

# ***MULTIPLE SETTING UNTUK PENGATURAN BANDWIDTH MENGUNAKAN SQUID PROXY SERVER***

**Yan Gunawan, \* Nina Sevani**  
**Fakultas Teknik dan Ilmu Komputer Jurusan Teknik Informatika**  
**Universitas Kristen Krida Wacana**  
**[\\*nina.sevani@ukrida.ac.id](mailto:nina.sevani@ukrida.ac.id)**

## ***Abstract***

*Unappropriate in bandwidth scheduling making some effect with internet café's performance. Lack of bandwidth can slowdown response time for internet. This condition absolutely contradict with user needs. Squid proxy server is one of solutions to keep internet access stability. We can configure proxy server to scheduling bandwidth depend with client's rush hour, blocking harmful website, and accelerate access from client to frequently visited website. The experiment result show significant difference access time to a website from client that using proxy server and client without using proxy server. The experiment also shows that with scheduling bandwidth, client of internet café get bandwidth correspond with their needs.*

**Keywords:** *scheduling, bandwidth, squid proxy server, blocking, access time, website.*

## **1. PENDAHULUAN**

### **1.1 Latar Belakang**

Kompleksitas fungsi dan penggunaan jaringan serta penambahan *user* jaringan yang cukup pesat, membutuhkan proses perancangan serta konfigurasi yang juga semakin kompleks. Penggunaan jaringan oleh *user* saat ini tidak hanya terbatas pada jaringan skala lokal/*local area network* (LAN), seperti dalam sebuah lantai atau gedung saja. Kehadiran internet sebagai salah satu bentuk jaringan skala global/*wide area network* (WAN) sudah menjadi kebutuhan pokok dalam kehidupan *user* jaringan sehari-hari. Beragam layanan yang disediakan internet mulai dari *chatting, browsing, e-mail, game online*, sampai dengan transaksi elektronik, dilakukan setiap hari oleh *user* dari beragam tingkat usia.

Maraknya penggunaan internet inilah yang pada akhirnya digunakan sebagian pihak untuk membangun bisnis warung internet (warnet). Dengan biaya yang cukup murah, layanan koneksi yang semakin cepat, bisnis warnet telah menjangkau berbagai kalangan *user* internet. Seiring maraknya penggunaan internet melalui warnet, pengelola warnet juga dituntut untuk dapat memberikan layanan yang semakin baik kepada *user*, khususnya dalam masalah koneksi dan keamanan akses. Bagi warnet yang berskala besar, operasional akan didukung oleh beberapa *server* (*multi server*), dimana setiap *server* mempunyai fungsinya masing-masing. Hal ini dilakukan untuk meningkatkan efektivitas layanan kepada *user*, karena akses internet tidak akan dibebankan kepada satu *server* saja.

Kebutuhan layanan setiap *user* akan internet sudah tentu berbeda-beda, termasuk pula waktu aksesnya. Kondisi ini mengharuskan pengelola warnet untuk dapat menyediakan *bandwidth* yang cukup memadai untuk mengakomodir setiap kebutuhan *user* mereka. Untuk menyediakan pengaturan *bandwidth* yang efektif dan efisien dapat digunakan *squid proxy server*. Melalui *proxy server*, pengelola warnet dapat mengoptimalkan penggunaan *bandwidth* sesuai dengan jam kesibukan setiap *server* yang

ada di warnet. *Proxy server* akan menyimpan alamat-alamat *website* yang sering dikunjungi oleh *user* pada *cache proxy* sehingga *user* akan lebih cepat untuk mendapatkan akses ke *website* tersebut.

Penggunaan *proxy server* juga dapat dimanfaatkan untuk proses autentikasi *user*, *blocking website*, dan sebagainya. Mengingat beragamnya *proxy server* yang dapat digunakan, seperti *elite proxy*, *transparent proxy*, *anonymous proxy*, serta *squid proxy*, pemilihan *proxy server* yang tepat tentunya dapat semakin meningkatkan daya saing warnet yang menggunakannya. Di antara beragamnya jenis *proxy server*, penggunaan *squid proxy server* menawarkan beberapa keunggulan, seperti kestabilan dalam penggunaan, kemampuan *filtering*, kemudahan konfigurasi, serta biaya yang terjangkau. Keunggulan inilah yang pada akhirnya membuat *squid proxy server* menjadi pilihan yang akan digunakan dalam implementasi kali ini.

## 1.2 Identifikasi Masalah

Berdasarkan latar belakang di atas, maka dapat diidentifikasi beberapa masalah menyangkut akses internet, khususnya melalui warnet yaitu:

1. Masih kurang optimalnya penggunaan *bandwidth* pada warnet.
2. Perbedaan waktu sibuk setiap jenis *server* dan *client* yang ada pada warnet.
3. Adanya ketidaksesuaian antara *bandwidth* yang disediakan dengan waktu sibuk dari setiap *server* dan *client* yang ada.

## 1.3 Tujuan

Tujuan dari perancangan dan implementasi pengaturan *bandwidth* dengan *proxy server* pada warnet ini adalah:

1. Membagi *bandwidth* secara lebih maksimal sesuai kebutuhan setiap *server* dan *client* pada warnet.
2. Mempercepat *user* dalam mengakses *website* yang sering dikunjungi.
3. Mempercepat proses unduh gambar melalui *email*.
4. Membatasi akses *user* pada *website* tertentu, khususnya *website* yang dianggap berbahaya.

## 1.4 Manfaat

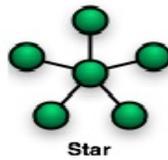
Manfaat yang dapat diperoleh dari pengaturan *bandwidth* menggunakan *squid proxy server* adalah:

1. Mengurangi kepadatan *traffic* data di internet.
2. Menghemat penggunaan *bandwidth*.
3. Meningkatkan pelayanan kepada *user* dengan memberikan akses internet yang lebih cepat dan aman.

## 2 LANDASAN TEORI

### 2.1 Topologi Star

Karakteristik dari topologi jaringan ini adalah *node (station)* berkomunikasi langsung dengan *station* lain melalui *central node (hub/switch)*, *traffic* data mengalir dari *node* ke *central node* dan diteruskan ke *node (station)* tujuan [1]. Jika salah satu segmen kabel putus, jaringan lain tidak akan terputus. Gambar 1 menunjukkan topologi *star*.

Gambar 1. Topologi *Star*

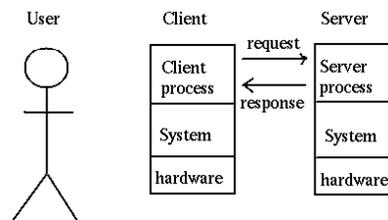
## 2.2 *Switch*

Switch adalah perluasan dari konsep *bridge*. Terdapat dua arsitektur dasar yang digunakan pada *switch*, yaitu *cut-through* dan *store-and-forward*. *Switch cut-through* memiliki kelebihan di sisi kecepatan karena ketika sebuah *packet* datang, *switch* hanya memperhatikan alamat tujuannya sebelum meneruskan ke segmen tujuan. *Switch store-and-forward*, kebalikannya, menerima dan menganalisis seluruh isi *packet* sebelum meneruskannya ke tujuan. Waktu yang diperlukan untuk memeriksa satu *packet* sangat banyak, tetapi ini memungkinkan *switch* untuk mengetahui adanya kerusakan pada *packet* dan mencegahnya agar tak mengganggu jaringan.

## 2.3 *Client Server*

Sistem *client - server* dapat diterapkan dengan teknologi *internet*, di mana terdapat suatu unit komputer yang berfungsi sebagai *server* yang hanya memberikan layanan kepada *client* [2]. *Client* hanya dapat memanfaatkan sumber daya yang disediakan oleh *server* sesuai dengan otoritas yang diberikan oleh *administrator*. Aplikasi yang dijalankan pada sisi *client* bisa saja merupakan sumber daya yang tersedia di *server* atau aplikasi yang di-*install* di sisi *client* namun hanya bisa dijalankan saat terhubung ke *server*.

*Server* bersifat pasif dan hanya menunggu *request* dari *client*, untuk kemudian dibalas dengan mengirimkan *service* kepada *client* yang mengirimkan *request* tersebut. *Client* bersifat aktif mengirimkan *request* kepada *server* dan menunggu balasan *service* yang dikirimkan oleh *server* [3]. Gambar 2 menunjukkan gambar dari sistem *client server*.

Gambar 2. Sistem *Client - Server*

## 2.4 *Proxy Server*

*Application gateway* dapat disebut juga dengan *bastion host*, *proxy gateway*, *proxy server* [4]. Definisi *proxy server* adalah suatu aplikasi yang berada di antara *client* dan *server* yang berfungsi untuk *security*, *monitoring*, dan *performance* antara 2 *network*.

*Proxy* ini berfungsi sebagai perantara yang berada di tengah untuk suatu *network* antara *client* dan *server* [5]. *Client* dapat berupa *browser web*, *client FTP (File Transfer Protocol)*, dan sebagainya, sedangkan *server* dapat berupa *server web*, *server FTP* dan sebagainya. *Proxy server* memiliki 3 fungsi utama, antara lain:

1. **Connection Sharing:** sebagai perantara antara pengguna dari *server* di *internet*.
2. **Filtering:** berfungsi sebagai *firewall packing filtering* yang digunakan untuk melindungi jaringan lokal dari serangan atau gangguan yang berasal dari jaringan *internet* dengan cara melakukan *filtering* atas *packet* yang lewat dari dan ke jaringan-jaringan yang dihubungkan dan dapat dikonfigurasi untuk menolak akses ke *website* tertentu pada waktu - waktu tertentu.
3. **Caching:** mekanisme penyimpanan objek-objek yang sudah pernah diminta dari *server* di *internet*, sehingga mempercepat proses pelayanan.

### 3 PEMBAHASAN

Proses perancangan dan konfigurasi *squid proxy server* untuk pengaturan *bandwidth* ini dilakukan melalui serangkaian metode, seperti studi literatur, observasi, serta uji coba hasil perancangan. Studi literatur dilakukan dengan membaca buku, informasi dari *website*, serta jurnal terkait untuk memperoleh data, informasi, serta teori-teori yang dibutuhkan untuk perancangan dan konfigurasi. Observasi dilakukan pada warnet SaymoreNet di daerah Bekasi, dan tujuannya untuk mengetahui kondisi warnet yang memerlukan penggunaan *proxy server* dalam operasionalnya, termasuk tentang berbagai permasalahan yang ada di dalam operasional warnet. Hasil observasi dijadikan referensi perancangan dan konfigurasi terhadap *proxy server*, yang kemudian hasilnya akan diuji coba untuk mengetahui dampak konfigurasi terhadap operasional warnet.

#### 3.1 Kondisi Jaringan SaymoreNet Tanpa *Squid Proxy Server*

Dari hasil observasi dapat diketahui bahwa SaymoreNet merupakan warnet yang berskala menengah dengan jumlah komputer *client* sebanyak 32 unit yang dibagi dalam dua divisi. Setiap divisi menyediakan jenis layanan berbeda, yaitu divisi *game* dan divisi *browsing* dimana 20 unit merupakan *client* untuk divisi *game* dan 10 unit merupakan *client* untuk divisi *browsing*. Dua unit komputer sisanya merupakan komputer *server*. *Server* pertama berfungsi untuk pengaturan akses jaringan termasuk pembagian *bandwidth* untuk *client* sedangkan *server* kedua berfungsi untuk menampung seluruh data dari setiap divisi yang ada. SaymoreNet merupakan warnet 24 jam dan tidak menggunakan *proxy server* dalam operasionalnya saat ini. Kondisi seperti ini yang membuat pembagian *bandwidth* pada SaymoreNet dirasa kurang maksimal, karena tidak dapat memenuhi kebutuhan *bandwidth* setiap *client*-nya pada saat diperlukan. Tabel 1 berikut ini menjelaskan pembagian *bandwidth* untuk setiap divisi di SaymoreNet.

Tabel 1. Pembagian *bandwidth* pada SaymoreNet

<b>Komputer</b>	<b>Bandwidth</b>
<i>Server</i>	256 Kbps
Divisi <i>Game</i>	1280 Kbps
Divisi <i>Browsing</i>	1280 Kbps

Dengan pembagian *bandwidth* seperti terlihat pada Tabel 1 di atas, beberapa *user* di SaymoreNet merasa bahwa akses internet mereka terasa lebih lambat, khususnya pada waktu-waktu tertentu, karena pengaturan *bandwidth* yang tidak seimbang. Pada pagi hari

sekitar pukul 08.00-12.00 WIB, lebih banyak *user* dari divisi *browsing*, sedangkan siang sampai malam hari sekitar pukul 12.00-20.00 WIB, lebih banyak *user* dari divisi *game*. Di atas pukul 20.00 WIB, jumlah *user* dari divisi *game* dan divisi *browsing* lebih berimbang. Melihat perbedaan kesibukan dari setiap divisi, maka untuk dapat memenuhi kebutuhan dari setiap divisi secara lebih maksimal diperlukan pengaturan *bandwidth* yang juga berbeda dari waktu ke waktu. Dengan pengaturan *bandwidth* yang tepat untuk masing-masing divisi, maka *bandwidth* yang tersedia dapat digunakan dengan lebih efektif dan efisien, seluruh *user* juga akan mendapatkan layanan yang lebih baik, khususnya terkait layanan dalam hal koneksi internet.

### 3.2 Perbandingan Beberapa Proxy Server

Pemilihan *squid proxy server* pada SaymoreNet dilakukan setelah mengobservasi dan membandingkan beberapa *proxy server* yang ada, seperti *middleman*, *oops*, dan *privoxy*. Tabel 2 berikut ini merupakan hasil perbandingan dari empat *proxy server* yang dilakukan sebelum proses konfigurasi *squid proxy server*.

Tabel 2. Hasil perbandingan fitur Proxy Server

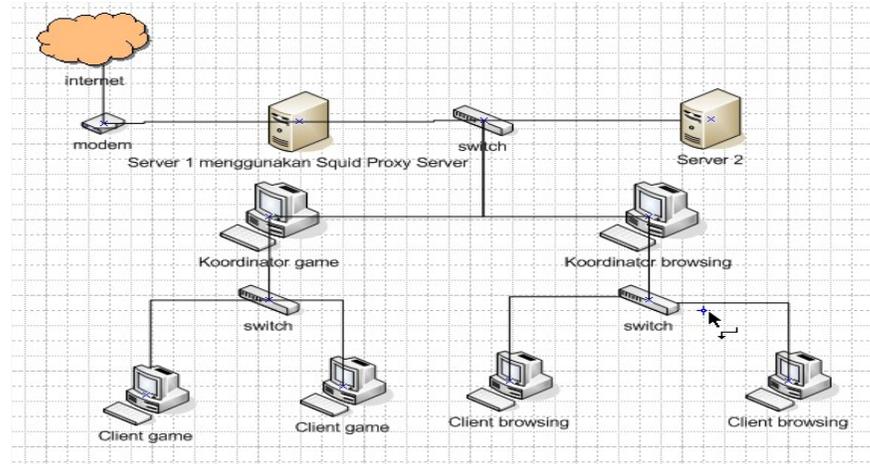
Proxy Server	Fitur						
	Content catching	Transparent proxy	ACL	Support HTTP dan FTP	Autentifikasi modul	Bandwidth management	Third party software
Middleman	×	×	×	×	×	-	×
Oops	×	-	×	×	×	×	-
Privoxy	×	-	×	×	×	-	-
Squid	×	×	×	×	×	×	×

Keterangan : × terdapat fitur tersebut  
- tidak terdapat fitur tersebut

Dari Tabel 2 di atas dapat dilihat bahwa *squid proxy server* mempunyai fitur yang lebih lengkap. Selain itu *squid proxy server* juga dapat membantu pihak pengelola warnet untuk menekan biaya operasional. Dengan menggunakan *squid proxy server* yang berupa *software* yang dapat dikombinasikan dengan *switch*, pengelola warnet tidak perlu membeli *router* mikrotik. Berdasarkan kelengkapan fitur yang dimilikinya serta kemudahan dalam konfigurasi, diharapkan penerapan *squid proxy server* akan membuat pemanfaatan *bandwidth* di SaymoreNet menjadi lebih efisien dan efektif sesuai kondisi *user* yang membutuhkan.

### 3.3 Kondisi Jaringan pada SaymoreNet dengan Squid Proxy Server

Fitur-fitur yang akan digunakan dalam *squid proxy server* berfungsi untuk mengatur pembagian *bandwidth* secara terjadwal, berdasarkan waktu kesibukan setiap divisi yang ada di SaymoreNet. Selain itu juga akan digunakan fitur untuk melakukan *blocking* terhadap beberapa situs yang dianggap berbahaya karena mengandung materi kekerasan atau pornografi. Dengan melakukan pengaturan *bandwidth* secara terjadwal serta *blocking* situs berbahaya, maka kecepatan akses internet bagi *user* dapat lebih maksimal karena sesuai dengan kondisi *user* di warnet serta *bandwidth* benar-benar hanya digunakan untuk akses pada situs-situs yang tidak berbahaya. Gambar 3 berikut ini merupakan topologi jaringan yang menggunakan *squid proxy server* di SaymoreNet.



Gambar 3. Topologi jaringan SaymoreNet menggunakan *Squid Proxy Server*

Dengan topologi seperti pada Gambar 3 di atas, maka pada setiap divisi akan terdapat satu komputer *client* yang berfungsi sebagai koordinator divisi tersebut. Koordinator ini yang akan bertugas sebagai perantara antara *server* dengan *client* pada divisi tersebut. Pembagian *bandwidth* diatur oleh *squid proxy server* yang terdapat pada *server* pertama yang sejak awal berfungsi sebagai pengatur *bandwidth*. Pembagian *bandwidth* diatur secara manual kepada setiap koordinator divisi, dan setiap divisi mendapatkan *bandwidth* sesuai dengan jam kesibukannya. Pada waktu-waktu sibuk suatu divisi akan mendapatkan *bandwidth* yang lebih besar dibanding divisi lain yang tidak sibuk. Pada saat tingkat kesibukan divisi menurun, maka *bandwidth* akan disalurkan kepada divisi lain yang membutuhkan.

Melihat topologi jaringan dengan menggunakan *squid proxy server* pada Gambar 3 di atas, akan tampak bahwa tidak dilakukan banyak perubahan terhadap kondisi awal jaringan yang tidak menggunakan *proxy server*. Perbedaan utama terletak pada *server* pertama dimana akan diinstal *squid proxy server*. *Server* kedua tidak mengalami perubahan fungsi apapun, tetap sebagai penyimpan data dari *client* seluruh divisi. Perubahan berikut terletak pada keberadaan komputer *client* yang berfungsi sebagai koordinator setiap divisi. Jumlah *switch* dan spesifikasi komputer *client* serta *server* yang digunakan masih sama dengan kondisi sebelum dilakukan instalasi *squid proxy server*.

Selain sedikit perubahan pada topologi jaringan yang digunakan, pengaturan *bandwidth* juga ikut mengalami perubahan sesuai waktu kesibukan setiap divisi. Tabel 3 sampai 5 berikut ini menampilkan pembagian *bandwidth* untuk setiap divisi, berdasarkan waktu kesibukan masing-masing yang dibagi menjadi tiga *shift*. *Shift* pertama yaitu *shift* pagi sampai siang, dimana divisi *browsing* akan diberikan *bandwidth* lebih besar daripada divisi *game*, karena pada *shift* pertama lebih banyak *user* divisi *browsing* daripada *user* divisi *game*. Pada *shift* kedua, yaitu *shift* siang sampai malam, jumlah *user* divisi *game* lebih banyak sehingga dialokasikan *bandwidth* lebih besar untuk divisi *game*. *Shift* ketiga, yaitu malam sampai pagi akan dialokasikan *bandwidth* dalam jumlah yang sama untuk kedua divisi, mengingat pada *shift* ini jumlah *user* untuk setiap divisi sama banyaknya.

Tabel 3. Pembagian *bandwidth shift* 1 (08.00-12.00 WIB)

Nama Divisi	<i>Bandwidth yang Didapat</i>	Pemblokiran Situs
<i>Server</i>	256 Kbps	<i>Block</i>
<i>Game</i>	1024 Kbps	<i>Block</i>
<i>Browsing</i>	1500 Kbps	<i>Block</i>

Tabel 4. Pembagian *bandwidth shift* 2 (12.01-20.00 WIB)

Nama Divisi	<i>Bandwidth yang Didapat</i>	Pemblokiran Situs
<i>Server</i>	256 Kbps	<i>Block</i>
<i>Game</i>	1500 Kbps	<i>Block</i>
<i>Browsing</i>	1024 Kbps	<i>Block</i>

Tabel 5. Pembagian *bandwidth shift* 3 (20.01-07.59 WIB)

Nama Divisi	<i>Bandwidth yang Didapat</i>	Pemblokiran Situs
<i>Server</i>	256 Kbps	<i>Block</i>
<i>Game</i>	1280 Kbps	<i>Block</i>
<i>Browsing</i>	1280 Kbps	<i>Block</i>

### 3.4 Konfigurasi Squid Proxy Server

Mekanisme konfigurasi *squid proxy server* tergantung pada *web browser* yang digunakan. Pada implementasi kali ini akan digunakan Mozilla Firefox versi 3.5.15 karena dapat mencegah tindakan *phising* ke komputer. *Phising* sendiri adalah upaya seseorang untuk mencuri identitas pribadi yang dimiliki yang biasanya anda masukkan pada komputer. Data-data yang dicuri biasanya berupa *mail*, *password*, nomor telepon, nomor rekening bank, *paypal*, dan data penting lain yang Anda miliki.

#### 3.4.1 Konfigurasi pada Web Browser Client dan Server

Berikut ini adalah tahapan konfigurasi *squid proxy server* pada *web browser client* dan *server*:

- a. Buka *web browser* Mozilla, kemudian pilih menu *Tools* dan pilih *Option*, untuk masuk ke *windows Options*. Pada *windows* ini, pilih menu *Advanced*, pilih *tab Network* dan klik tombol *Settings*, untuk masuk ke *windows Connection Settings*.
- b. Pada *windows Connection Settings* ini, lakukan beberapa konfigurasi tentang sistem yang sedang berjalan, seperti:
  - i. Untuk komputer *server*, set *HTTP Proxy* 127.0.0.1 *port* 3128 sedangkan untuk komputer *client*, set *HTTP Proxy* sesuai dengan *IP server* dari *client* tersebut.
  - ii. Pada *checkbox use this proxy server for all protocols*, berikan tanda centang.

- iii. Klik tombol OK untuk menyimpan dan mengakhiri konfigurasi dan kembali ke *windows Options*.
- c. Pada *windows Options*, di bagian *Offline Storage*, ganti angka yang ada dengan angka 0 dan klik tombol *Clear Now*, yang berarti tidak akan membatasi *space cache memory*.
- d. Klik tombol OK untuk mengakhiri konfigurasi pada *web browser client* dan *server* ini.
- e. Untuk memastikan apakah *squid proxy server* sudah berjalan, masuk ke *monitor squid access.log*, melalui menu *Start*. Pilih *All Program*, lalu *squid ccpb*, dan pilih *monitor squid access.log*. Apabila ada penambahan *command* setiap kali *user* melakukan percobaan akses ke suatu situs, maka *squid proxy server* sudah berjalan.

Setelah melakukan konfigurasi pada *web browser server* dan *client*, maka hal berikutnya adalah konfigurasi fungsi pada *squid proxy server* sebelum melakukan konfigurasi fitur lain pada *squid proxy server*. Konfigurasi fungsi yang diperlukan antara lain penetapan kapasitas *cache memory* yang akan digunakan, serta berapa lama waktu untuk menyimpan alamat situs yang diakses oleh *user* dalam *cache memory* sebelum dilakukan penghapusan alamat situs tersebut.

### 3.4.2 Konfigurasi Access Control List (ACL)

Konfigurasi berikutnya adalah konfigurasi ACL yang berfungsi untuk mengendalikan apa yang dapat diakses dan tidak dapat diakses oleh *squid proxy server*. Gambar 4 berikut adalah potongan *coding* yang digunakan dalam konfigurasi ACL.

```
acl all src 0.0.0.0/0.0.0.0
acl manager proto cache_object
acl localhost src 127.0.0.1/255.255.255.255
acl to_localhost dst 127.0.0.0/24
acl localnet src 192.168.0.1/24
acl SSL_ports port 443 #https
acl Safe_ports port 80 # http
acl Safe_ports port 21 # ftp
```

Gambar 4. Potongan konfigurasi ACL

### 3.4.3 Konfigurasi Cache

*Cache* dapat diartikan sebagai tempat penyimpanan sementara, untuk menampung *request* yang datang, sehingga dapat mempercepat pelayanan kepada *client*. Konfigurasi di bawah ini adalah untuk dapat membagi *cache* ke dalam 4 *folder* yaitu *cache0*, *cache1*, *cache2*, dan *cache3*, agar tidak bertumpuk pada satu *folder* saja. *Admin* harus secara teratur memeriksa dan melakukan *defrag folder cache* minimal sebulan sekali, karena *folder* ini dapat memenuhi *drive D* yang menjadi tempat *folder cache*. Gambar 5 berikut ini merupakan potongan dari konfigurasi *cache* yang dilakukan.

```

cache_mem 64 MB
cache_dir ufs d:/squid/cache0 1000 16 256 #1000 = 1Gb
cache_dir ufs d:/squid/cache1 1000 16 256

```

Gambar 5. Potongan konfigurasi *Cache*

Fungsi sintak di atas adalah menentukan kapasitas setiap *cache memory* yang dibentuk, yaitu 256 Mb. Dimana *file cache* akan disimpan dalam *drive D*.

### 3.4.4 Konfigurasi *Bandwidth*

Konfigurasi berikutnya adalah konfigurasi *range IP* serta pengaturan *bandwidth* yang diperlukan sesuai waktu kesibukan setiap divisi. Gambar 6 berikut ini menampilkan sintak konfigurasi *bandwidth* pada SaymoreNet, di mana dapat dilihat *range IP* yang diberikan *server* untuk *client* dari setiap divisi. Termasuk sintak untuk mengizinkan ACL dari *server*, serta setiap divisi untuk mengakses internet melalui *squid proxy server*.

```

acl src server 192.168.0.1/24 - 192.168.0.2/24
acl src game 192.168.0.3/24 - 192.168.0.22/24
acl src browsing 192.168.0.23/24 - 192.168.0.32/24

http_access allow server
http_access allow game
http_access allow browsing
http_access allow localhost
http_access allow deny all

icp_access allow all

acl pagi time MTWTFAS 08:01-12:00
acl siang time MTWTFAS 12:01-20:00
acl malam time MTWTFAS 20:01-08:00

#MTWTFAS adalah Monday Tuesday dan seterusnya
#file yg bs di unduh

acl download url_regex -i .mp3 .mp4 .zip .rar .zip .avi .mpeg .jpg .bmp
      .gif .iso .wav .doc .pdf .ppt

#pembatasan besar file yang di unduh 100mb

```

Gambar 6. Sintak konfigurasi *bandwidth*

### 3.4.5 Konfigurasi *Blocking Situs*

*Blocking* situs ditujukan untuk memblokir situs yang dianggap berbahaya oleh pengelola SaymoreNet, karena mengandung virus, materi pornografi, dan perjudian. Sebelum melakukan pemanggilan fungsi blok situs, akan dibuat dahulu daftar nama situs yang ingin diblokir lalu simpan dengan format *text* (\*.txt) dan simpan dalam *drive C*, *folder squid* kemudian *folder etc* dan *save*. Pada implementasi kali ini di SaymoreNet,

akan dibentuk tiga file *txt* yaitu *bloksitus.txt*, *porno.txt*, dan *judi.txt*. Gambar 7 menunjukkan sintak konfigurasi blocking situs yang sudah tersimpan dalam tiga *file text* yang dibuat.

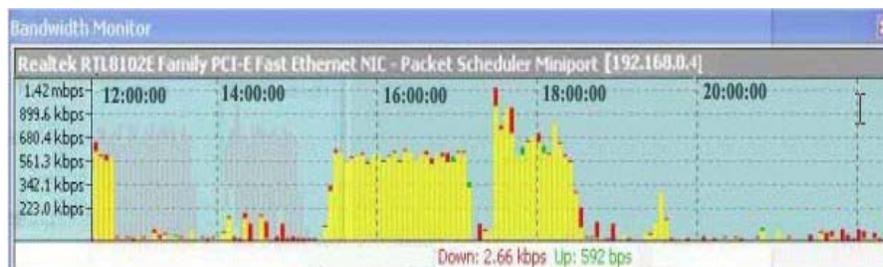
```
#Definisi waktu kerja warnet  
  
acl kerja time MTWTFAS 00:00-23:59  
  
#acl divisi sudah di jabarkan di atas  
  
acl bloksitus dstdomain "c:\squid\etc\bloksitus.txt"  
  
acl porno dstdomain "c:\squid\etc\porno.txt"  
  
acl judi dstdomain "c:\squid\etc\judi.txt"  
  
http_access deny bloksitus  
  
http_access deny porno  
  
http_access deny judi  
  
http_access deny all
```

Gambar 7. Konfigurasi *Blocking* situs

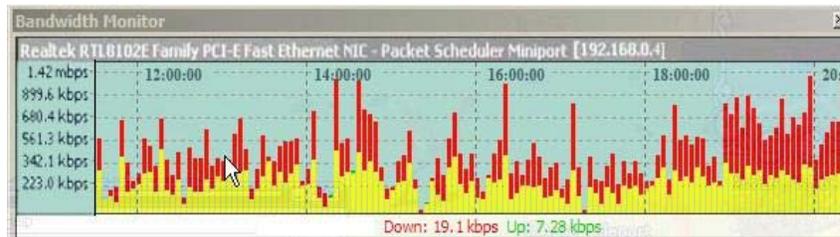
#### 4 EVALUASI

Setelah konfigurasi pada *web browser client* dan *server*, serta konfigurasi pada *squid proxy server*, maka dilakukan percobaan untuk melihat hasilnya. Untuk melihat hasil pengaturan *bandwidth* serta membandingkan dengan kondisi sebelum menggunakan *squid proxy server*, maka digunakan *software bandwidth monitoring* yang pengaturan waktunya telah di-*setting* untuk menampilkan *traffic* dari kecepatan mengunduh dan mengunggah setiap divisi. Di mana *traffic* merah menunjukkan kecepatan mengunduh (*download*), *traffic* hijau menunjukkan kecepatan mengunggah (*upload*), serta *traffic* kuning menunjukkan kecepatan melakukan *upload* dan *download* bersamaan.

Gambar 8 dan 9 menunjukkan grafik divisi *game* pada jam operasional siang sampai malam (*shift 2*), dan Gambar 8 merupakan grafik sebelum digunakan *squid proxy server*, sedangkan Gambar 9 merupakan grafik setelah digunakan *squid proxy server*. Tampak jelas bahwa pada kedua gambar bahwa pada *shift 2*, divisi *game* memang membutuhkan kapasitas sekitar 1.500 Kbps, di mana penggunaan *bandwidth* pada divisi *game* setelah ada *squid proxy server* lebih maksimal, baik untuk *download* ataupun melakukan *upload* dan *download* bersamaan.



Gambar 8. Grafik *shift 2* divisi *game* sebelum ada *Proxy Server*

Gambar 9. Grafik *shift 2* divisi *game* setelah ada *Proxy Server*

Sedangkan untuk melihat perbedaan kecepatan akses internet sebelum dan setelah menggunakan *squid proxy server*, dapat dilihat hasil percobaan pada Tabel 6. Di sana akan terlihat ketika *squid on* dan *cache on*, ternyata dapat mengakses situs lebih baik pada pembukaan kedua dan seterusnya. Namun apabila kondisi *squid off* dan *cache off*, saat mengakses situs kecepatannya lebih lambat dari pada saat *squid on* dan *cache on*.

Tabel 6. Perbandingan akses *internet*

Website	Kecepatan/Second	
	<i>Squid On</i>	<i>Squid Off</i>
www.mangafox.com	2,249	10.086
www.kaskus.com	1,073	1,221
www.google.com	0,899	1,182
www.krucil.com	2,355	3,318

## 5 KESIMPULAN

Berdasarkan percobaan yang telah dilakukan dengan mengamati waktu dan hasil akses yang dilakukan dari komputer *client* ke beberapa *website*, dapat disimpulkan bahwa:

1. Keberadaan *cache* pada *proxy server* meningkatkan kecepatan akses *user* dalam membuka *website*.
2. Penjadwalan pembagian *bandwidth* sesuai dengan waktu kesibukan setiap divisi dapat mengoptimalkan kinerja warnet, karena dapat meningkatkan ketersediaan *bandwidth* sesuai dengan kebutuhan *client* pada divisi tersebut.

## REFERENSI

- [1]. [http://en.wikipedia.org/wiki/OSI\\_model](http://en.wikipedia.org/wiki/OSI_model).
- [2]. Syahrizal, Melwin. "Pengantar Jaringan Komputer", ANDI, Yogyakarta, 2005.
- [3]. [http://id.wikipedia.org/wiki/Warung\\_Internet](http://id.wikipedia.org/wiki/Warung_Internet).
- [4]. Bejtlich, Richard. "The Tao Of Network Security Monitoring", Addison-Wesley, USA, 2005.
- [5]. <http://www.my-proxy.com/>