

Instalasi Cache Server Dan Transparent Proxy pada FreeBSD 4.10-STABLE

by godril.at.IndoIT.net

Proxy server yang hendak dibahas di sini adalah jenis cache server, selain ada juga proxy firewall. Selama ini saya lebih banyak menginstal cache server pada platform Linux, dan kali ini saya mencoba untuk menyetup nya pada sebuah box FreeBSD. Selain itu saya juga akan mencoba untuk membuat transparent proxy dengan menggunakan [ipfw](#). Apa yang saya lakukan adalah semata-mata basic belaka, dan saya belum begitu advanced dalam hal ini dan saya akan berterimakasih bila kawan-kawan dapat memberikan masukan ke [saya](#).

Kebutuhan

Dalam percobaan kali ini saya menggunakan sebuah server bermesin Intel Pentium II 350 MHz, Memory PC-100 128 MB, dan harddisk berkapasitas 20 GB. Server ini hanya menangani 8 pc client. Versi FreeBSD yang saya gunakan adalah FreeBSD 4.10-STABLE dan menggunakan cache proxy daemon [squid-2.5.STABLE5](#). Kebutuhan optional: kacang merah Garuda + Coffeemix minus rokok (sangat di sarankan untuk menghindari rokok!!!!).

Instalasi Squid

Untuk memulai instalasi, pilihlah direktori default Anda untuk mengekstrak file kompresan squid. Untuk direktori default saya adalah /usr/local/src, dan Anda bebas memilih direktori Anda.
tar -zxvf squid-2.5.STABLE5.tar.gz

```
cd squid-2.5.STABLE5

./configure --prefix=/usr/local/squid --exec-prefix=/usr/local/squid --
enable-async-io=32 \

--enable-storeio=diskd,ufs --enable-poll --enable-delay-pools --enable-
cache- digests \

--disable-ident-lookups --enable-removal-policies

make all

make install && make clean && make distclean
```

Mengedit file konfigurasi squid.conf Dan Menyelesaikan Instalasi

Setelah kompilasi di atas, baiknya kita mencari lokasi untuk meletakkan direktori cache. Ketikkan perintah df -h dan perhatikan output yang di berikan oleh mesin Anda. Output pada mesin saya adalah sebagai berikut:

kurowo# df -h

Filesystem	Size	Used	Avail	Capacity	Mounted on
/dev/ad0s1a	126M	44M	72M	38%	/
/dev/ad0s1f	252M	4.0K	232M	0%	/tmp
/dev/ad0s1g	18G	1.9G	14G	12%	/usr
/dev/ad0s1e	252M	27M	205M	12%	/var
procfs	4.0K	4.0K	0B	100%	/proc

Dapat dilihat bahwa lokasi partisi harddisk saya terbesar dan masih kosong adalah pada direktori /usr, dan saya cd ke direktori /usr lalu membuat direktori baru yaitu cache(mkdir cache). Lalu setelah itu saya membuat group dan user baru bernama squid dan mengubah kepemilikan direktori /usr/cache

```
pw groupadd squid
pw adduser -g squid -s /nonexistent -d /dev/null
chown -R squid:squid /usr/cache
```

Perlu juga di ingat bahwa file access.log dan cache.log tidak ada secara default, oleh karena itu kita perlu membuatnya terlebih dahulu:

```
cd /usr/local/squid/var/logs
touch access.log && touch cache.log
chown squid:squid *.log
```

Lalu edit file squid.conf pada baris-baris berikut ini:

```
cache_dir diskd /usr/cache 5000 32 256 Q1=64 Q2=72
```

Artinya: direktori cache dimount dengan diskd dan lokasinya di /usr/cache. Ukuran cache dir adalah 5 GB berisi 32 direktori dan setiap direktori tersebut berisi 256 direktori lagi. Q1 menentukan jumlah request I/O yang tidak diketahui pada saat Squid berhenti membuka file baru. Jika terdapat antrean sebanyak ini, Squid tidak akan membuka file baru lagi. Defaultnya adalah 64.

Sementara Q2 menentukan jumlah pesan yang tidak diketahui pada saat Squid melakukan blocking. Jika Squid menerima antrean pesan sebanyak ini, Squid akan memblokir sementara sampai mendapatkan respon balik. Defaultnya adalah 72. Ada baiknya jika kita membiasakn mengeset ACL agar proxy kita tidak open bagi pihak di luar site kita. Misalkan block ip address di network kita adalah 192.168.0.0/24, maka kita buat ACL bagi network kita tersebut pada squid.conf. Tambahkan baris berikut ini bersama ACL lainnya:

```
acl network-kita src 192.168.0.0/255.255.255.0
acl all src 0.0.0.0/0.0.0.0
```

Lalu kita tambahkan:

```
http_access allow network-kita
http_access deny all
```

PERHATIKAN SUSUNANNYA!!! Squid membaca file konfigurasinya dari atas kebawah. Jika rule match dengan sebuah kondisi, maka rule pertama akan langsung di jalankan tanpa menginspeksi rule2 selanjutnya. Ini berarti bahwa jika kita keliru menempatkan http_access deny all sebelum http_access allow network-kita maka tak pelak lagi network kita sendiri akah terkena block. Untuk konfigurasi lengkapnya dapat di baca di file squid.conf yang disertakan dalam Squid. Untuk menyelesaikan setting jalankan perintah berikut:

```
/usr/local/squid/sbin/squid -k parse ==> untuk check syntax konfigurasi, apakah ada yang salah atau tidak.
```

```
/usr/local/squid/sbin/squid -z ==> untuk membentuk direktori cache. Pastikan tidak ada error.
```

```
/usr/local/squid/sbin/squid -DN & ==> untuk memulai squid
```

Optimalisasi Kernel

Karena kita akan mentransparent proxy, maka ada baiknya kita edit konfigurasi kernel kita. Tambahkan baris berikut ini pada konfigurasi kernel Anda untuk setting ipfw:

```
options      IPFIREWALL
options      IPFIREWALL_VERBOSE
options      IPFIREWALL_VERBOSE_LIMIT=10
options      IPFIREWALL_DEFAULT_TO_ACCEPT
options      IPDIVERT
```

Lalu untuk optimalisasi squid:

```
options      MSGMNB=32768
options      MSGMNI=82
options      MSGSEG=4098
options      MSGSSZ=128
options      MSGTQL=1024
options      SHMSEG=32
options      SHMMNI=64
options      SHMMAX=4194304
options      SHMALL=8192
```

(NB: setting saya ambil dari buku "Optimasi Akses Internet Dengan Squid" oleh Dody Marianto terbitan ElexMedia Komputindo. (Thanx Sabriena for buying me this book, Luv u always). Setelah itu kompilasi kernel Anda. Cara untuk kompilasi ulang kernel ada di sini. Tambahkan baris berikut ini pada file /etc/rc.conf.

```
gateway_enable="YES"
firewall_enable="YES"
firewall_type="OPEN"
natd_enable="YES"
natd_enable="wi0" # sesuaikan dengan interface luar Anda.
```

Lalu reboot.

Setting Transparent Proxy Dengan ipfw

Karena contoh yang saya ambil adalah server saya, maka setting nya adalah sebagai berikut (sesuaikan dengan server Anda) Interface luar saya adalah wi0 dan interface dalam adalah r10 dengan ip address 192.168.0.254, maka settingnya adalah:

```
ipfw add 100 allow tcp from 192.168.0.254 to any 80
ipfw add 101 fwd 192.168.0.254,8080 tcp from any to any 80 via r10
```

Buatlah sebuah file executable dengan ekstensi .sh di dalam direktori /usr/local/etc/rc.d agar setiap reboot Anda mendapatkan kembali setting ipfw Anda seperti itu. Nah.. selesai sudah. Selamat, Anda punya cache engine FreeBSD baru!!!