

Setting Bandwidth Management di Linux

Oleh : Ripto Mukti Wibowo
(rip.wibowo@gmail.com)

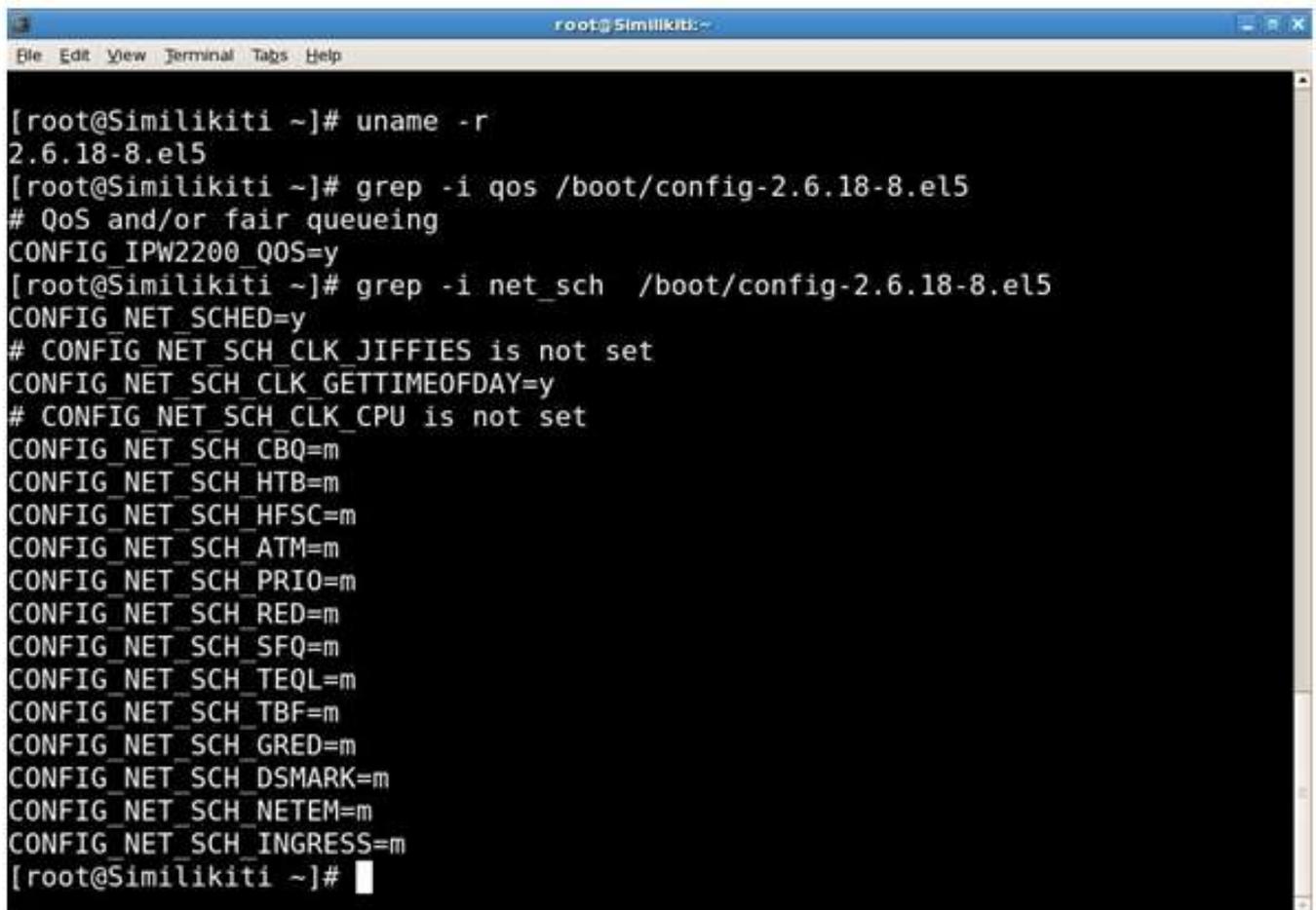
Pendahuluan

Bandwidth management sering dipertukarkan dengan istilah traffic control, yang dapat didefinisikan sebagai pengalokasian yang tepat dari suatu bandwidth untuk mendukung kebutuhan atau keperluan aplikasi atau suatu layanan jaringan. Istilah bandwidth dapat didefinisikan sebagai kapasitas atau daya tampung suatu channel komunikasi (medium komunikasi) untuk dapat dilewati sejumlah traffic informasi atau data dalam satuan waktu tertentu. Umumnya bandwidth dihitung dalam satuan bit, kbit atau bps (byte per second). Pengalokasian bandwidth yang tepat dapat menjadi salah satu metode dalam memberikan jaminan kualitas suatu layanan jaringan (QoS = Quality Of Services).

Bandwidth Management di Linux adalah bagaimana kita menerapkan pengalokasian atau pengaturan bandwidth dengan menggunakan sebuah komputer linux. Umumnya komputer linux dapat digunakan sebagai gateway/router sehingga memungkinkan untuk mengatur traffic data atau mengalokasikan bandwidth dari traffic data yang melewati komputer linux tersebut untuk memberikan jaminan kualitas akses layanan internet bagi komputer-komputer dalam jaringan lokal. Dalam artikel ini penulis tidak akan membahas secara panjang lebar mengenai bagaimana cara kerja kernel linux dalam melaksanakan fungsi traffic control atau bandwidth management, penulis hanya akan menjelaskan langkah-langkah praktis membangun sebuah bandwidth management dengan sistem operasi linux beserta tools administrasi yang mudah diperoleh dan free. Untuk mengetahui dengan detail bagaimana cara kerja traffic control pada kernel linux sebaiknya Anda membaca guide atau tutorial tentang Linux Advanced Routing & Traffic Control dari website <http://lartc.org> atau tutorial lainnya di internet.

QoS pada Kernel Linux

Umumnya kernel linux sudah dilengkapi dengan dukungan fungsi implementasi QoS (Quality Of Services) atau traffic control, dengan berbagai modul network scheduler. Untuk membuktikan bahwa kernel linux menyediakan dukungan QoS beserta modul-modul Queueing Disciplines Anda dapat mengeceknya dengan melihat konfigurasi kernel yang Anda gunakan saat ini, seperti dalam gambar-1.

A terminal window titled 'root@Similikiti:~' showing the execution of several commands to check kernel configuration. The first command is 'uname -r', which returns '2.6.18-8.el5'. The second command is 'grep -i qos /boot/config-2.6.18-8.el5', which returns '# QoS and/or fair queueing' and 'CONFIG_IPW2200_QOS=y'. The third command is 'grep -i net_sch /boot/config-2.6.18-8.el5', which returns a list of network scheduler configurations: 'CONFIG_NET_SCHED=y', '# CONFIG_NET_SCHED_CLK_JIFFIES is not set', 'CONFIG_NET_SCHED_CLK_GETTIMEOFDAY=y', '# CONFIG_NET_SCHED_CLK_CPU is not set', 'CONFIG_NET_SCHED_CBQ=m', 'CONFIG_NET_SCHED_HTB=m', 'CONFIG_NET_SCHED_HFSC=m', 'CONFIG_NET_SCHED_ATM=m', 'CONFIG_NET_SCHED_PRI0=m', 'CONFIG_NET_SCHED_RED=m', 'CONFIG_NET_SCHED_SFQ=m', 'CONFIG_NET_SCHED_TEQL=m', 'CONFIG_NET_SCHED_TBF=m', 'CONFIG_NET_SCHED_GRED=m', 'CONFIG_NET_SCHED_DSMARK=m', 'CONFIG_NET_SCHED_NETEM=m', and 'CONFIG_NET_SCHED_INGRESS=m'. The prompt returns to '[root@Similikiti ~]#'.

```
[root@Similikiti ~]# uname -r
2.6.18-8.el5
[root@Similikiti ~]# grep -i qos /boot/config-2.6.18-8.el5
# QoS and/or fair queueing
CONFIG_IPW2200_QOS=y
[root@Similikiti ~]# grep -i net_sch /boot/config-2.6.18-8.el5
CONFIG_NET_SCHED=y
# CONFIG_NET_SCHED_CLK_JIFFIES is not set
CONFIG_NET_SCHED_CLK_GETTIMEOFDAY=y
# CONFIG_NET_SCHED_CLK_CPU is not set
CONFIG_NET_SCHED_CBQ=m
CONFIG_NET_SCHED_HTB=m
CONFIG_NET_SCHED_HFSC=m
CONFIG_NET_SCHED_ATM=m
CONFIG_NET_SCHED_PRI0=m
CONFIG_NET_SCHED_RED=m
CONFIG_NET_SCHED_SFQ=m
CONFIG_NET_SCHED_TEQL=m
CONFIG_NET_SCHED_TBF=m
CONFIG_NET_SCHED_GRED=m
CONFIG_NET_SCHED_DSMARK=m
CONFIG_NET_SCHED_NETEM=m
CONFIG_NET_SCHED_INGRESS=m
[root@Similikiti ~]#
```

Gambar-1. Pengecekan dukungan kernel linux terhadap fungsi QoS

Jika hasil pengecekan seperti tampak pada gambar-1, maka dapat dikatakan kernel linux Anda saat ini sudah siap untuk implementasi traffic control.

Persiapan dan kebutuhan software

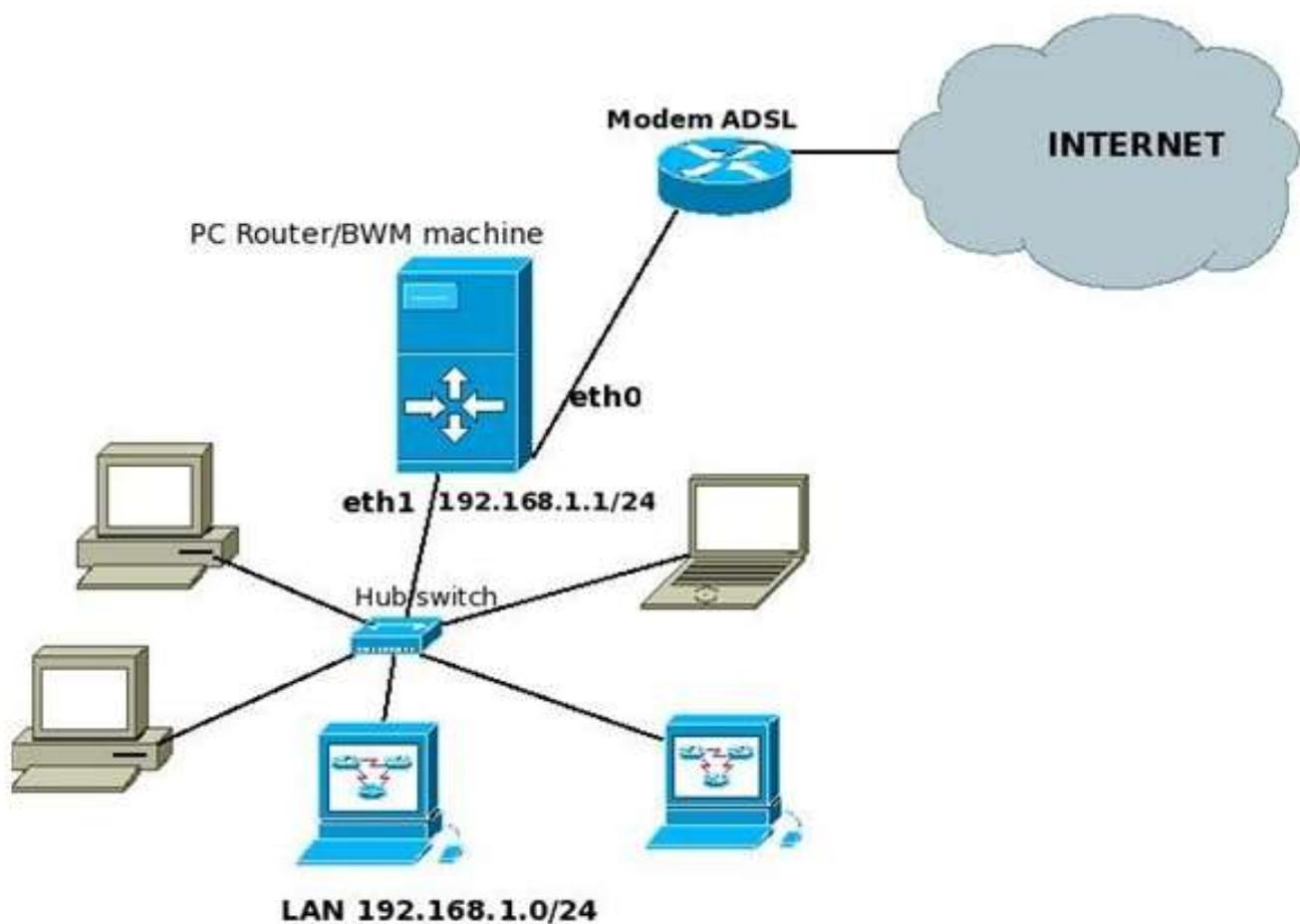
Penulis dalam contoh artikel ini menggunakan kernel 2.6.18-8.el5 dengan distribusinya menggunakan CentOS 5. Dalam artikel ini penulis menggunakan modul network scheduler atau Queueing Disciplines SFQ (Stochastic Fairness Queueing) dan HTB (Hierarchical Token Bucket) untuk metode antrian traffic data dalam kernel linux dan filtering u32 untuk memfilter kelas-kelas paket. Penulis juga menggunakan htb.init scripts (<http://sourceforge.net/projects/htbinit/>) untuk men-setup traffic control berbasiskan HTB. Untuk memudahkan konfigurasi htb.init penulis menggunakan webmin (<http://prdownloads.sourceforge.net/webadmin/webmin-1.390-1.noarch.rpm>) front-end QoS yang berbasis HTB (<http://www.sehier.fr/webmin-htb/webmin-htb.tar.gz>) yang sebelumnya harus Anda instal kedalam webmin sebagai modul. Selanjutnya Anda juga memerlukan tool untuk management bandwidth yang disertakan oleh paket software iproute2 (ip dan tc) untuk itu pastikan bahwa software iproute2 ini telah terinstal, sebaiknya Anda cek dengan perintah berikut:

```
[root@Similikiti ~]# rpm -qa|grep iproute  
iproute-2.6.18-4.el5
```

Jika belum terinstal Anda dapat segera menginstalnya, umumnya pakete software iproute2 disertakan dalam CD/DVD distro yang Anda gunakan, jadi tidak perlu mendownloadnya dari Internet.

Skenario jaringan

Agar lebih mudah dipahami maka penulis membuat sebuah skenario jaringan seperti tampak dalam gambar-2.



Gambar-2. Skema jaringan

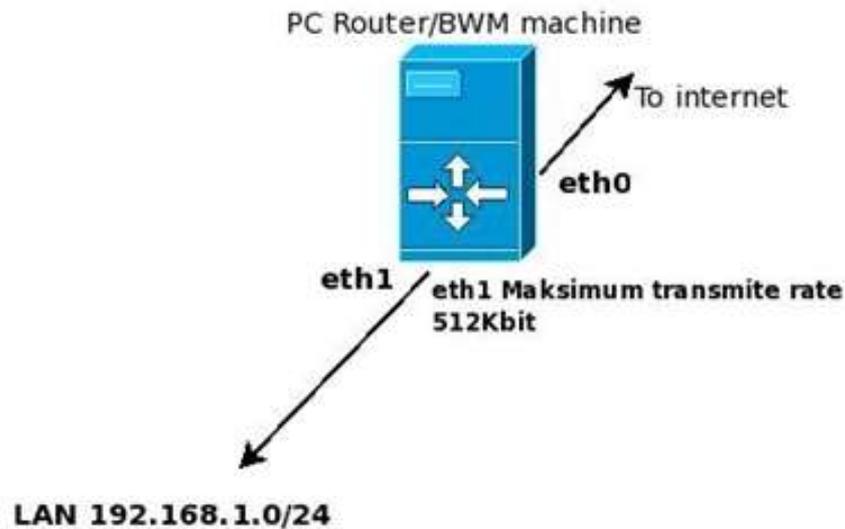
Sebagaimana tampak pada gambar-2, bahwa komputer linux yang berfungsi sebagai PC Router (gateway) juga difungsikan sebagai bandwidth management untuk mengatur alokasi bandwidth dalam LAN. Yang perlu diperhatikan adalah bahwa kita hanya akan mengatur atau mengalokasikan bandwidth dari traffic data yang dikirim oleh network interface pada komputer linux PC Router dan

tidak mengatur bandwidth traffic data yang datang dari jaringan ke network interface pada linux PC Router (alasan nya coba Anda baca tutorial dari lartc.org) . Dan dalam contoh ini penulis akan mencontohkan bagaimana kita dapat mengatur bandwidth dari traffic data yang dikirimkan/keluar ke komputer-komputer jaringan LAN, sehingga dengan kata lain kita hanya akan mengatur bandwidth traffic data yang datang dari Internet menuju komputer-komputer pada jaringan lokal (LAN).

Sebagaimana dalam gambar-2, maka kita hanya akan menerapkan traffic control pada network interface eth1 dari komputer PC Router. Namun sebaiknya Anda juga menerapkan traffic control pada interface eth0 yang dapat bermanfaat dalam mengatur traffic data yang dikirimkan dari eth0

menuju jaringan Internet sehingga harapannya menghindari terjadinya bottleneck pada Modem ADSL. Tetapi untuk mempermudah pemahaman maka penulis membatasi hanya membahas bagaimana mengatur alokasi bandwidth dari traffic yang akan dikirimkan ke LAN dari PC Router ataupun dari Internet. Untuk itu berikut ini penulis menggambarkan skenario maksimum rate yang dapat ditransmit dari eth1 pada PC router sebesar 512Kbit seperti tampak dalam gambar-3, dan selanjutnya akan dibagi-bagi kedalam kelas-kelas aliran traffic data. Pengaturab akan dilakukan via

webmin.



Gambar-3. Skenario maksimum transmit rate

Instalasi dan konfigurasi software

Ok, agar lebih cepat maka penulis mengasumsikan bahwa software dan kebutuhan yang lain telah terinstal dan dikonfigurasi dengan benar, kecuali software webmin, kemudian modul webmin-htb, serta htb.init yang belum terinstal. Untuk itu mari kita mulai dengan menginstal webmin.

Instalasi webmin

Download terlebih dahulu webmin :

```
# wget http://prdownloads.sourceforge.net/webadmin/webmin-1.390-1.noarch.rpm
```

Kemudian instal seperti berikut ini:

```
[root@Similikiti ~]# rpm -ivh webmin-1.390-1.noarch.rpm
```

Selanjutnya aktifkan webmin jika belum aktif.

```
[root@Similikiti ~]# service webmin start
```

atau

```
[root@Similikiti ~]# /etc/init.d/webmin start
```

Kemudian Anda dapat mengakses webmin dengan menggunakan web client (misal: firefox), dengan URL <http://localhost:10000> atau jika mendukung secure connection URL nya adalah <https://localhost:10000>.

Instalasi htb.init script

Download dahulu htb.init script dengan menggunakan web client (misal:firefox) dan ketik URL

<http://http://sourceforge.net/projects/htbinit/>. Kemudian instal dan setup sebagai berikut:

```
[root@Similikiti ~]# cp htb.init-v0.8.5 /etc/init.d/htb.init
```

```
[root@Similikiti ~]# mkdir -p /etc/sysconfig/htb
```

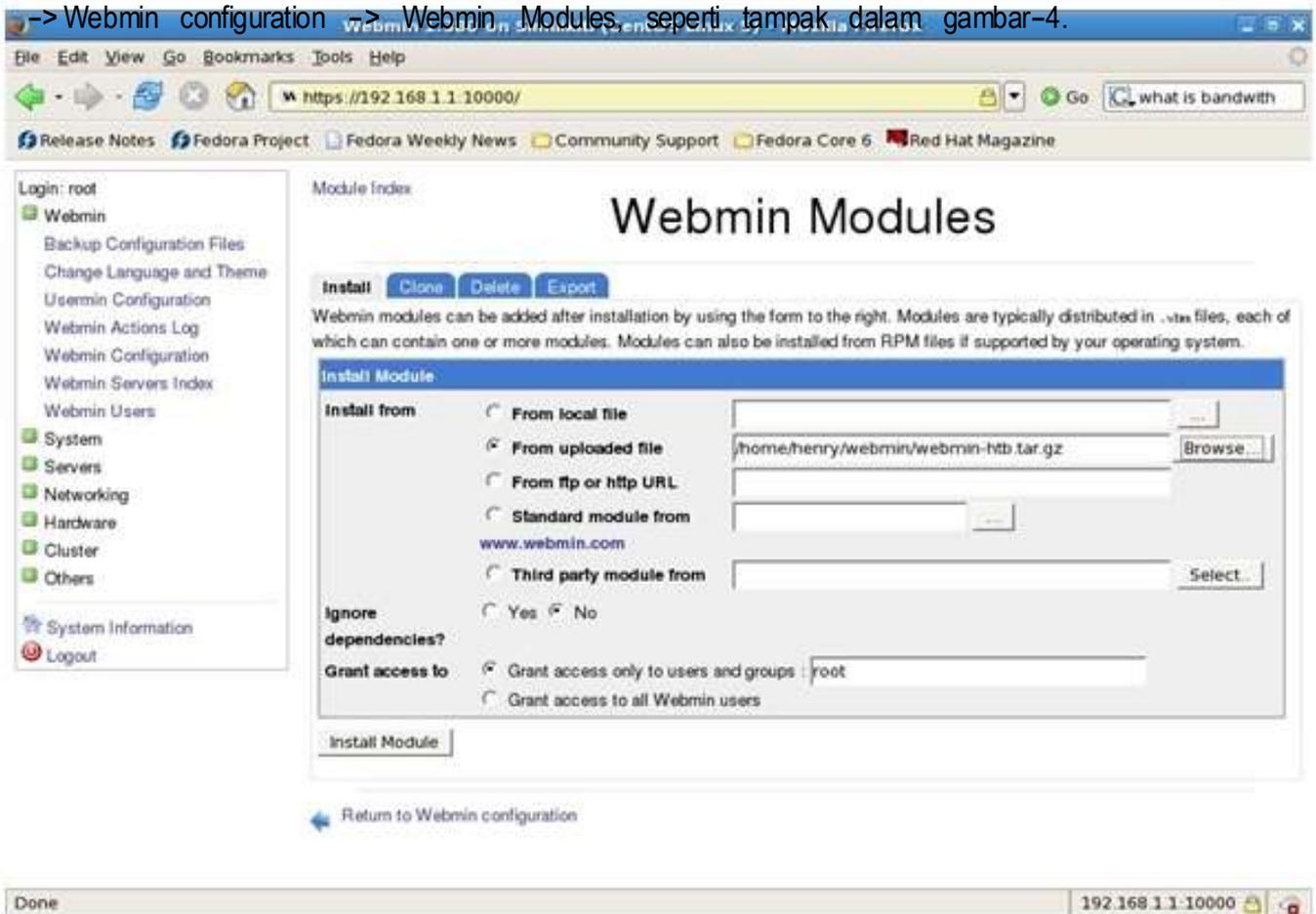
Instalasi webmin-htb modul

Download terlebih dahulu webmin-htb modul :

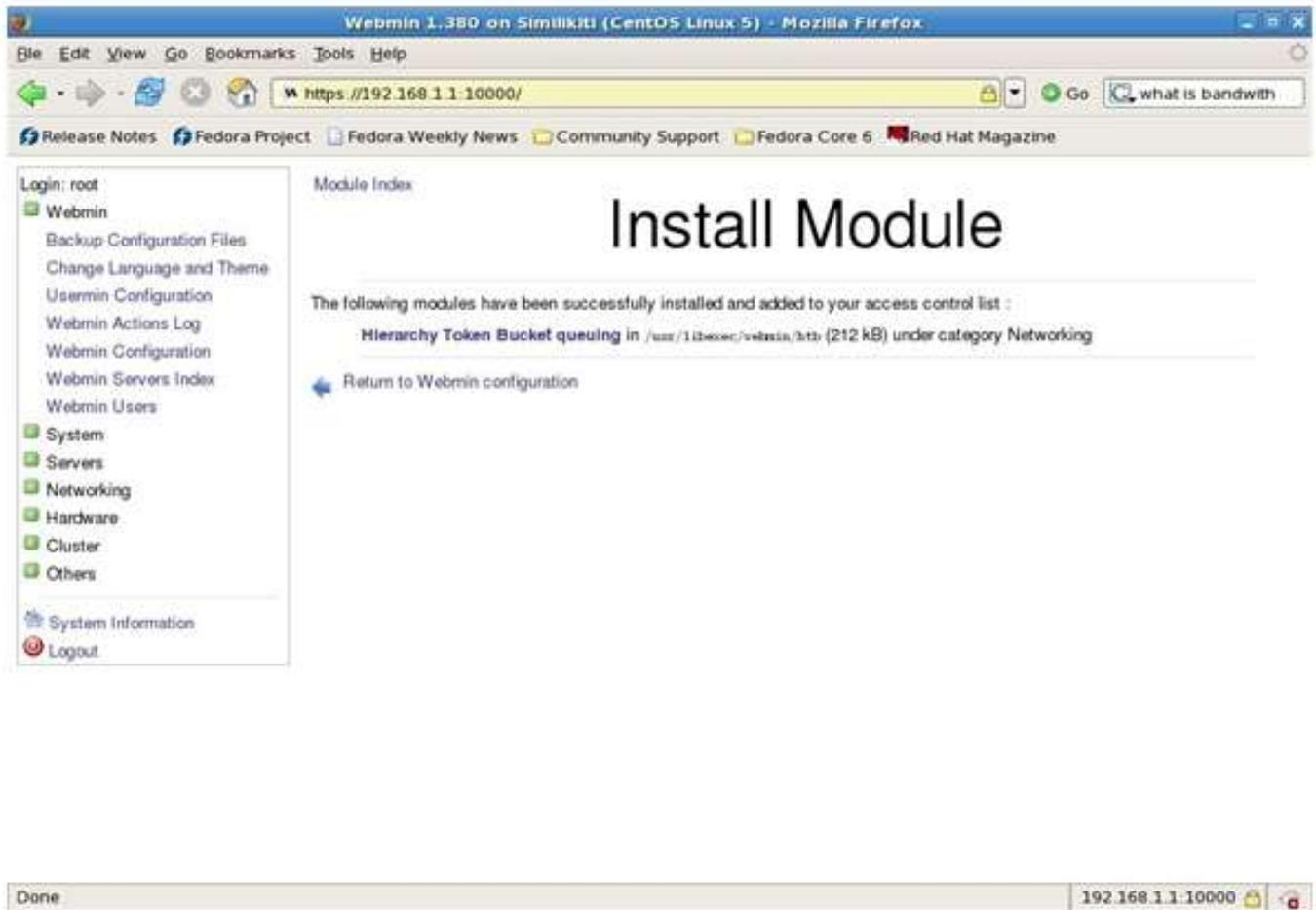
```
# wget http://www.sehier.fr/webmin-htb/webmin-htb.tar.gz
```

Kemudian instal modul webmin-htb tersebut melalui webmin, dengan memilih menu Webmin

-> Webmin configuration -> Webmin Modules, seperti tampak dalam gambar-4.



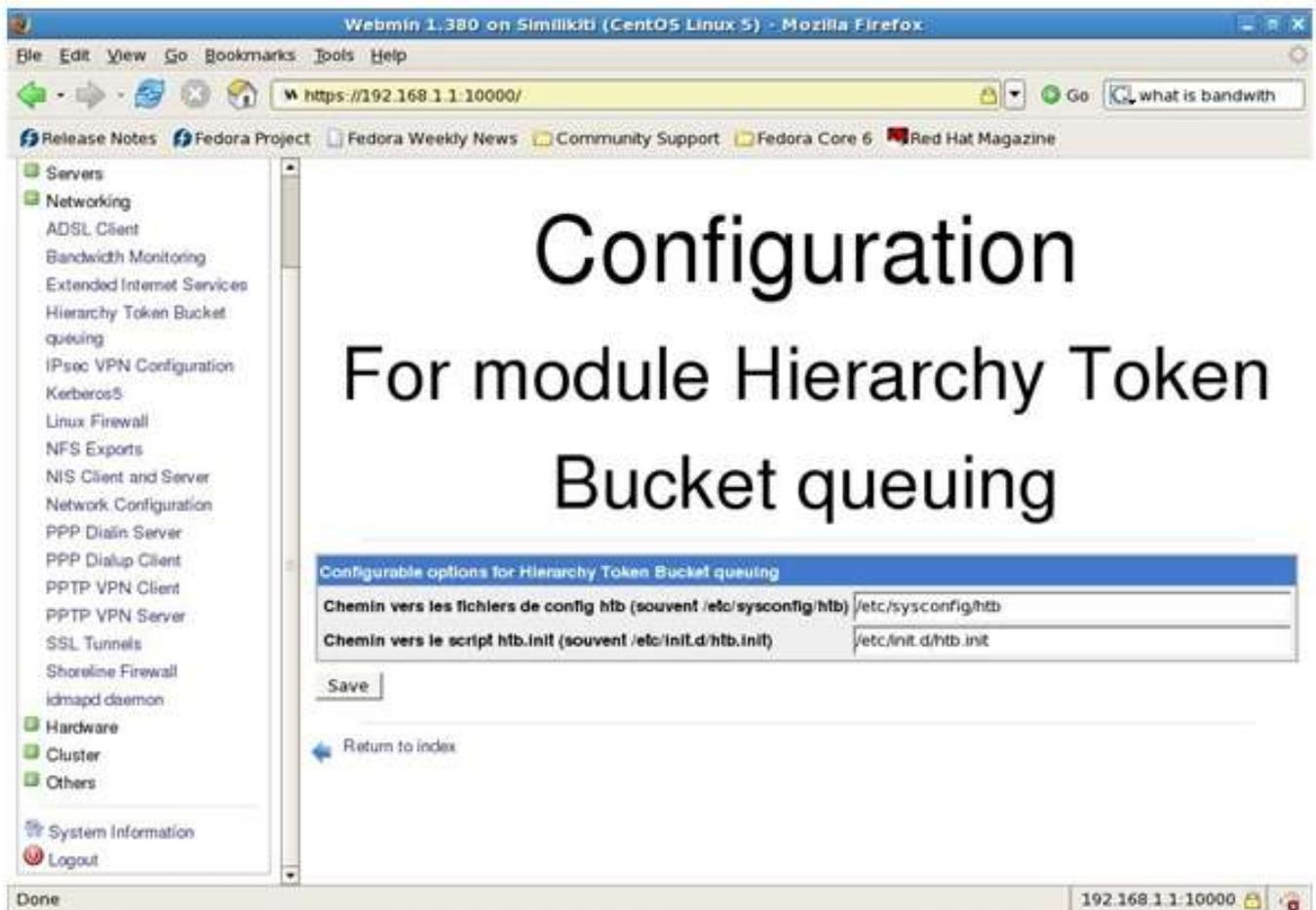
Gambar-4. Instalasi modul HTB via webmin modules



Gambar-5. Halaman selesai instal module HTB

Selanjutnya Anda harus mengkonfigurasi dahulu modul webmin-htb dengan mengklik hyperlink Hierarchical Token Bucket yang tampak pada gambar-5 setelah instalasi modul HTB. Dan selanjutnya Anda akan menemui halaman seperti gambar-6. Sebagai catatan sebaiknya Anda pastikan dahulu apakah sudah terinstal modul perl Tree::DAG_Node, jika belum Anda dapat menginstalnya terlebih dahulu sebelum melakukan konfigurasi modul webmin-HTB, seperti berikut ini:

```
[root@labtop1 ~]# cpan -i Tree::DAG_Node
```



Gambar-6. Halaman konfigurasi modul HTB .

Setup traffic control menggunakan front-end webmin-htb modul

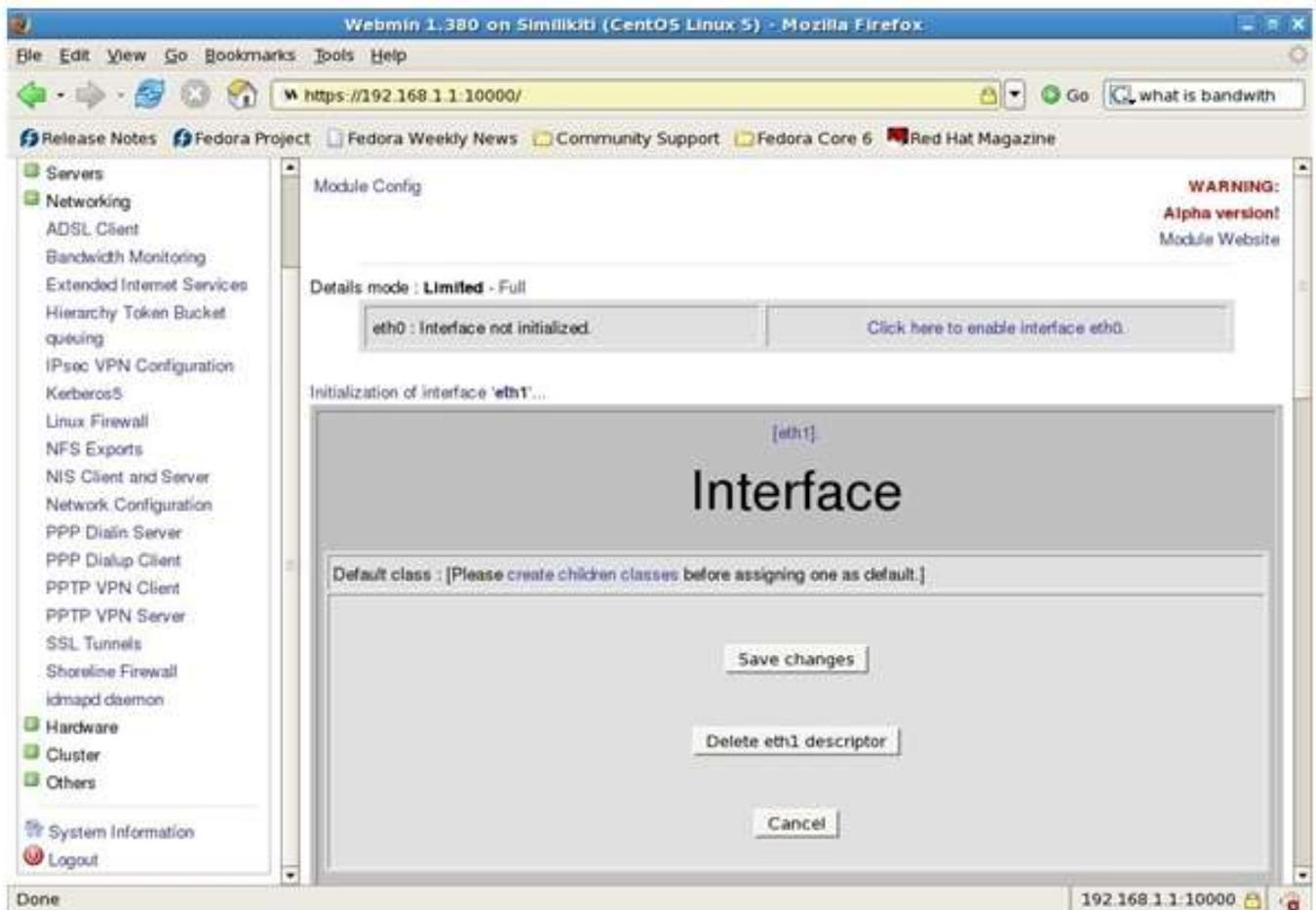
Sebelum memulai mensetup traffic control menggunakan front-end webmin-htb melalui webmin,

penulis akan membuat skenario pengalokasian bandwidth dan filteringnya, yakni sebagai berikut:

- Bandwidth Traffic data akan disetup maksimum hanya 512Kbit yang dapat ditransmit dari eth1 ke jaringan.
- Jumlah kelas traffic data yang akan disetup ada 3 kelas yang terdiri dari:
 - kelas yang pertama dengan informasi parameter sbb:
 - Name : http
 - Rate: 400Kbit
 - Ceil: 512Kbit

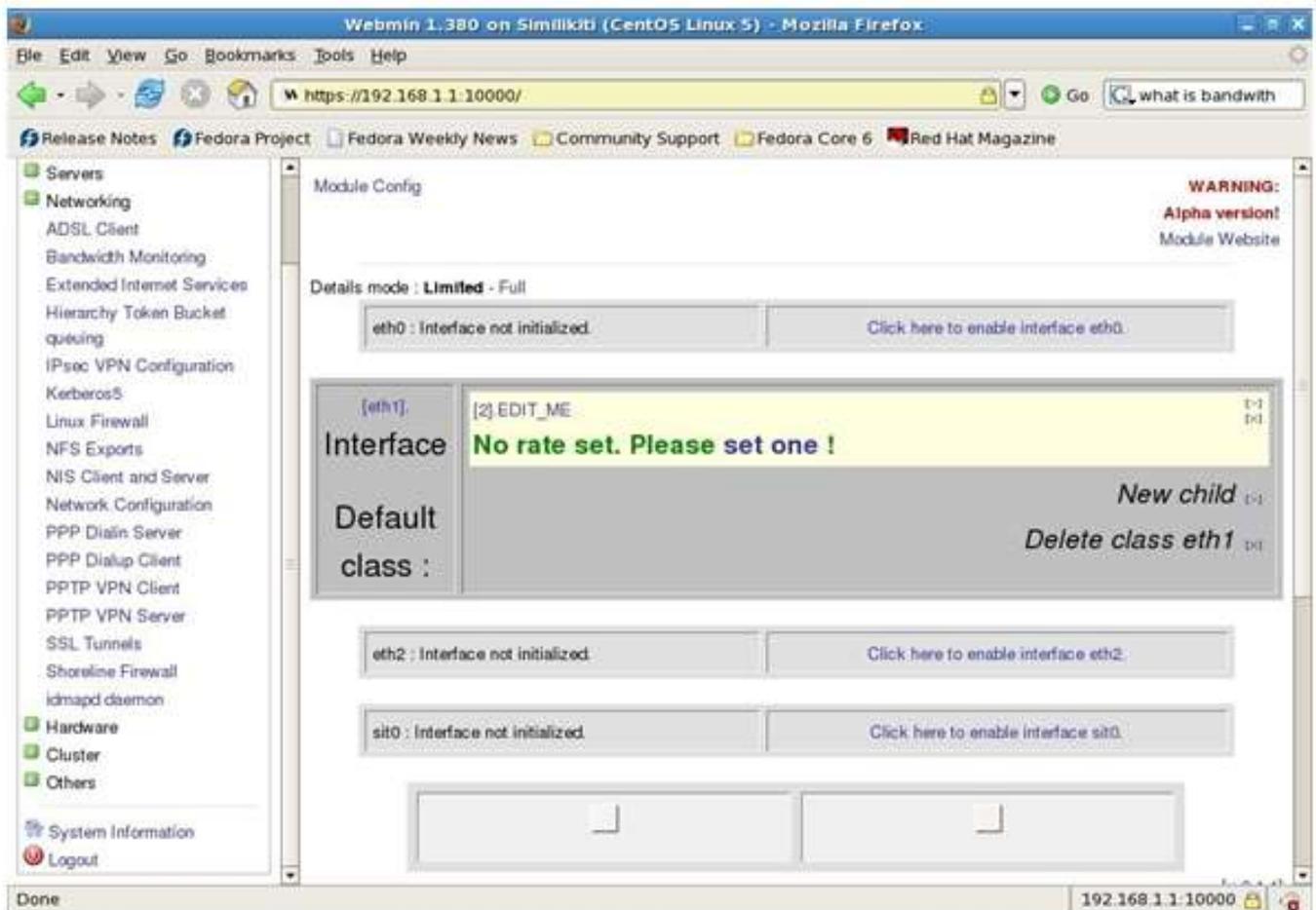
- Priority: 1
- Dengan rules filtering sbb:
 - Source address 0.0.0.0/0 (dari manapun) dengan port asal(source port) adalah 80,443 dan 3128, dengan destination ke jaringan 192.168.1.0/24
- Kelas yang kedua dengan informasi parametersbb:
 - Name: ssh
 - Rate: 80 Kbit
 - Ceil: 112Kbit
 - Priority: 2
 - Dengan rules filtering sbb:
 - Source address 0.0.0.0/0 (dari manapun) dengan port asal(source port) adalah 22, dengan destination ke host 192.168.1.251.
- Kelas yang ketiga dengan informasi parameter sbb:
 - Name: other
 - Rate: 32Kbit
 - Ceil: 32Kbit
 - Priority: 3
 - Dengan rules filtering sbb:
 - Source address 0.0.0.0/0 (dari manapun) dengan port asal(source port) berapapun dan dengan destination ke host manapun dalam jaringan LAN.

Sekarang Anda tinggal mencoba menerapkan skenario tersebut dengan mengakses webmin (<http://localhost:10000> atau <https://localhost:10000>), kemudian login masukkan username 'root' dan berikan password root, maka selanjutnya Anda akan masuk kehalaman utama webmin, Pada halaman utama tersebut Anda lihat menu sebelah kiri (jika webmin menggunakan default themes) pilihlah menu Networking ➤ **Hierarchy Token Bucket queuing** dan Anda akan menemui halaman utama modul HTB kemudian klik hyperlink “ Click here to enable interface eth1” maka selanjutnya akan tampak halaman inisialisasi seperti tampak pada gambar 7, dan tekan tombol Save changes.



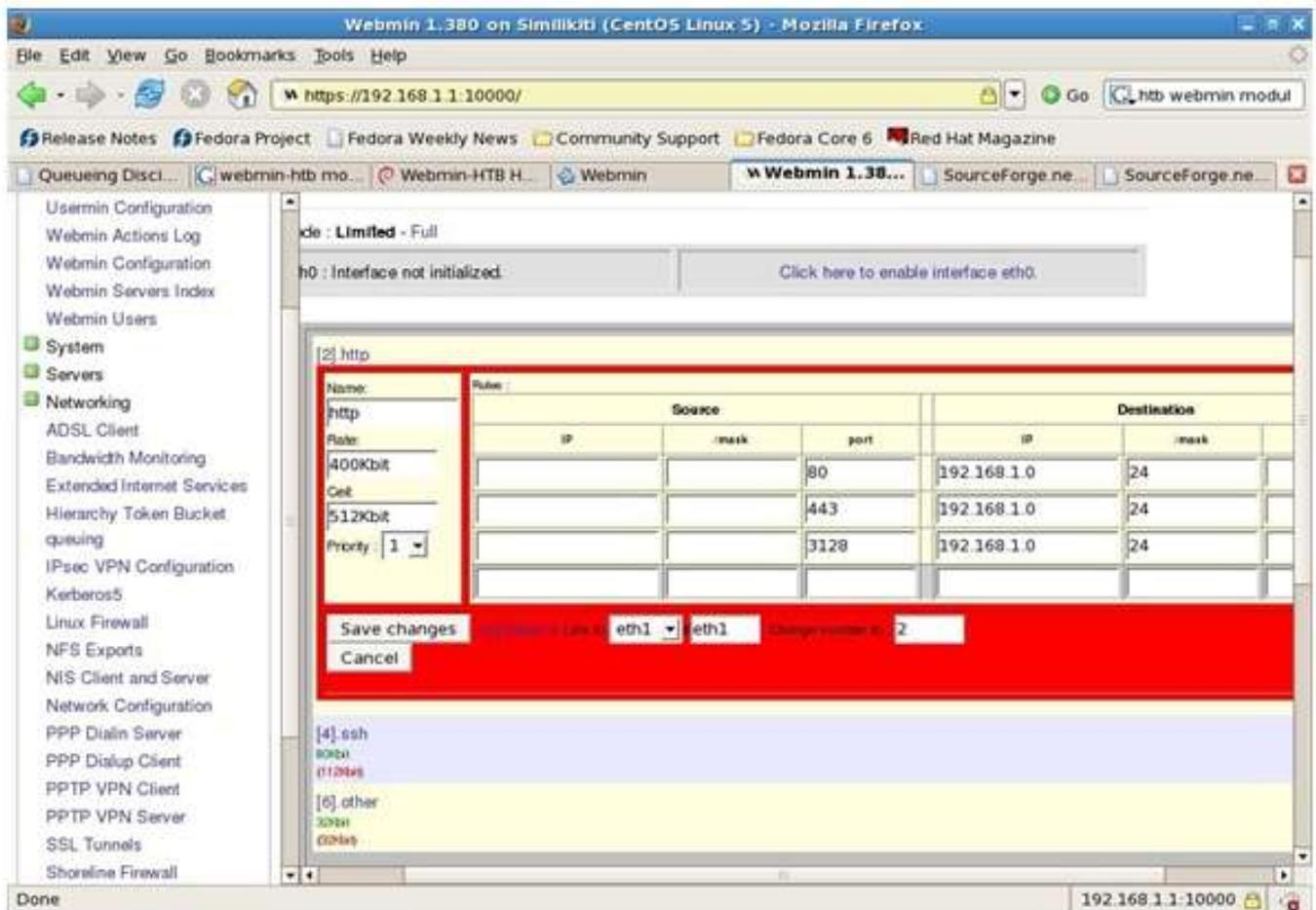
Gambar 7. Halaman inialisasi interface eth1

Langkah berikutnya adalah Anda membuat atau mendefinisikan masing-masing kelas sebagaimana skenario yang telah dipaparkan sebelumnya. Untuk membuat kelas lihat pada halaman konfigurasi module HTB, pada bagian Interface eth1, kliklah menu/hyperlink "New child [>]". Kemudian akan muncul halaman seperti pada gambar 8.



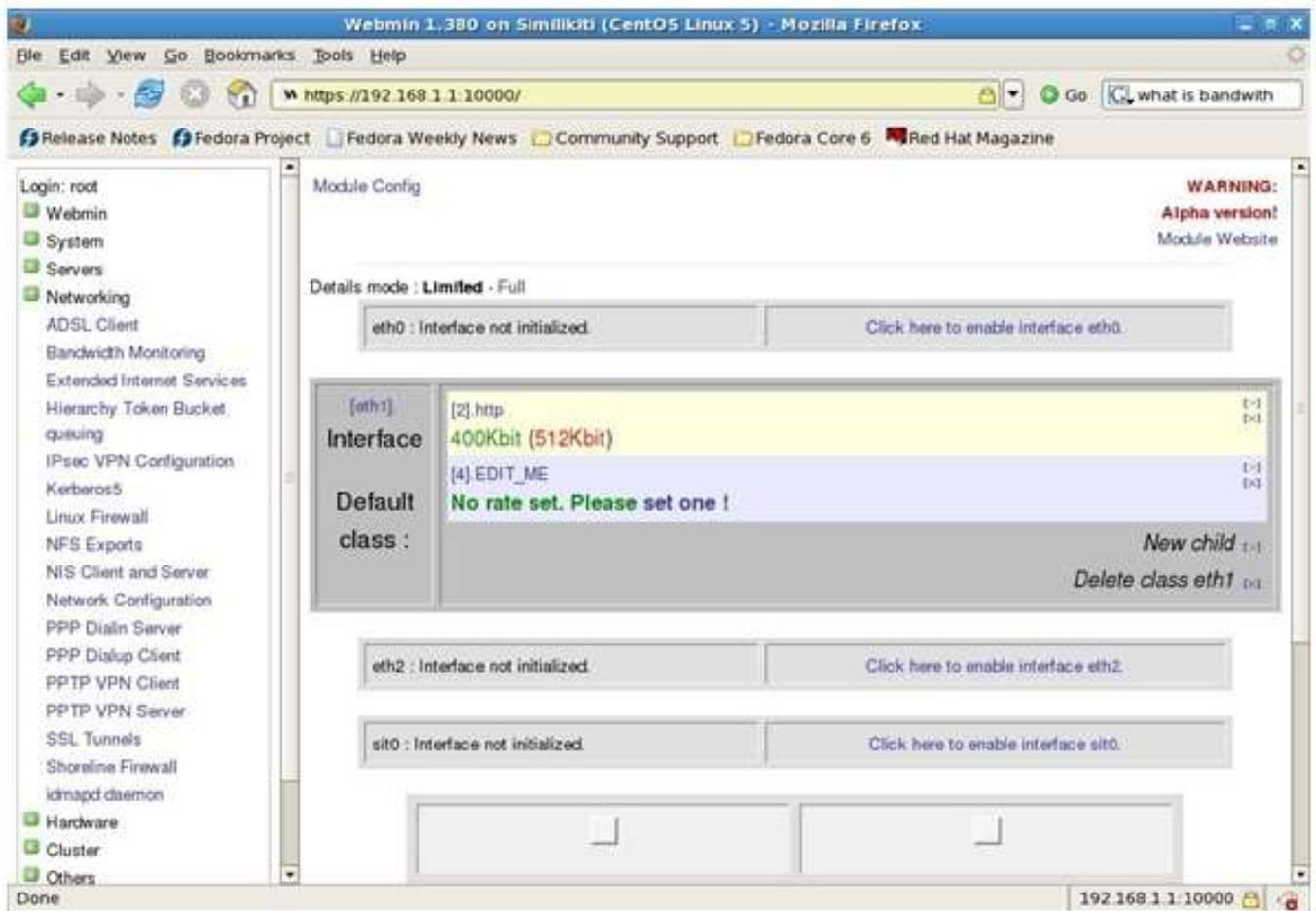
Gambar 8. Halaman Create New Child pertama

Berikutnya edit kelas yang baru Anda buat dengan mengklik menu/hyperlink “ [2]EDIT_ME” , dan anda akan menjumpai halaman edit kemudian isilah seluruh elemen form yang ada sesuai dengan skenario yang telah dijelaskan untuk kelas yang pertama. Lihat gambar 9 untuk kelas yang pertama.

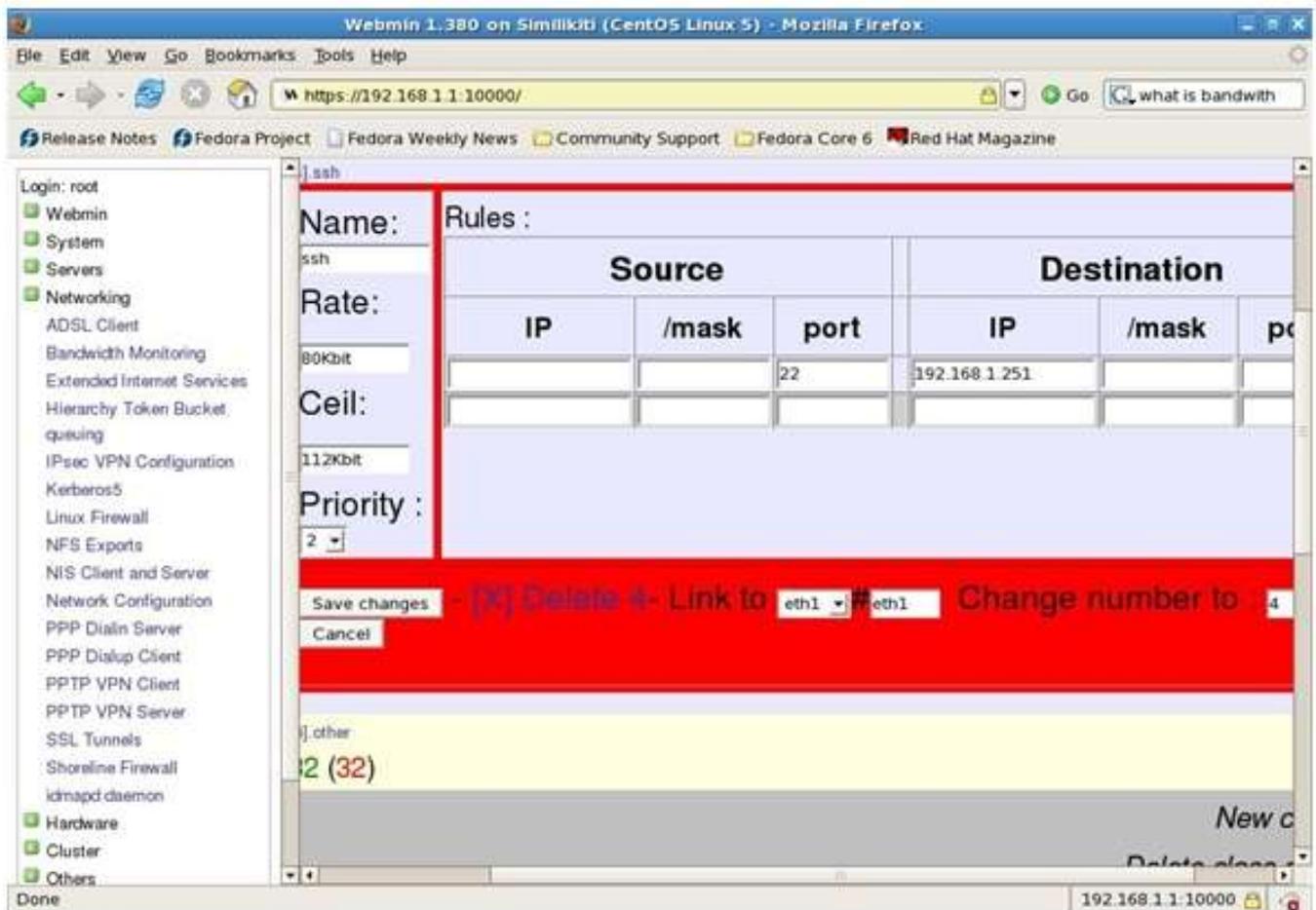


Gambar 9. Halaman Edit Me, untuk pembuatan kelas pertama.

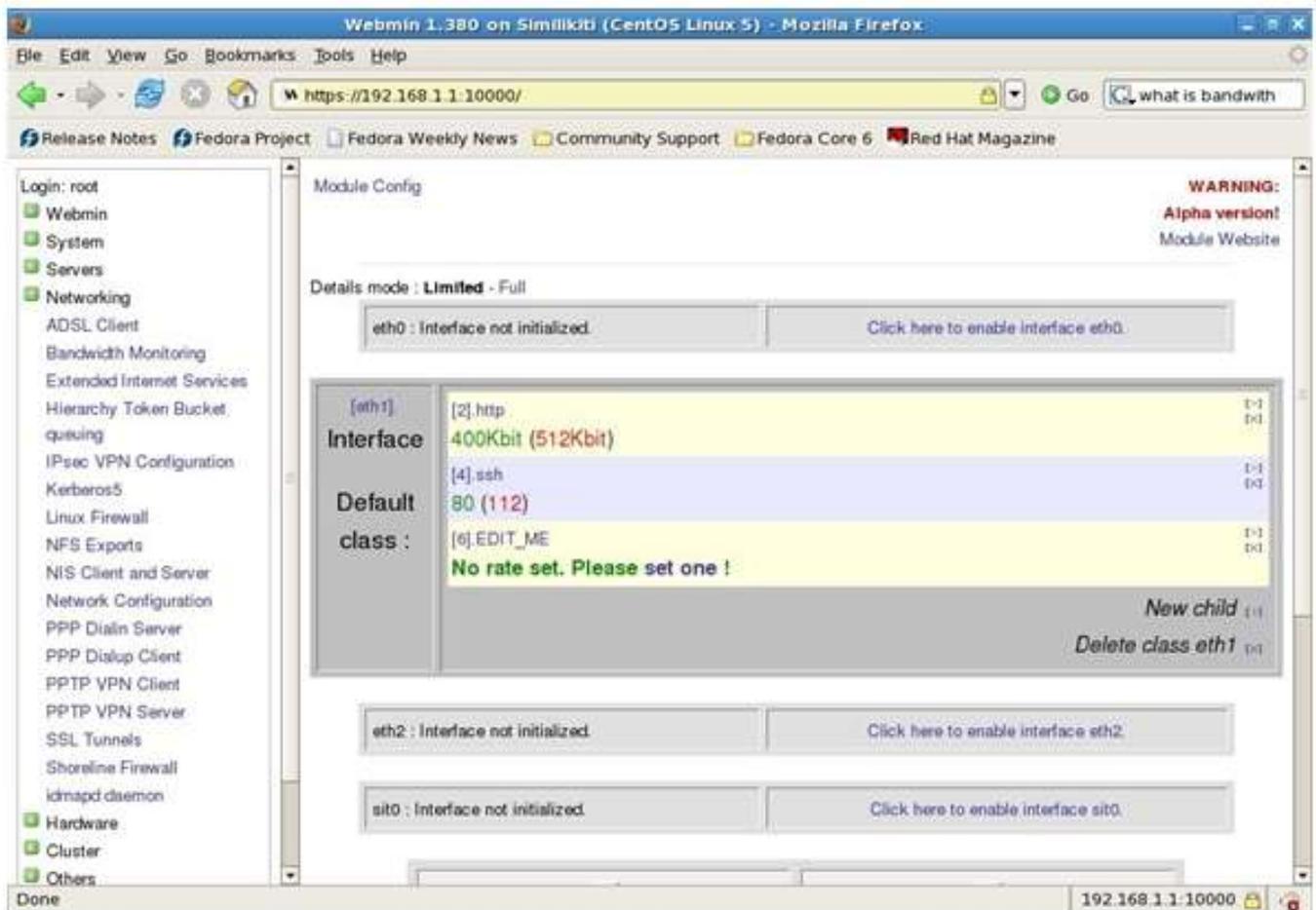
Setelah mengedit/membuat kelas traffic control yang pertama beserta rule filteringnya, tekanlah tombol " Save changes" . Langkah pembuatan kelas yang kedua dan ketiga mengikuti langkah-langkah yang telah Anda lakukan untuk kelas yang pertama, hanya saja sesuaikan skenarionya. Berikut ini gambar-gambar yang menjelaskan langkah-langkah pembuatan kelas kedua dan ketiga beserta rule filternya. Catatan jika anda ingin membuat satu kelas lagi maka pastikan total rate nya untuk semua kelas sama dengan maksimum rate yang anda inginkan. Anda juga dapat membuat child kelas dalam sebuah kelas yang telah didefinisikan untuk membentuk hirarki kelas (lihat menu/hyperlink newcid yang ada pada bagian kanan setiap kelas yang ditandai dengan tanda " [>]" .



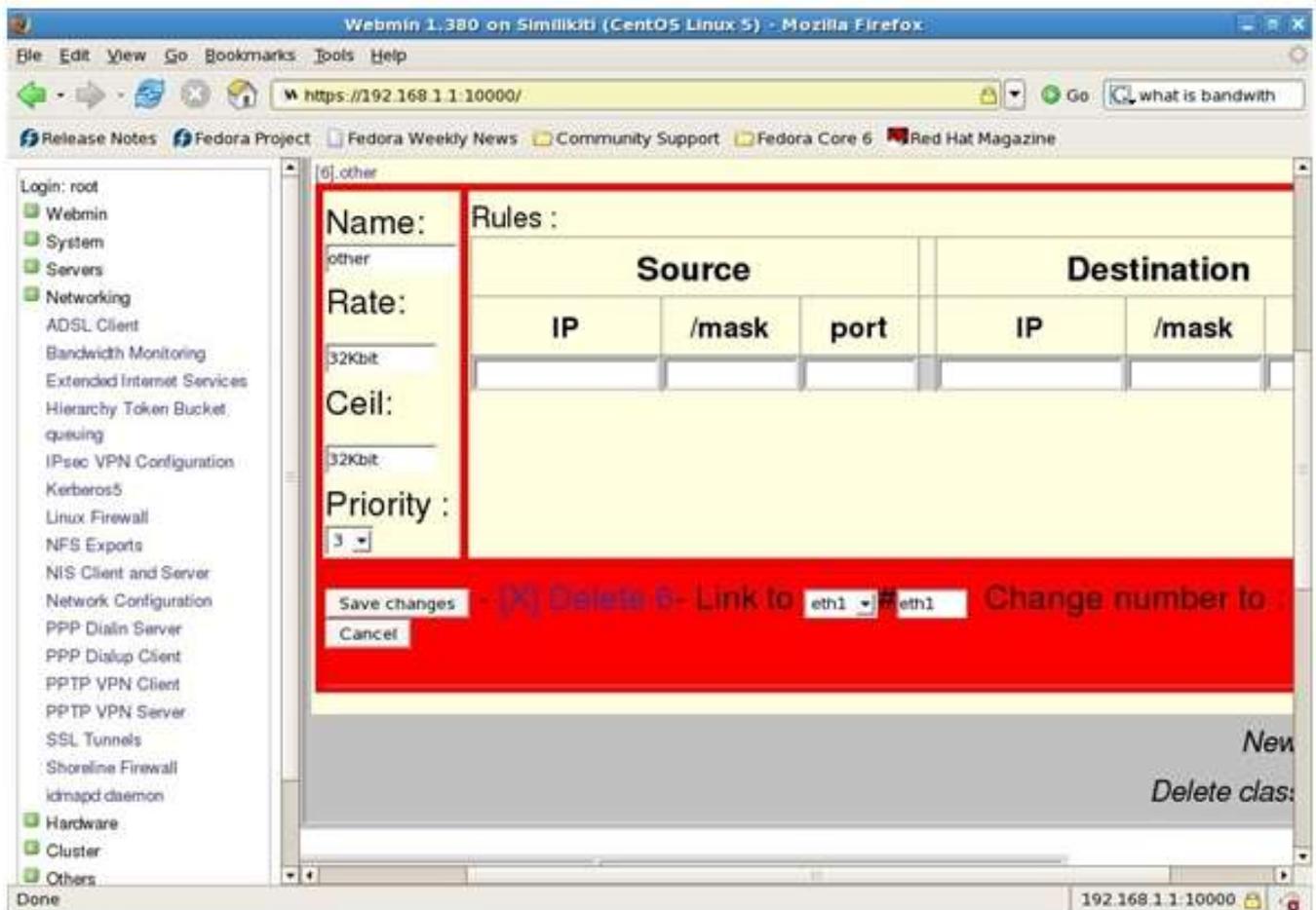
Gambar 10. Halaman Create New Child kedua



Gambar 1. Halaman Edit Me, untuk pembuatan kelas kedua

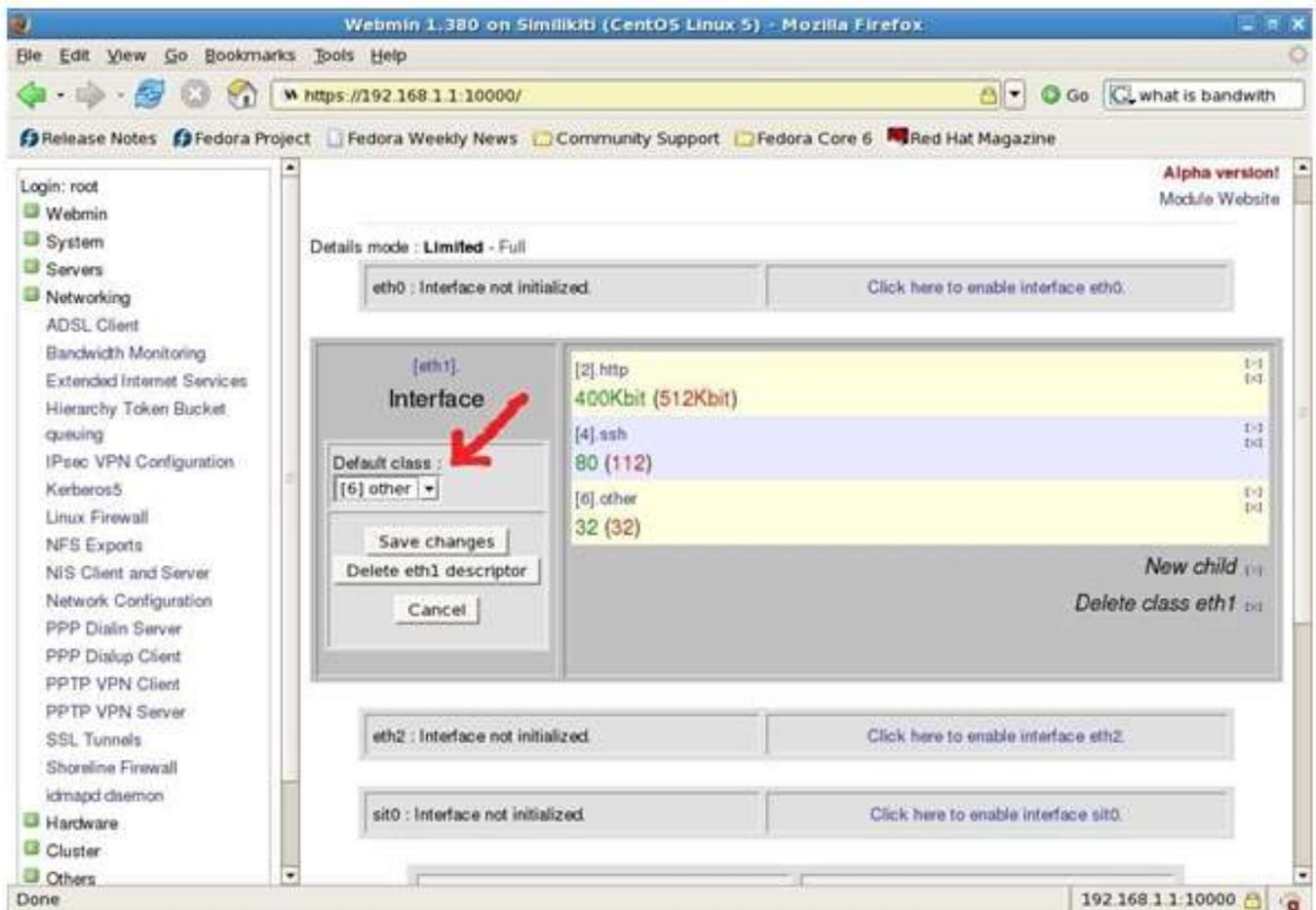


Gambar 12. Halaman Create New Child ketiga



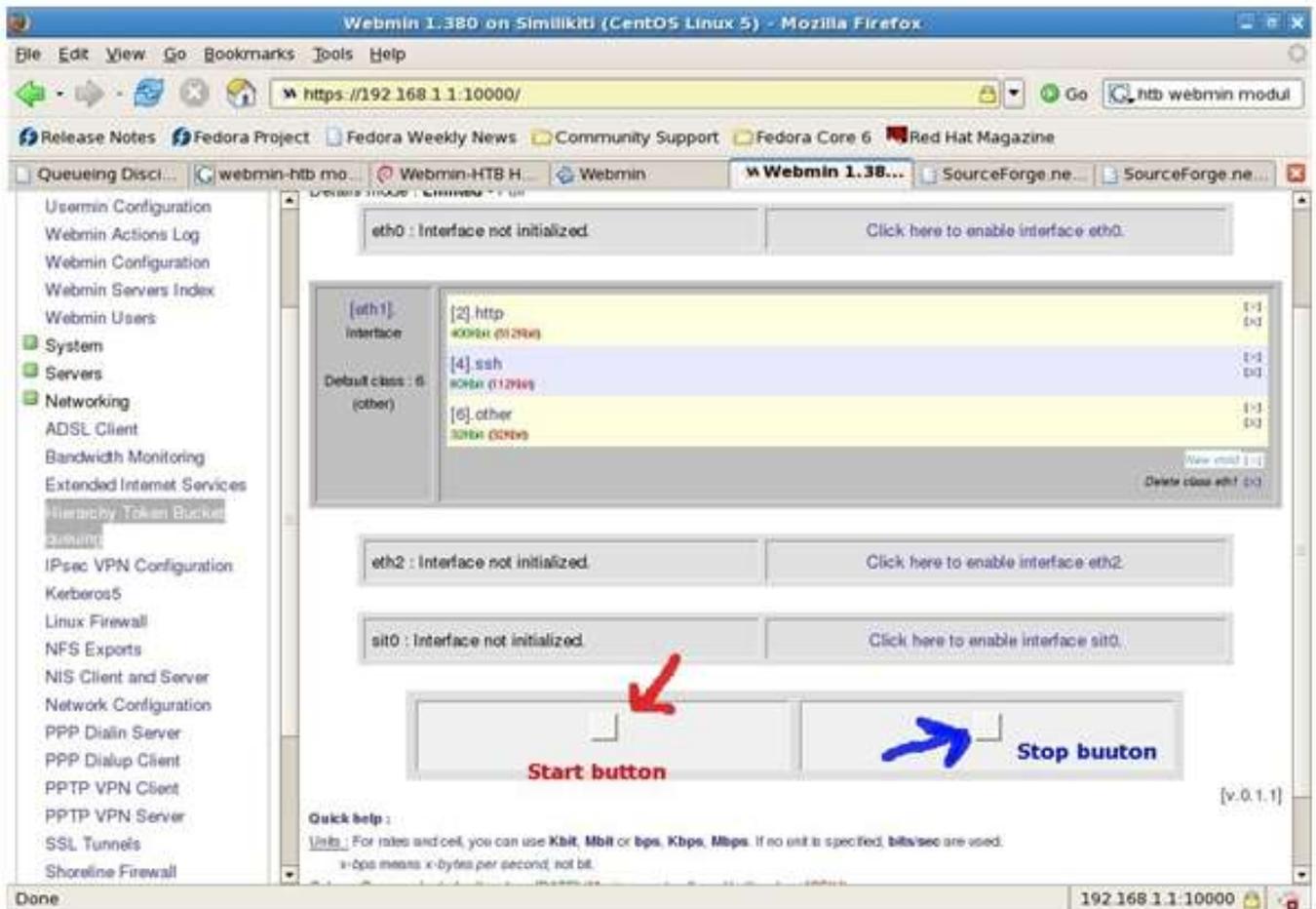
Gambar 13. Halaman Edit Me, untuk pembuatan kelas ketiga

Langkah berikutnya set default class (defaultclass diset untuk kelas " other"), seperti tampak dalam gambar 14.

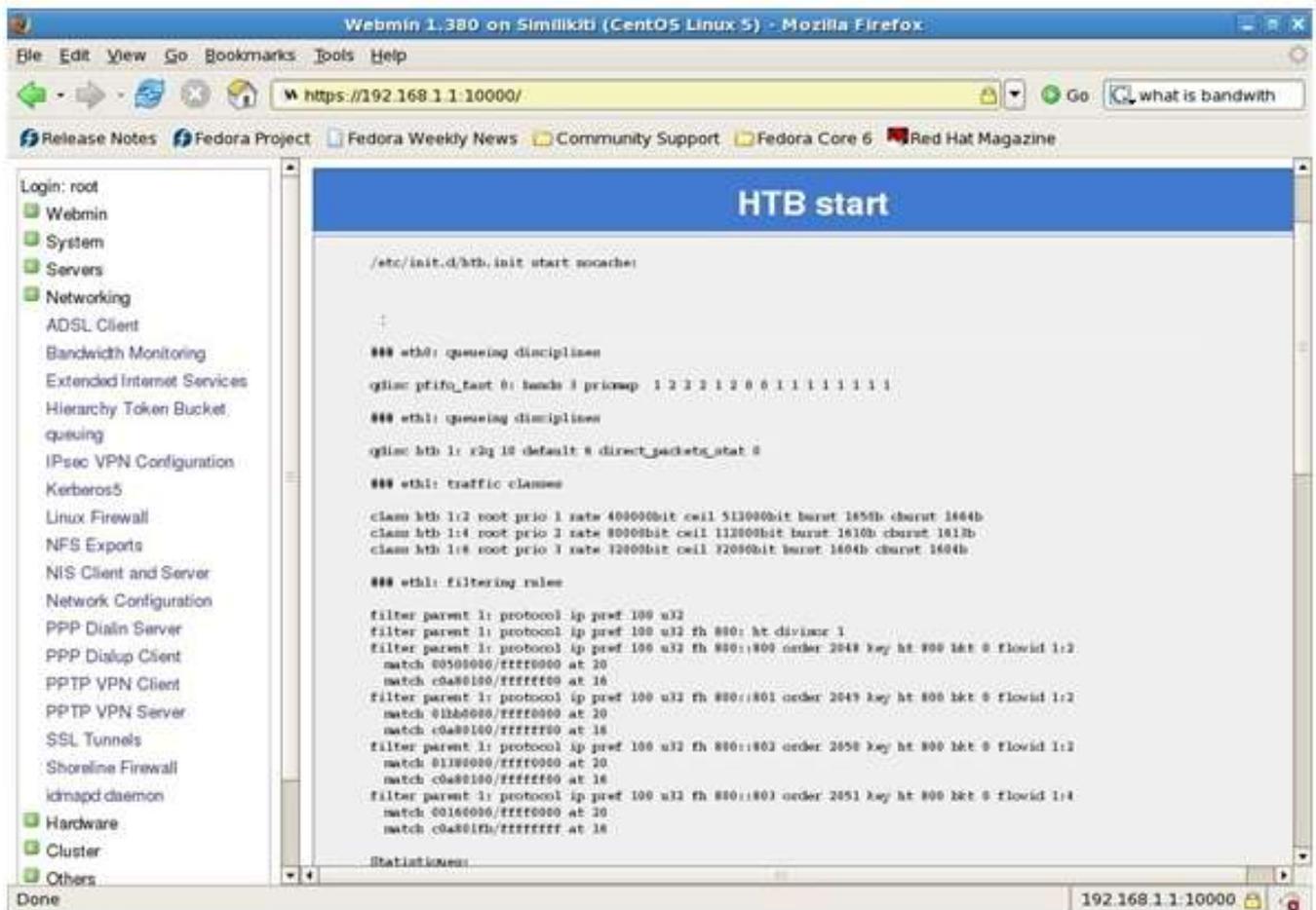


Gambar 4. Setup Default Class

Langkah selanjutnya jika semua kelas dan rule filteringnya sudah selesai Anda definisikan maka Anda dapat segera mengaktifkan htb.init dengan menekan tombol start sebagaimana ditunjukkan dalam gambar 5. (Modul webmin_{htb} ini sepertinya perlu patch agar muncul label start dan stop pada tombol yang ditunjukkan dalam gambar 5.).



Gambar 5. Mengaktifkan HTB init



Gambar 16. Status pengaktifan HTB init.

Semua konfigurasi kelas dan rule dari traffic control menggunakan webmin htb modul disimpan dalam direktori /etc/sysconfig/htb. Konfigurasi setiap kelas disimpan dalam masing-masing file seperti berikut ini:

```

[root@Similikiti ~]# ls -al /etc/sysconfig/htb/
total 28
drwxr-xr-x  2 root root  4096 Feb  3 20:11 .
drwxr-xr-x 12 root root  4096 Feb  3 19:36 ..
-rw-r--r--  1 root root   10 Feb  3 20:04 eth1
-rw-r--r--  1 root root  107 Feb  3 20:07 eth1-2.http
-rw-r--r--  1 root root   55 Feb  3 20:11 eth1-4.ssh
-rw-r--r--  1 root root   31 Feb  3 20:11 eth1-6.other

```

Agar setiap kali booting htb.init diaktifkan maka anda dapat mendaftarkan kepada sistem agar service htb.init diaktifkan saat boot oleh init secara otomatis, untuk itu lakukan perintah berikut:

```

[root@Similikiti ~]# chkconfig htb.init on

```

Selanjutnya cobalah hasil konfigurasi traffic control tersebut dengan cara melakukan transfer file (download) yang terdapat di sebuah server di Internet via SSH/SCP ke komputer 192.168.1.251 untuk menguji apakah rule pada kelas yang kedua berjalan dengan tepat. Atau coba download file via HTTP dari sebuah server di internet ke komputer jaringan lokal untuk menguji rule yang diterapkan pada kelas pertama. Anda dapat mengamati proses download apakah ada perbedaan ketika sebelum di atur menggunakan bandwidth manager pada PC Router dengan setelah diatur. Ada baiknya Anda gunakan tools untuk monitoring network traffic seperti : BW Monitor :monitor.pl (<http://www.docum.org/docum.org/monitor/download/monitor.pl>), HTB Monitor: monitor.pl (http://www.docum.org/docum.org/monitor/download/monitor_tc.pl), dan lain-lain.

Referensi:

<http://lartc.org>

<http://www.docum.org/docum.org>

<http://www.sehier.fr/webmin-htb>