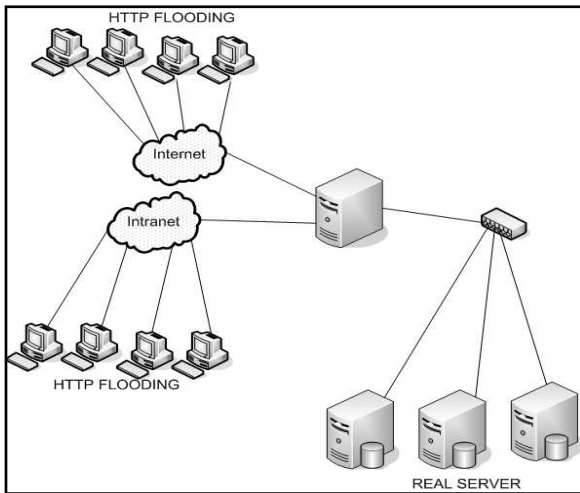
### Gambar 8. Director Meneruskan Paket

Dari gambar 8 terlihat bawah *director* telah siap untuk meneruskan paket yang datang pada `http://elearning.binadarma.ac.id` menggunakan *port* 80 baik dari *intranet* maupun *internet*. Monitoring secara *real time* ini bertujuan untuk melihat jumlah koneksi yang datang maupun koneksi yang sedang terjadi terhadap *real server*.

### 3.7 Uji Ketersediaan LVS NAT E-learning

Pengujian terhadap *e-learning* setelah menggunakan LVS NAT dianggap perlu sebagai acuan untuk membuktikan apakah tingkat ketersediaan layanan dapat meningkat. Dalam pengujian ini peneliti mengharapkan ada hasil berupa angka dan gambar untuk membuktikan bahwa LVS NAT telah mampu membagi beban proses yang harus diselesaikan oleh *e-learning*. Cara pengujian dilakukan terhadap layanan HTTP dan topologi jaringan LVS yang telah diimplementasikan. *Tools* yang digunakan sebagian diambil dari *internet* dan buku hacking yang peneliti pelajari untuk menguji ketahanan sebuah layanan *website*.

Pada pengujian terhadap Apache, pertama peneliti akan menggunakan *tools* ab. Pengujian akan dilakukan dari 2 arah yaitu *intranet* dan *internet* dengan menggunakan 8 komputer yang bertindak sebagai *client*. Setiap *client* akan mengirim 1000 paket dalam 10 tahapan melalui *port* 80. Topologi pengujian ditunjukkan pada gambar 9.



**Gambar 9. Topologi Pengujian**

Saat pengujian menggunakan ab, peneliti juga mencoba mengakses layanan *e-learning* secara bersamaan dengan tujuan untuk membuktikan bahwa beban *Apache* yang tinggi tidak menghilangkan ketersediaan layanan *e-learning*. Hasil pengujian dari masing komputer penyerang peneliti buat dalam bentuk tabel.

**Tabel 1. Hasil Pengujian ab**

Network	Jumlah Paket	Respon	Waktu (s)
Intranet	1000	243	526.559
Intranet	1000	239	517.267
Intranet	1000	48	543.123
Intranet	1000	37	548.415
Internet	1000	14	574.647
Internet	1000	292	700.007
Internet	1000	200	647.784
Internet	1000	161	753.003

Dengan menjalankan perintah `top` pada terminal, terlihat jelas bahwa layanan *Apache* memiliki beban kerja yang sangat tinggi ketika dibanjiri dengan ribuan paket yang datang secara bersamaan. Layanan *Apache* menduduki peringkat ke satu sebagai proses yang paling tinggi. Bahkan *Apache* dibebani dengan proses yang berlapis, di mana proses yang datang belum

terselesaikan namun proses yang baru terus menerus datang sehingga sangat terbebani. Dari hasil pengujian peneliti masih dapat mengakses *e-learning* dan menggunakan fasilitas yang disediakan tanpa merasa ada gangguan yang berarti. Beban *director* dan *real server* yang menjalankan *Apache* ditunjukkan pada gambar 10.

```

root@director-server:~# ipvsadm
IP Virtual Server version 1.2.1 (size=4096)
Prot LocalAddress:Port Scheduler Flags
  -> RemoteAddress:Port   Forward Weight ActiveConn InActConn
TCP  141.subnet222-124-3.astinet. rr
  -> elearning.local:www    Masq   5   3   297
  -> elearning.local:www    Masq   5   7   342
TCP  director-server.local:www rr
  -> elearning.local:www    Masq   5  14   710
  -> elearning.local:www    Masq   5  26   737
root@director-server:~# ipvsadm
IP Virtual Server version 1.2.1 (size=4096)
Prot LocalAddress:Port Scheduler Flags
  -> RemoteAddress:Port   Forward Weight ActiveConn InActConn
TCP  141.subnet222-124-3.astinet. rr
  -> elearning.local:www    Masq   5   7   885
  -> elearning.local:www    Masq   5  33   870
TCP  director-server.local:www rr
  -> elearning.local:www    Masq   5   7   970
  -> elearning.local:www    Masq   5  33   955
root@director-server:~# ipvsadm -f
Try 'ipvsadm -h' or 'ipvsadm --help' for more information.
root@director-server:~# ipvsadm
IP Virtual Server version 1.2.1 (size=4096)
Prot LocalAddress:Port Scheduler Flags
  -> RemoteAddress:Port   Forward Weight ActiveConn InActConn
TCP  141.subnet222-124-3.astinet. rr
  -> elearning.local:www    Masq   5   0   731
  -> elearning.local:www    Masq   5  46   689
TCP  director-server.local:www rr
  -> elearning.local:www    Masq   5   0   710
  -> elearning.local:www    Masq   5  40   683
root@director-server:~#

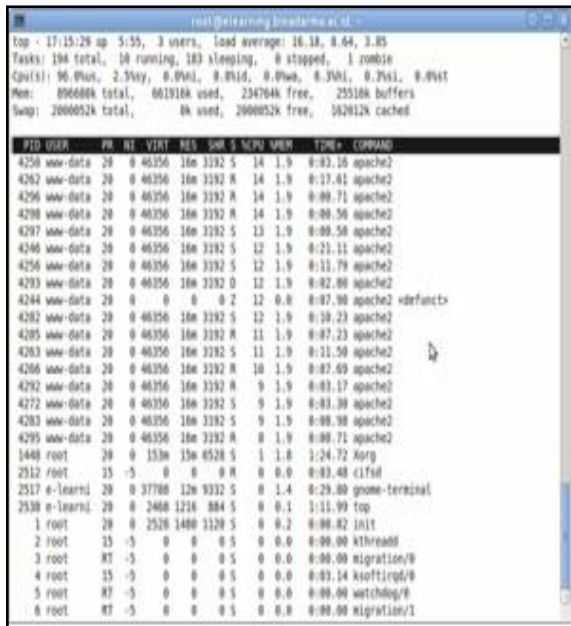
```

**Gambar 10. Director Dibanjiri Paket**

The screenshot shows a terminal window with the output of the `top` command. At the top, it displays system statistics: 'Top - 17:40:32 up 0:20, 3 users, load average: 33.05, 32.04, 8.09'. Below this, it lists tasks: 'Tasks: 288 total, 4 running, 284 sleeping, 0 stopped, 0 zombie'. The CPU usage is shown as 'Cpu(s): 48.24us, 10.74sy, 0.04ni, 0.0xid, 39.06sw, 0.59hi, 1.06si, 0.00st'. Memory usage is 'Mem: 896080k total, 888756k used, 7924k free, 852k buffers'. Swap usage is 'Swap: 2000032k total, 47608k used, 1523904k free, 000k cached'. The main part of the screenshot is a table of running processes, with 'www-data' (Apache) processes at the top, indicating high activity.

**Gambar 11. Apache Real Server 1**





Gambar 12. Apache Real Server 2

### 3.8 Mengukur Waktu Respon *E-learning*

Untuk mengukur waktu respon *e-learning*, peneliti menggunakan *tools* *httperf* yang berjalan pada sistem operasi linux. Dengan menggunakan *httperf* peneliti dapat memperlihatkan *time response*, *throughput* dan *request lost* dari layanan *e-learning*. Untuk menggunakan *httperf* akan dilakukan installasi dengan menjalankan perintah `sudo apt-get install httperf`. Topologi pengujian menggunakan *httperf* masih tetap menggunakan topologi sebelumnya saat menggunakan *tools* *ab*. Perintah *httperf* yang digunakan cukup sederhana yaitu `httperf -hog -server elearning.binadarma.ac.id-num-conn 1000 -ra 100 -timeout 5`. Perintah tersebut akan mengirimkan paket kepada *server e-learning* dengan 1000 koneksi. Penulis menjalankan perintah tersebut secara bersamaan pada delapan unit komputer dengan harapan *server* akan sangat terbebani. Nilai *Time*

*response* didapat dari *reply rate*, *throughput* didapatkan dari *miscellaneous section* yaitu *Net I/O* dan *request lost* didapatkan dari *error section* pada *connrefused* dan *connreset*.

Tabel 2. Pengujian *httperf*

Network	Time Response (ms)	Thoughtput	Request Lost
Intranet	2793.6 (23.2 req/s)	10.0 KB/s	0
Intranet	4463.3 (13.2 req/s)	1.5 KB/s	0
Intranet	2406.1 (6.0 req/s)	1.1 KB/s	0
Intranet	1863.5 (5.4 req/s)	2.4 KB/s	0
Internet	0.0 (11.9 req/s)	0.9 KB/s	0
Internet	0.0 (4.2 req/s)	0.3 KB/s	0
Internet	0.0 (3.0 req/s)	0.2 KB/s	0
Internet	0.0 (6.6 req/s)	0.5 KB/s	0

## 4. SIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan dengan melakukan sejumlah tahapan perancangan dan pemodelan hingga pada tahap pengujian, maka penelitian ini mendapatkan sejumlah kesimpulan sebagai berikut: (1) Penelitian ini telah menghasilkan sebuah model implementasi jaringan yang menggunakan teknologi LVS NAT. Model yang diimplementasikan adalah dengan menambah sejumlah perangkat komputer yang memiliki fungsi sebagai *director* dan *real server*; (2) Dengan menggunakan LVS NAT *server e-learning* pada Universitas Bina Darma telah mampu untuk menghadapi beban besar pada layanan Apache dimana jumlah *session* yang datang untuk melakukan koneksi dapat dipenuhi

dengan konsistensi tinggi sehingga *e-learning* mampu tersedia walaupun dibanjiri *request* pada *service* HTTP. Pernyataan tersebut didukung oleh sejumlah pengujian dan percobaan secara nyata pada objek penelitian.

/0/miscuts.html, diakses pada 18 Oktober 2010).

## DAFTAR RUJUKAN

- Ashari, Ahmad. 2008. *Linux System Administrator*. Informatika. Jakarta.
- Aidil, Chendramata dan Nasrullah. 2008. *Keamanan Web Server*. DEPKOMINFO. Jakarta.
- Ferrianto, Gozali dan Alex. 2002. *Virtual Server*. Jurnal JETri Universitas Trisakti. Vol. 2 No. 1, Agustus: 54:68.
- Heriyadi, Zulhaidi. 2002. *On The Construction & Development of Simple PCs Cluster for High Performance Computers. Komputer dan Sistem Intelijen (KOMMIT2002)*. Universitas Gunadarma. Jakarta.
- Kresno R. Aji. 2002. *Kejahatan Internet Trik Aplikasi dan Tip Penanggulangannya*. Elex Media Komputindo. Jakarta.
- KEMENDIKNAS. *Visi Misi Pendidikan*. Online. (<http://www.kemdiknas.go.id/kemdikbud/tentang-kemdikbud-visi>, diakses pada 18 Oktober 2011).
- LVS. *Linux Virtual Server*. Online. (<http://linuxvirtualserver.org>, diakses pada 18 Oktober 2011).
- Romi, Satria, Wahono. 2008. *Pengantar e-Learning dan Pengembangannya*. Online. <http://ilmukomputer.org/2008/11/25/pengantar-elearning-dan-pengembangannya>, diakses tanggal 18 Oktober 2010).
- UPT-SIM, Universitas Bina Darma. *Network Operation Center*. Online. (<http://http://binadarma.ac.id/content/120>