

Diasumsikan bahwa konfigurasi Router Gateway telah selesai, konfigurasinya sama dengan contoh sebelumnya. Namun dari jaringan yang Anda lihat, Anda bisa saja menambahkan fitur DHCP Server pada interface *wlan1* Router Gateway. Sehingga komputer client pada kedua Router CPE bisa mendapatkan IP Address dari Router Gateway.

Tahap pertama yang harus dilakukan tentunya adalah menghubungkan interface *wlan1* Router CPE-1 ke interface *wlan1* milik Router Gateway. Perintah yang dapat digunakan adalah sebagai berikut.

```
[admin@CPE-1] > interface wireless set wlan1 mode=station-pseudobridge \
band=2ghz-b/g ssid="Point to Multi Point Link" scan-list=2400-2500
```

Pastikan Router CPE-1 terhubung ke Gateway, periksalah apakah ada label R di depan interface *wlan1*, periksa pula *registration table*. Pemeriksaan bisa dilakukan pada kedua Router, namun jika ternyata jarak kedua perangkat tersebut berjauhan, dan Anda belum bisa melakukan akses remote ke perangkat yang jauh, pemeriksaan di satu router saja sudah cukup. Contoh uraian berikut memperlihatkan bahwa Router CPE-1 sudah terhubung ke Gateway.

```
[admin@CPE-1] > interface wireless print
Flags: X - disabled, R - running
0 R name="wlan1" mtu=1500 mac-address=00:0C:42:6A:E7:8E arp=enabled
  interface-type=Atheros AR5413
  mode=station-pseudobridge ssid="Point to Multi Point Link" frequency=2412 band=2ghz-b/g
  channel-width=20mhz scan-list=2400-2500 wireless-protocol=unspecified antenna-mode=ant-a
  wds-mode=disabled wds-default-bridge=none wds-ignore-ssid=no bridge-mode=enabled
  default-authentication=yes default-forwarding=yes default-ap-tx-limit=0 default-client-tx-limit=0
  hide-ssid=no security-profile=default compression=no

[admin@CPE-1] > interface wireless registration-table print
# INTERFACE      MAC-ADDRESS      AP  SIGNAL-STRENGTH  TX-RATE  UPTIME
0 wlan1          00:1E:E5:76:CE:E8 yes -64dBm@1Mbps    54.0... 1m31s
```

Langkah berikut adalah membuat interface bridge pada Router CPE-1, dan kemudian memasukkan interface *wlan1* dan *ether1* ke dalam interface bridge tersebut. Perintah yang dapat digunakan adalah sebagai berikut.

```
[admin@CPE-1] > interface bridge add name=bridge1
```

```
[admin@CPE-1] > interface bridge port add interface=wlan1 bridge=bridge1
[admin@CPE-1] > interface bridge port add interface=ether1 bridge=bridge1
```

Konfigurasi IP Address pada interface *bridge1*, paling tidak Anda akan membutuhkannya untuk keperluan konfigurasi lanjutan, pengujian maupun monitoring.

```
[admin@CPE-1] > ip address add address=192.168.10.11/24
interface=bridge1
```

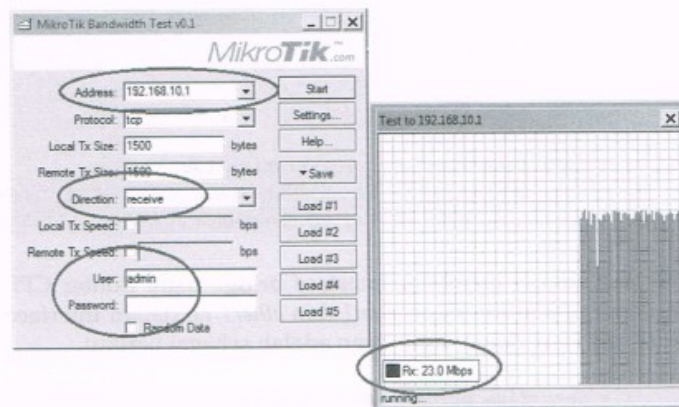
```
[admin@CPE-1] > ip address print
```

Flags: X - disabled, I - invalid, D - dynamic

#	ADDRESS	NETWORK	INTERFACE
0	192.168.10.11/24	192.168.10.0	bridge1

Sampai pada tahapan ini, konfigurasi wireless bridge Anda telah selesai. Komputer client yang berada di interface *ether1* Router CPE, seharusnya sudah dapat mengakses Internet. Tentunya komputer client tersebut harus menggunakan 192.168.10.1 sebagai default gateway maupun DNS Server.

Bila Anda ingin menguji *link* wireless antara komputer client ke Router Gateway, Anda bisa menggunakan aplikasi *btest.exe*. Aplikasi ini dapat didownload pada web site www.mikrotik.co.id dan dapat Anda install pada komputer client. Contoh berikut memperlihatkan pengujian dari komputer client, dimana komputer tersebut mendapatkan bandwidth 23 Mbps ke Router Gateway



Gambar 4.19
Hasil pengujian bandwidth dari komputer client ke Gateway

Bagaimana dengan Router CPE-2? Konfigurasi pada Router ini sama dengan Router CPE-1. Yang membedakan hanyalah IP Address 192.168.10.12 yang harus Anda konfigurasi pada interface *bridge1*. Secara singkat, hasil konfigurasinya adalah sebagai berikut.

```
[admin@CPE-2] > interface wireless set wlan1 mode=station-pseudobridge \
band=2ghz-b/g ssid="Point to Multi Point Link" scan-list=2400-2500 \
radio-name=CPE-1

[admin@CPE-2] > interface bridge add name=bridge1

[admin@CPE-2] > interface bridge port add interface=wlan1 bridge=bridge1

[admin@CPE-2] > interface bridge port add interface=ether1
bridge=bridge1

[admin@CPE-2] > ip address add address=192.168.10.12/24
interface=bridge1
```

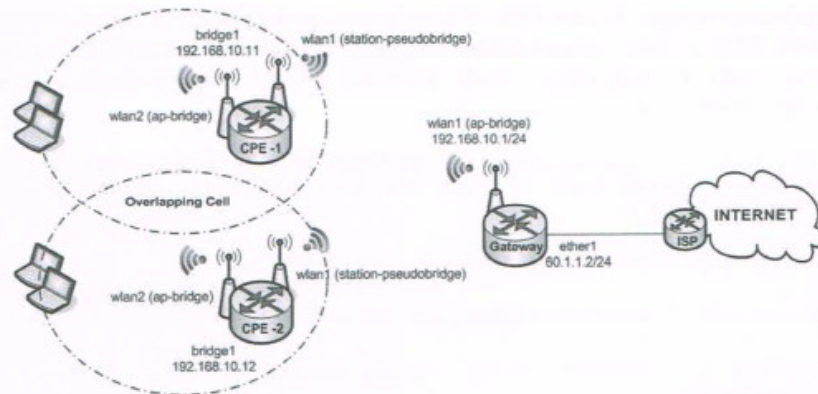
4

Skenario 2

Skenario ini digunakan jika ternyata komputer client pada Router CPE-1 dan CPE-2 adalah laptop yang akan menggunakan koneksi wireless untuk mengakses Internet. Skenario ini bisa dijalankan jika Router CPE memiliki dua interface wireless, satu interface (*wlan1*) digunakan untuk membangun *link* dengan Router Gateway dan satu interface lagi (*wlan2*) akan digunakan sebagai *access point* untuk melayani laptop yang berada di sekitar Router CPE tersebut.

Walaupun demikian, bisa saja Anda membangun topologi seperti skenario 2 ini, meskipun Router CPE hanya memiliki satu interface wireless. Anda bisa menggunakan teknik WDS (*Wireless Distribution System*). Sayangnya teknik ini baru akan dibahas di buku MikroTik Kung Fu selanjutnya, saya harap Anda bisa bersabar ☺.

Topologi pada skenario 2 ini dapat dilihat pada gambar berikut. Perhatikanlah interface *wlan1* pada Router CPE yang menggunakan mode *station-pseudobridge* karena berfungsi sebagai client, dan interface *wlan2* yang menggunakan mode *ap-bridge* karena berfungsi sebagai *access point* bagi client laptop.



Gambar 4.20

Client Laptop mendapatkan koneksi Internet dari wlan2 Router CPE

Jaringan wireless tersebut juga akan di-bridge, sehingga menghasilkan satu kesatuan jaringan wireless dengan network address 192.168.10.0/24. Jika Anda menginginkan fitur *roaming* di antara CPE-1 dan CPE-2, maka aturlah penyebaran sinyalnya sedemikian rupa sehingga tercipta *overlapping cell*, seperti pada gambar.

Pada area *overlapping cell*, laptop sebagai komputer client harus dapat mendeteksi adanya dua signal, baik dari CPE-1 maupun dari CPE-2. Dengan hadirnya kemampuan *roaming* pada jaringan wireless Anda, maka client laptop tidak akan mengalami *disconnected* (putus koneksi) walaupun client tersebut sering berpindah-pindah tempat. Tentunya selama masih dalam jangkauan signal dari CPE-1 maupun CPE-2.

Fitur *roaming* ini juga bisa dihadirkan jika Anda mengkonfigurasi SSID yang sama pada interface wlan2 dari kedua Router CPE. Frekuensi (channel) yang digunakan pun sebaiknya sama. Sedangkan untuk menguji apakah fitur *roaming* berhasil, cobalah Anda menggunakan laptop dan kemudian melakukan aktifitas download ke Internet. Cobalah terlebih dahulu dengan menghubungkan diri ke CPE-1, tentunya Anda harus berada di dekat Router tersebut. Kemudian berjalanlah ke area yang lebih dekat ke Router CPE-2. Seharusnya aktifitas download Anda tidak terputus sama sekali.

Setelah Anda memahami sedikit konsep dari jaringan wireless yang ingin menghadirkan fitur *roaming* tadi, maka langkah selanjutnya adalah melakukan eksekusi melalui beberapa perintah konfigurasi. Diasumsikan bahwa Anda hanya melanjutkan konfigurasi yang sudah pernah dibuat pada skenario 1 sebelumnya. Sehingga diasumsikan juga bahwa Router CPE sudah terhubung

ke Gateway dan konfigurasi bridge juga sudah dilakukan.

Anda tinggal mengaktifkan *access point* pada interface *wlan2* dengan SSID "*Kung Fu HotSpot*" dan frekuensi 2,412 GHz (channel 1) pada Router CPE-1 dengan menggunakan perintah sebagai berikut.

```
[admin@CPE-1] > interface wireless set wlan2 mode=ap-bridge \
band=2ghz-b/g frequency=2412 ssid="Kung Fu HotSpot"

[admin@CPE-1] > interface wireless print
Flags: X - disabled, R - running
 0 R name="wlan1" mtu=1500 mac-address=00:0C:42:6A:E7:8E arp=enabled
   interface-type=Atheros AR5413 mode=station-pseudobridge
   ssid="Point to Multi Point Link" frequency=2452 band=2ghz-b/g
   channel-width=20mhz scan-list=2400-2500 wireless-
   protocol=unspecified antenna-mode=ant-a wds-mode=disabled wds-de-
   fault-bridge=none wds-ignore-ssid=no bridge-mode=enabled default-
   authentication=yes default-forwarding=yes default-ap-tx-limit=0
   default-client-tx-limit=0 hide-ssid=no security-profile=default
   compression=no

 1 R name="wlan2" mtu=1500 mac-address=00:0C:42:6A:E7:87 arp=enabled
   interface-type=Atheros AR5413 mode=ap-bridge ssid="Kung Fu
   HotSpot" frequency=2412 band=2ghz-b/g channel-width=20mhz scan-
   list=default wireless-protocol=unspecified antenna-mode=ant-a wds-
   mode=disabled wds-default-bridge=none wds-ignore-ssid=no bridge-
   mode=enabled default-authentication=yes default-forwarding=yes
   default-ap-tx-limit=0 default-client-tx-limit=0 hide-ssid=no
   security-profile=default compression=no
```

Untuk mencegah terjadinya interferensi frekuensi, jangan sampai menggunakan frekuensi (channel) yang sama dengan *link wireless point to multi point* ke Router Gateway. Perhatikanlah konfigurasi di atas di mana *link point to multi point* menggunakan 2,452 GHz sedangkan interface *wlan2* yang digunakan untuk melayani client laptop menggunakan frekuensi 2,412 GHz.

Selanjutnya yang harus dilakukan adalah memasukkan interface *wlan2* ke dalam interface *bridge1* dengan menggunakan perintah sebagai berikut:

```
[admin@CPE-1] > interface bridge port add interface=wlan2 bridge=bridge1

[admin@CPE-1] > interface bridge port print
Flags: X - disabled, I - inactive, D - dynamic
#  INTERFACE  BRIDGE  PRIORITY  PATH-COST  HORIZON
0  wlan1      bridge1    0x80      10         none
1  ether1     bridge1    0x80      10         none
2  wlan2      bridge1    0x80      10         none
```


Perintah konfigurasi yang sama juga harus Anda terapkan pada Router CPE-2, sehingga hasil akhir konfigurasi pada router tersebut dapat dilihat seperti uraian berikut.

```
[admin@CPE-2] > interface wireless print
Flags: X - disabled, R - running
 0 R name="wlan1" mtu=1500 mac-address=00:0C:42:6A:E7:6E arp=enabled
   interface-type=Atheros AR5413 mode=station-pseudobridge
   ssid="Point to Multi Point Link" frequency=2412 band=2ghz-b/g
   channel-width=20mhz scan-list=2400-2500 wireless-
   protocol=unspecified
   antenna-mode=ant-a wds-mode=disabled wds-default-bridge=none
   wds-ignore-ssid=no bridge-mode=enabled default-authentication=yes
   default-forwarding=yes default-ap-tx-limit=0 default-client-tx-
lim
   it=0 hide-ssid=no security-profile=default compression=no

 1 R name="wlan2" mtu=1500 mac-address=00:0C:42:6A:E7:84 arp=enabled
   interface-type=Atheros AR5413 mode=ap-bridge ssid="Kung Fu
   HotSpot" frequency=2412 band=2ghz-b/g channel-width=20mhz scan-
   list=2400-2500 wireless-protocol=unspecified antenna-mode=ant-
   a wds-mode=disabled wds-default-bridge=none wds-ignore-ssid=no
   bridge-mode=enabled default-authentication=yes default-
   forwarding=yes default-ap-tx-limit=0
   default-client-tx-limit=0 hide-ssid=no security-profile=default
   compression=no

[admin@CPE-2] > interface bridge port print
Flags: X - disabled, I - inactive, D - dynamic
#  INTERFACE  BRIDGE  PRIORITY  PATH-COST  HORIZON
0  wlan1      bridge1  0x80      10         none
1  ether1     bridge1  0x80      10         none
2  wlan2      bridge1  0x80      10         none
```

Virtual Access point

Router MikroTik menyediakan fitur *Virtual Access Point (VAP)* pada interface wireless-nya. Dengan adanya fitur ini, Anda dapat membuat beberapa *access point* dengan berbekal satu interface wireless fisik. *Virtual Access Point* tersebut nantinya dapat menggunakan SSID yang berbeda-beda, IP Address yang berbeda-beda bahkan MAC-Address yang berbeda pula.

Anda juga dapat menghidupkan DHCP Server pada masing-masing *Virtual Access Point* tersebut. Mengatur firewall di antara interface virtual pun bisa Anda lakukan. Fitur ini sama dengan teknik VLAN (*Virtual Local Area Network*) pada

jaringan kabel, uppsss... mohon maaf, saya menambah lagi perbendaharaan istilah jaringan Anda. Kapan-kapan kita bisa membahas VLAN tersebut.

Kembali ke *Virtual Access Point*, berikut adalah kondisi di mana Anda bisa memanfaatkan fitur ini, misalnya pada suatu sekolah yang ingin menghadirkan dua hotspot yang terpisah antara guru dan siswanya. Anda bisa melihatnya pada skenario 1 berikut ini.

Skenario 1

Pada skenario ini, Anda dihadapkan untuk membuat jaringan wireless pada suatu sekolah. Namun diinginkan jaringan wireless (hotspot) untuk guru terpisah dengan jaringan untuk para siswa. Jika kondisi keuangan Anda memungkinkan.... J, bisa saja digunakan dua Router MikroTik, atau sebuah Router yang memiliki dua interface wireless. Namun untuk hitungan efisiensi, fitur *Virtual Access Point* dapat menjadi pertimbangan untuk diterapkan. Anda bisa melihat skenario ini pada gambar berikut.

4



Gambar 4.21
Implementasi Virtual Access Point pada dua HotSpot

Diasumsikan bahwa Router Gateway telah terhubung ke Internet dengan IP Address 60.1.1.2, default gateway, DNS Server dan konfigurasi *masquerade*. Langkah berikutnya adalah mengkonfigurasi *wlan1* agar bisa menjadi *access point* bagi laptop para guru. Perintah yang dapat digunakan adalah sebagai berikut.

```
[admin@Gateway] > interface wireless set wlan1 mode=ap-bridge \
band=2ghz-b/g frequency=2412 ssid="HotSpot Guru"
```

Tentunya Anda harus memberikan IP Address pada interface *wlan1*, seperti perintah berikut ini.

```
[admin@Gateway] > ip address add address=192.168.10.1/24 interface=wlan1
```

Langkah terakhir yang harus lakukan adalah meng-aktifkan DHCP Server pada interface *wlan1* dengan menggunakan perintah berikut ini.

```
[admin@Gateway] > ip dhcp-server setup
Select interface to run DHCP server on

dhcp server interface: wlan1
Select network for DHCP addresses
```

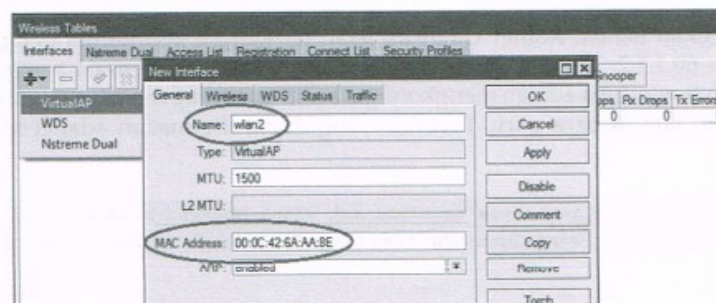
4

Anda bisa saja melengkapi jaringan wireless untuk para guru tersebut dengan HotSpot Gateway System yang sudah di bahas pada buku *MikroTik Kung Fu Kitab 1*. Dengan fitur ini, maka para guru yang akan menggunakan jaringan wireless harus memasukkan username dan password. Perintah yang dapat digunakan untuk mengaktifkan HotSpot Gateway System pada interface *wlan1* adalah sebagai berikut.

```
[admin@Gateway] > ip hotspot setup
Select interface to run HotSpot on

hotspot interface: wlan1
Set HotSpot address for interface
```

Bagaimana dengan konfigurasi jaringan wireless untuk para siswa? Jaringan wireless untuk siswa akan menggunakan *Virtual Access Point* yang dapat diaktifkan dengan menggunakan WinBox melalui menu **Wireless** → tab **Interfaces** → tombol **Add** → pilih **VirtualAP**, seperti terlihat pada gambar berikut ini.



Gambar 4.22
Menambahkan virtual access point

Setelah *Virtual Access Point* diaktifkan dengan nama *wlan2*, Anda dapat melakukan konfigurasi SSID, IP Address, DHCP Server maupun HotSpot Gateway System dengan perintah seperti berikut ini.

```
[admin@Gateway] > interface wireless set wlan2 ssid="HotSpot Siswa" \
                    master-interface=wlan1 mac-address=00:0C:42:6A:AA:8E

[admin@Gateway] > ip address add address=192.168.20.1/24 interface=wlan2

[admin@Gateway] > ip dhcp-server setup
Select interface to run DHCP server on

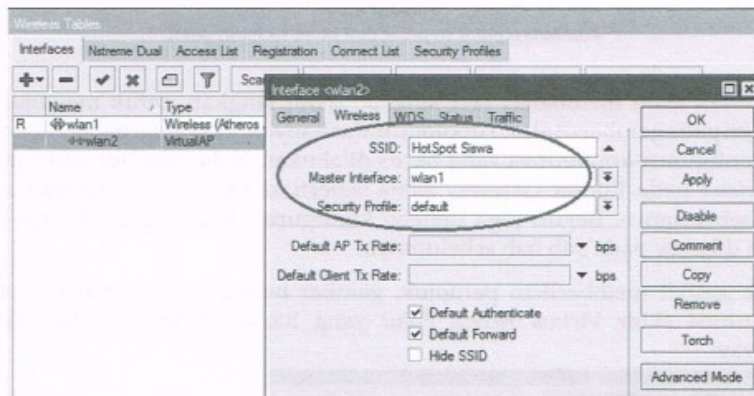
dhcp server interface: wlan2
Select network for DHCP addresses

[admin@Gateway] > ip hotspot setup
Select interface to run HotSpot on

hotspot interface: wlan2
Set HotSpot address for interface
```

4

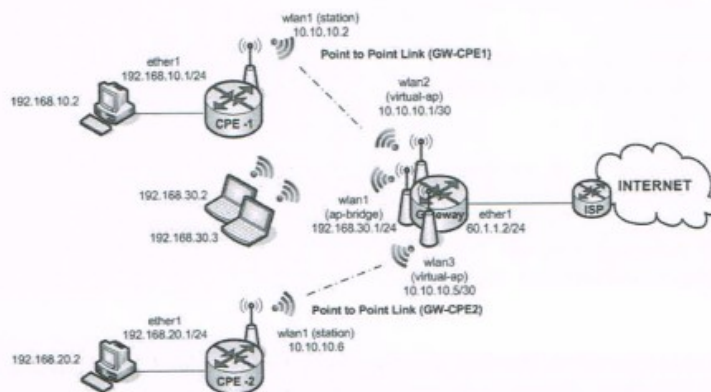
Dengan menggunakan WinBox, konfigurasi interface *wlan2* ini dapat dilakukan pada tab *Wireless*, seperti pada gambar berikut ini:



Gambar 4.23
Parameter SSID pada interface wlan2 (Virtual AP)

Skenario 2

Anda bisa saja menggunakan *Virtual Access Point* untuk membuat *link* topologi *point to point*, sekaligus menghadirkan *point to multi point* untuk melayani beberapa client laptop. Perhatikanlah gambar berikut, Router Gateway memiliki 2 (dua) *Virtual Access Point* (*wlan2* dan *wlan3*) yang digunakan untuk membuat *link* dengan Router CPE-1 dan CPE-2. Sedangkan interface fisik (*wlan1*) digunakan untuk memberikan akses Internet kepada beberapa laptop client.



Gambar 4.24

Kombinasi point to point dan point to multi point

Saya tidak akan membahas lagi langkah demi langkah untuk mewujudkan jaringan pada gambar di atas. Topologi di atas hanyalah topologi pengembangan, jadi konfigurasi-konfigurasi yang harus dilakukan pada interface *wlan1*, *wlan2* dan *wlan3* pada Router Gateway sama seperti konfigurasi-konfigurasi *access point* sebelumnya. Begitu juga dengan konfigurasi pada kedua Router CPE, sudah dibahas pada sub bab sebelumnya.

Untuk sedikit memberikan petunjuk, gambar berikut memperlihatkan hasil konfigurasi akhir *Virtual Access Point* yang harus dilakukan pada Router Gateway.

Name	Type	MAC Address	SSID	Auth	Band	Channel	Frequency (MHz)	Mode
wlan1	VirtualAP	02:00:00:00:00:00	10.10.10.1	none	2.4G-HZ	2412	2412	Point to Multi CPE
wlan2	VirtualAP	02:00:00:00:00:00	10.10.10.2	none	2.4G-HZ	2412	2412	Point to Point CPE1
wlan3	VirtualAP	02:00:00:00:00:00	10.10.10.3	none	2.4G-HZ	2412	2412	Point to Point CPE2

Gambar 4.25

Virtual access point pada Router Gateway

Adapun konfigurasi dasar pada Router Gateway dapat dilihat seperti berikut ini:

```
[admin@Gateway] > ip address print
Flags: X - disabled, I - invalid, D - dynamic
#   ADDRESS           NETWORK           INTERFACE
0   60.1.1.2/30        60.1.1.0          ether1
1   192.168.30.1/24    192.168.30.0      wlan1
2   10.10.10.1/30      10.10.10.0        wlan2
3   10.10.10.5/30      10.10.10.4        wlan3

[admin@Gateway] > ip route print
Flags: X - disabled, A - active, D - dynamic, C - connect, S - static, r
- rip, b - bgp, o - ospf, m
B - blackhole, U - unreachable, P - prohibit
#   DST-ADDRESS      PREF-SRC      GATEWAY          DISTANCE
0 A S 0.0.0.0/0        60.1.1.1      1
1 ADC 10.10.10.0/30  10.10.10.1    wlan2             0
2 ADC 10.10.10.4/30  10.10.10.5    wlan3             0
3 ADC 60.1.1.0/30    60.1.1.2      ether1            0
4 A S 192.168.10.0/24 10.10.10.2    1
5 A S 192.168.20.0/24 10.10.10.6    1
6 ADC 192.168.30.0/24 192.168.30.1  wlan1             0

[admin@Gateway] > ip dns print
servers: 60.1.1.1,70.1.1.1
dynamic-servers:
allow-remote-requests: yes
max-udp-packet-size: 4096
cache-size: 2048KiB
cache-max-ttl: 1w
cache-used: 8KiB

[admin@Gateway] > ip firewall nat print
Flags: X - disabled, I - invalid, D - dynamic
chain=srcnat action=masquerade out-interface=ether1
```

Sedangkan konfigurasi interface wireless beserta *Virtual Access Point* dapat dilihat pada uraian berikut ini.

```
[admin@Gateway] > interface wireless print
Flags: X - disabled, R - running
```



```

0  name="wlan1" mtu=1500 mac-address=00:0C:42:6A:E7:8E arp=enabled
    interface-type=Atheros AR5413 mode=ap-bridge
    ssid="HotSpot Kung Fu" frequency=2412 band=2ghz-b/g channel-
    width=20mhz scan-list=default
    wireless-protocol=unspecified antenna-mode=ant-a wds-mode=disabled
    wds-default-bridge=none wds-ignore-ssid=no
    bridge-mode=enabled default-authentication=yes default-
    forwarding=yes default-ap-tx-limit=0
    default-client-tx-limit=0 hide-ssid=no security-profile=wpa-kung-
    fu-1 compression=no

1  name="wlan2" mtu=1500 mac-address=02:0C:42:6A:E7:1E arp=enabled
    interface-type=virtual-AP master-interface=wlan1
    ssid="Point to Point GW-CPE1" wds-mode=disabled wds-default-
    bridge=none wds-ignore-ssid=no bridge-mode=enabled
    default-authentication=yes default-forwarding=yes default-ap-tx-
    limit=0 default-client-tx-limit=0 hide-ssid=no
    security-profile=default

2  name="wlan3" mtu=1500 mac-address=02:0C:42:6A:E7:2E arp=enabled
    interface-type=virtual-AP master-interface=wlan1
    ssid="Point to Point GW-CPE2" wds-mode=disabled wds-default-
    bridge=none wds-ignore-ssid=no bridge-mode=enabled
    default-authentication=yes default-forwarding=yes default-ap-tx-
    limit=0 default-client-tx-limit=0 hide-ssid=no
    security-profile=default
    
```

Konfigurasi DHCP Server yang memungkinkan pembagian IP Address yang lebih mudah, dapat diterapkan pada interface *wlan1* yang akan melayani beberapa client laptop. Walaupun Anda menginginkan pembagian IP Address secara otomatis bagi kedua Router CPE, DHCP Server bisa juga diaktifkan pada interface *wlan2* dan *wlan3*.

Keamanan Jaringan Wireless

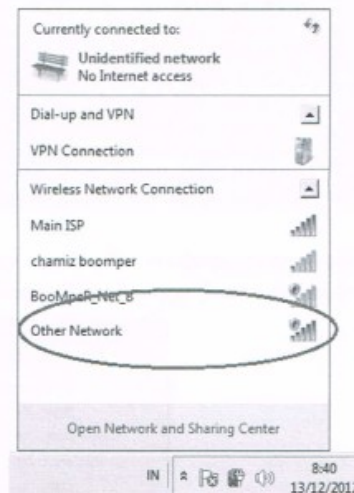
Mengamankan jaringan wireless membutuhkan perhatian yang lebih, baik dari segi desain maupun konfigurasi apa yang akan diterapkan. Ini disebabkan signal wireless bisa menyebar ke mana saja dan setiap user bisa dengan leluasa mendapatkan signal tersebut.

Jika tidak menerapkan standar keamanan yang tepat, maka semudah user legal Anda mendapatkan signal, semudah itu pula user (ilegal) masuk ke dalam jaringan Anda. Pada sub bab ini akan dijelaskan beberapa teknik dasar untuk mengamankan jaringan wireless.

Hidden SSID

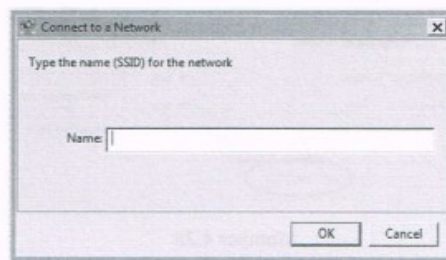
Salah satu teknik paling dasar yang dapat Anda terapkan adalah menyembunyikan SSID dari *access point* Anda. Dengan SSID yang tersembunyi user harus mengetahui dengan pasti apa nama SSID yang digunakan. Walaupun user tersebut melakukan *scanning*, maka tidak akan terlihat nama atau SSID dari jaringan wireless Anda. Biasanya teknik ini dapat diterapkan pada topologi *point to point* maupun *point to multi point* untuk menghubungkan Router MikroTik wireless.

Jangan menerapkan teknik ini pada jaringan wireless publik, misalnya hotspot yang diterapkan di kampus, sekolah, cafe maupun area publik di kantor. Teknik ini akan menyebabkan pengguna akan selalu menanyakan apa nama SSID yang harus digunakan, tentu akan merepotkan Anda. Jika menggunakan client laptop dengan mesin Windows, maka jaringan wireless yang menyembunyikan SSID akan terlihat seperti berikut ini.



Gambar 4.26
Jaringan wireless dengan Hidden SSID

Dan pada saat ada user yang akan menggunakan "Other Network" tersebut maka akan muncul permintaan nama SSID seperti gambar berikut ini:



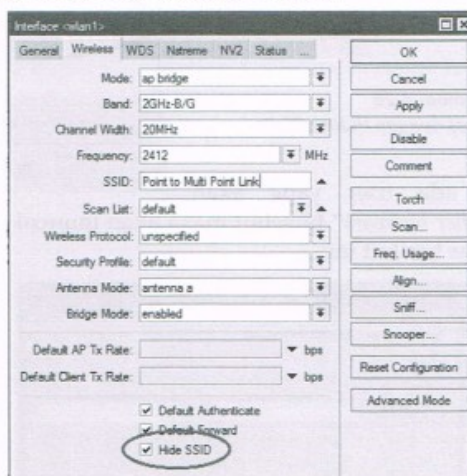
Gambar 4.27
Permintaan nama SSID kepada client

Kesimpulannya, bijaklah dalam menentukan kapan Anda akan menyembunyikan SSID dan kapan Anda akan mempublikasikannya. Untuk menyembunyikan SSID yang ada pada interface wireless *access point*, dapat digunakan perintah seperti berikut ini.

```
[admin@Gateway] > interface wireless set wlan1 mode=ap-bridge \
ssid="Point to Multi Point Link" band=2ghz-b/g frequency=2412 \
radio-name=Gateway hide-ssid=yes

[admin@Gateway] > interface wireless print
Flags: X - disabled, R - running
0   name="wlan1" mtu=1500 mac-address=00:0C:42:6A:E7:8E arp=enabled
    interface-type=Atheros AR5413 mode=ap-bridge ssid="Point to Multi
    Point Link" frequency=2412 band=2ghz-b/g channel-width=20mhz
    scan-list=default wireless-protocol=unspecified antenna-mode=ant-
    a wds-mode=disabled wds-default-bridge=none wds-ignore-ssid=no
    bridge-mode=enabled default-authentication=yes default-
    forwarding=yes default-ap-tx-limit=0 default-client-tx-limit=0
    hide-ssid=yes security-profile=default compression=no
```

Jika menggunakan WinBox, Anda dapat melakukannya pada parameter interface *wlan1*, seperti pada gambar berikut ini:



Gambar 4.28
Hidden SSID pada wlan1

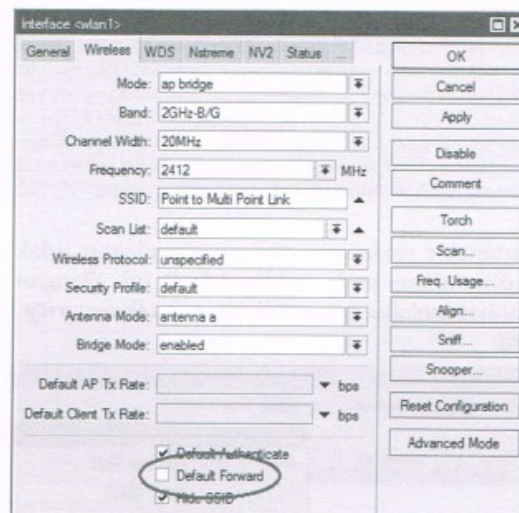
Default Forward

Parameter *default-forward* dapat dinon-aktifkan pada Router yang berfungsi sebagai *access point*. Bila menggunakan skenario-skenario sebelumnya maka

parameter ini dapat non-aktifkan pada Router Gateway. Dengan kondisi *default-forward* di non-aktifkan, maka client wireless Anda tidak akan dapat saling berhubungan, Router CPE-1 nantinya tidak akan dapat berhubungan dengan CPE-2 pada topologi *point to multi point* sebelumnya. Namun kedua Router CPE masih dapat menuju Internet melalui Router Gateway. Perintah yang dapat Anda gunakan untuk menon-aktifkan fitur ini adalah sebagai berikut.

```
[admin@Gateway] > interface wireless set wlan1 default-forwarding=no
```

Dengan menggunakan WinBox, dapat Anda lihat seperti gambar berikut ini.



Gambar 4.29
Default Forward dalam kondisi disable

Security Profiles

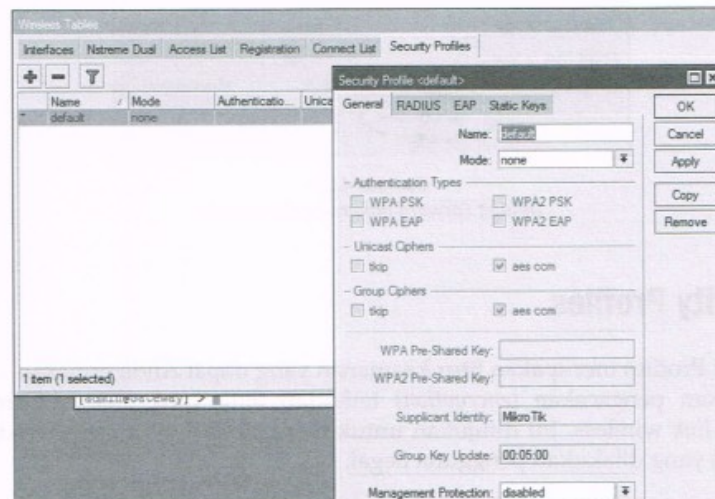
Security Profiles merupakan fitur keamanan yang dapat Anda gunakan untuk melakukan pengacakan (*encryption*) terhadap data yang akan dikirimkan melalui *link* wireless. Ini ditujukan untuk menghindari aktifitas penyadapan (*sniffing*) yang dilakukan pengguna ilegal.

Untuk menerapkan *encryption* ini, Anda harus mengkonfigurasi "key" baik pada sisi *access point* maupun pada sisi *wireless client*. Ada berbagai macam teknik *encryption* yang dapat Anda gunakan, namun yang mempunyai tingkat keamanan yang lebih baik adalah Wi-Fi Protected Access (WPA2).

Untuk menerapkan WPA2 pada jaringan wireless, Anda dapat melakukannya dengan menggunakan menu *Security Profiles*. Secara default MikroTik sudah menyediakan satu profile yang dapat langsung digunakan namun profile tersebut tidak menerapkan *encryption*. Untuk melihat *profile default*-nya Anda dapat menggunakan perintah seperti berikut ini.

```
[admin@Gateway] > interface wireless security-profiles print
Flags: * - default
0 *  name="default" mode=none authentication-types="" unicast-
    ciphers=aes-ccm group-ciphers=aes-ccm wpa-pre-shared-key=""
    wpa2-pre-shared-key="" supplicant-identity="MikroTik" eap-
    methods=passthrough tls-mode=no-certificates tls-certificate=none
    static-algo-0=none static-key-0="" static-algo-1=none static-
    key-1="" static-algo-2=none static-key-2="" static-algo-3=none
    static-key-3="" static-transmit-key=key-0 static-sta-private-
    algo=none static-sta-private-key="" radius-mac-authentication=no
    radius-mac-accounting=no radius-eap-accounting=no interim-
    update=0s
    radius-mac-format=XX:XX:XX:XX:XX:XX radius-mac-mode=as-username
    radius-mac-caching=disabled group-key-update=5m management-
    protection=disabled management-protection-key=""
```

Perhatikanlah parameter *mode=none* yang menandakan tidak adanya teknik *encryption* yang diterapkan pada *profile default* ini. Dengan menggunakan WinBox, dapat dilihat melalui menu *Wireless* → tab *Security Profiles*, seperti gambar berikut ini:



Gambar 4.30
Default Security Profile

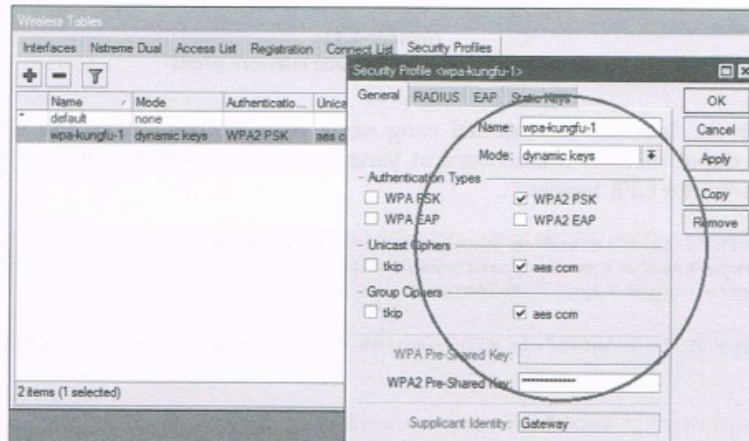
Anda dapat saja menggunakan profile default ini untuk mengkonfigurasi WPA Key. Namun, akan lebih baik jika membuat profile baru, bahkan bisa saja membuat beberapa profile dengan *key* yang berbeda-beda. Sehingga nantinya Anda tinggal memilih *key* yang mana akan digunakan. Bisa saja Anda menerapkan *key* yang berbeda-beda untuk setiap interface. Atau pun menerapkan *key* yang berbeda antara *access point* dan *virtual access point*.

Sebagai contoh penerapan, misalnya Anda memiliki topologi *point to multi point* dengan antara Router Gateway dengan Router CPE-1 maupun CPE-2. Pada skenario sebelumnya, tidak diterapkan fitur keamanan sedikit pun, sehingga bisa saja Router Gateway menerima *wireless client* selain CPE-1 maupun CPE-2. Pada contoh ini, WPA *key* yang akan digunakan adalah WPA2 PSK dengan *Pre Shared Key* yang akan dikonfigurasi pada seluruh Router adalah "*KungFuSecret*".

Langkah pertama yang harus dilakukan adalah membuat profile baru pada Router Gateway dengan menggunakan parameter-parameter tadi. Perintah yang dapat digunakan adalah sebagai berikut.

```
[admin@Gateway] > interface wireless security-profiles add \
name=wpa-kungfu-1 mode=dynamic-keys authentication-types=wpa2-psk \
wpa2-pre-shared-key=KungFuSecret
```

Jika ingin melakukannya dengan menggunakan WinBox, maka menu yang dapat digunakan adalah *Wireless* → *Security Profiles* → tombol *Add*, seperti pada gambar berikut.

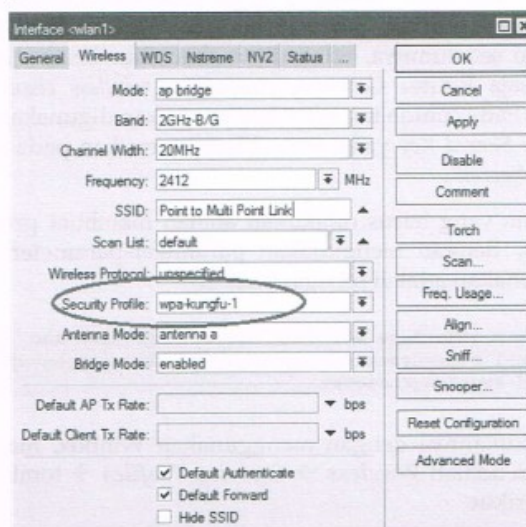


Gambar 4.31
Konfigurasi WPA2 Key

Kemudian konfigurasi interface *wlan1* untuk menggunakan *security profile* ini dengan menggunakan perintah sebagai berikut.

```
[admin@Gateway]> interface wireless set wlan1 security-profile=wpa-kungfu-1
```

Dengan menggunakan WinBox, parameter *security profile* pada interface *wlan1* akan terlihat seperti berikut ini.



Gambar 4.32
Parameter Security Profile pada interface wlan1

Setelah itu, lakukan konfigurasi yang sama pada Router CPE-1 dan CPE-2. Anda dapat melihat uraian berikut yang memperlihatkan konfigurasi pada kedua Router CPE tersebut .

```
[admin@CPE-1] > interface wireless security-profiles add \
name=wpa-kungfu-1 mode=dynamic-keys authentication-types=wpa2-psk \
wpa2-pre-shared-key=KungFuSecret

[admin@CPE-1] > interface wireless set wlan1 security-profile=wpa-kung-
fu-1

[admin@CPE-2] > interface wireless security-profiles add \
name=wpa-kungfu-1 mode=dynamic-keys authentication-types=wpa2-psk \
wpa2-pre-shared-key=KungFuSecret

[admin@CPE-2] > interface wireless set wlan1 security-profile=wpa-kungfu-1
```

Snoop and Scan

Snooper

Snooper adalah tool yang dapat digunakan untuk melihat rentang frekuensi yang tersedia beserta perangkat-perangkat wireless apa saja yang sedang aktif pada frekuensi tertentu. Yang perlu diperhatikan pada saat akan menggunakan tool ini adalah parameter band yang harus dipilih sesuai frekuensi yang akan Anda intip (snoop).

Pada saat menjalankan *Snooper* pada sebuah interface, maka interface tersebut akan tidak bekerja untuk melayani client. Sehingga bila menjalankan pada interface yang berfungsi sebagai *access point*, maka semua *wireless client* Anda akan mengalami *disconnected*.

Untuk menjalankan *Snooper* ini dapat dilakukan melalui perintah-perintah command line, maupun melalui WinBox. Bila menggunakan WinBox, maka tampilan grafis akan lebih memudahkan Anda untuk melihat aktifitas perangkat pada masing-masing frekuensi. Dengan Winbox, maka menu yang dapat digunakan adalah *Wireless* → tab *Interfaces* → tombol *Snooper*, kemudian pilih interface yang akan digunakan → tombol *Start*, seperti terlihat pada gambar berikut ini.

Frequency (MHz)	Band	Address	SSID	Signal	Of Freq. (%)	Of Traf. (%)	Bandwidth	Networks	Stations
2412	2GHz-G	00:0C:42:6A:E7:8E	Kung Fu HotSpot	-79	100.0	100.0	11.4 kbps	1	2
2412	2GHz-G	00:0C:42:6A:E7:8E	Kung Fu HotSpot	-79	100.0	100.0	11.4 kbps	1	2
2412	2GHz-G	00:0C:42:6A:E7:6E	Kung Fu HotSpot	-35	0.0	0.0	0 bps	0	0
2417	2GHz-G			1.6	0.0	0.0	13.4 kbps	0	0
2422	2GHz-G			0.0	0.0	0.0	0 bps	0	0
2427	2GHz-G			0.0	0.0	0.0	0 bps	0	0
2432	2GHz-G			0.0	0.0	0.0	0 bps	0	0
2437	2GHz-G			0.0	0.0	0.0	0 bps	0	0
2442	2GHz-G			0.0	0.0	0.0	0 bps	0	0
2447	2GHz-G			0.3	0.0	0.0	2.0 kbps	0	0
2452	2GHz-G			0.0	0.0	0.0	0 bps	0	0
2457	2GHz-G			0.0	0.0	0.0	0 bps	0	0
2462	2GHz-G			0.0	0.0	0.0	0 bps	0	0

Gambar 4.33
Snooper untuk monitoring rentang frekuensi 2,4 GHz

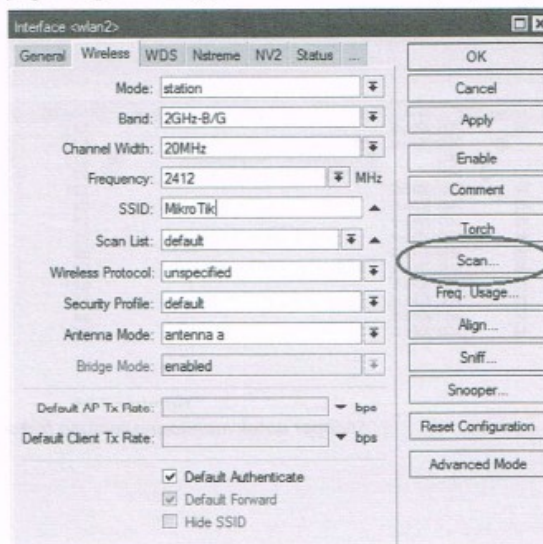
Dari hasil mengintip (snoop) di atas, Anda dapat melihat bahwa pada frekuensi 2412 MHz (Channel 1) terdapat dua perangkat wireless dengan MAC Address 00:0C:42:6A:E7:8E sebagai *access point*, ditandai dengan label N (Network). Selain itu ada pula perangkat wireless dengan MAC Address 00:0C:42:6A:E7:6E yang berfungsi sebagai *wireless client*. Keduanya menggunakan SSID "Kung Fu HotSpot".

Bila Anda ingin menggunakan *command line*, maka perintah berikut dapat Anda gunakan untuk menjalankan Snooper.

```
[admin@CPE-1] > interface wireless snoop snoop wlan2
BAND      FREQ    USE      BW NET-COUNT NOISE-FLOOR STA-COUNT
2ghz-g    2412MHz 1.3%    11.5kbps    1      -101      2
2ghz-g    2417MHz 0.6%     5.8kbps    0      -101      0
2ghz-g    2422MHz 0%        0bps      0      -101      0
2ghz-g    2427MHz 0%        0bps      0      -101      0
2ghz-g    2432MHz 0%        0bps      0      -101      0
2ghz-g    2437MHz 0%        0bps      0      -101      0
2ghz-g    2442MHz 0%        0bps      0      -100      0
2ghz-g    2447MHz 0%        0bps      0      -101      0
2ghz-g    2452MHz 0%        0bps      0      -102      0
2ghz-g    2457MHz 0%        0bps      0      -100      0
2ghz-g    2462MHz 0%        0bps      0      -101      0
-- [Q quit|D dump|C-z pause|n networks|s stations]
```

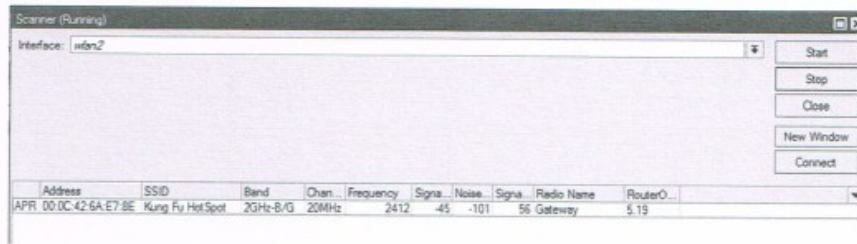
Scan

Selain fitur *Snooper*, Anda juga dapat menggunakan tool *Scan* yang dapat digunakan pada Router yang berfungsi sebagai *wireless client* untuk melihat SSID apa yang dikonfigurasi pada *access point*. Anda dapat menggunakan scanner ini melalui menu *Wireless* → pilih interface yang akan digunakan → tab *Wireless* → tombol *Scan*, seperti pada gambar berikut ini.



Gambar 4.34
Fitur Scanner pada interface wlan

Sebagai contoh, Scan yang dijalankan pada salah satu interface Router CPE akan terlihat seperti berikut ini:



Gambar 4.35
Hasil Scan terhadap Access Point Gateway

4

Dengan menggunakan command line Anda dapat menggunakan perintah seperti berikut ini:

```
[admin@CPE] > interface wireless scan wlan2
Flags: A - active, P - privacy, R - routeros-network, N - nstreme, T - tdma, W -
wds, B - bridge

ADDRESS          SSID      BAND   CHANNEL-  FREQ  SIG  NF  SNR  RADIO-N
APRB 00:0C:42:6A:E7:8E Kung Fu  2ghz-g  20mhz   2412 -45 -101 56 Gateway
-- [Q quit|D dump|C-z pause]
```

---END OF CHAPTER ---

Halaman ini sengaja dikosongkan

Belanja Online ?

www.TokoMahal.com

Bab 5

Firewall (Filter)

5

Pada buku *MikroTik Kung Fu Kitab 1*, saya sudah membahas fitur firewall pada satu bab khusus. Namun pada bab itu, pembahasan hanya meliputi konfigurasi sederhana untuk membuat jaringan lokal dapat mengakses Internet. Buku tersebut juga telah membahas sedikit konfigurasi tentang bagaimana mengamankan Router MikroTik Anda.

Adapun pembahasan konfigurasi firewall pada buku *MikroTik Kung Fu Kitab 2* ini adalah konfigurasi yang lebih mendalam lagi. Itu juga yang menyebabkan akhirnya saya memutuskan untuk menulis materi firewall menjadi 3 (tiga) bab terpisah. Anda sudah mengetahui bahwa firewall pada Router MikroTik memiliki tiga fitur utama, yaitu Filter, NAT dan Mangle. Ketiga fitur itu akan dibahas pada bab terpisah. Dan untuk Bab 5 ini, yang akan dibahas adalah fitur *filter*.

Tentunya masih ada fitur lain seperti Layer 7 (L7) Protocol yang sering digunakan, namun pada buku ini fitur tersebut belum dibahas J.

Sebelumnya saya mengharapkan Anda sudah membaca dan memahami pembahasan firewall pada buku *MikroTik Kung Fu Kitab 1*, karena pada buku ini saya hanya melanjutkan pembahasan firewall, saya tidak akan mengulang apa yang sudah dijelaskan pada buku tersebut.

Konsep Paket Filtering

Firewall berfungsi meningkatkan keamanan jaringan dengan cara menentukan paket data apa saja yang bisa masuk maupun keluar dari jaringan tersebut. Untuk menentukan paket mana yang akan diterima (*accept*) atau dibuang (*drop*), firewall akan memeriksa *header* dari sebuah IP paket. Umumnya yang diperiksa pada *header* adalah IP Address pengirim (*src-address*), IP Address tujuan (*dst-address*), jenis protokol, port pengirim (*src-port*), port tujuan (*dst-*

port) bahkan *fragmentation* yang terjadi pada paket juga dapat diperiksa oleh firewall. Beberapa firewall malah melakukan pemeriksaan pada isi dari data yang dikirimkan, jenis firewall ini disebut *Layer 7 (L7) Firewall*.

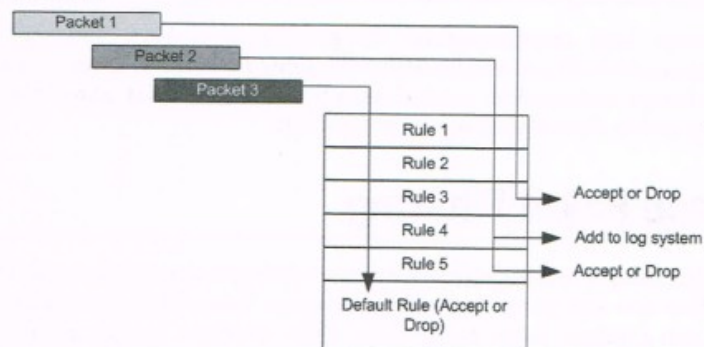
Saran saya, jika Anda ingin mahir bermain firewall, pelajarilah dahulu ilmu TCP/IP dengan baik. Permainan firewall adalah permainan pemeriksaan IP paket. Bagaimana mungkin Anda memeriksa paket kalau Anda sendiri tidak tahu apa yang akan Anda periksa. Namun, Anda jangan mundur untuk membaca buku ini, khususnya tentang firewall, walaupun ilmu TCP/IP Anda masih kurang. Anda dapat saja menggunakan "alur maju mundur" untuk mempelajari sesuatu. Belajar firewall dulu, untuk kemudian belajar TCP/IP, walaupun yang paling baik adalah belajar TCP/IP dulu baru kemudian belajar firewall.

The Rule

5

Kembali ke firewall I, Bagaimana firewall memeriksa *IP packet* (paket IP)? Untuk memeriksa paket, Anda harus membuat rule (aturan) firewall. Dan dalam rule tersebut terdapat parameter-parameter yang akan Anda gunakan untuk memeriksa paket data. Parameter-parameter itu adalah *src-address*, *dst-address*, *protocol*, *src-port*, *dst-port* dan lain-lain.

Paket yang masuk ke dalam firewall akan dicocokkan dengan rule-rule yang sudah Anda buat tadi. Bila ada paket yang cocok dengan rule yang sudah dibuat, maka barulah firewall akan menentukan tindakan (*action*) apa yang akan diambil, apakah paket tersebut akan diterima (*accept*) atau akan dibuang (*drop*). Anda dapat melihat gambaran antara paket dan rule firewall pada gambar berikut.



Gambar 5.1
Rule pada firewall mengeksekusi paket

Pada gambar di atas, Anda dapat melihat ada 3 (tiga) buah paket yang akan diproses oleh firewall pada Router MikroTik. Router MikroTik tersebut telah memiliki 5 (lima) buah rule dan sebuah rule default. Mari kita lihat perlakuan dari masing-masing paket tersebut.

- ➡ **Packet 1**, pada saat paket ini diterima, firewall menggunakan *Rule 1* untuk memeriksanya. Ternyata *header* pada *Paket 1* tidak cocok dengan parameter yang ada di *Rule 1*, paket tersebut diberikan lagi ke *Rule 2*. Sama seperti sebelumnya, ternyata *header* yang ada pada paket tersebut tidak cocok dengan *Rule 2*, kembali paket diberikan ke *Rule 3*. Ternyata *header* pada paket tersebut cocok dengan *Rule 3*, disinilah firewall mengambil *action*, apakah paket tersebut akan di-*accept* atau di-*drop*.
Jika sebuah paket telah cocok dengan suatu rule, maka paket tersebut tidak akan lagi diberikan kepada rule selanjutnya. Anda dapat melihat bahwa *Paket 1* tidak lagi diberikan kepada *Rule 4* setelah dieksekusi oleh *Rule 3*.
- ➡ **Packet 2**, Anda dapat melihat bahwa *header* dari paket tersebut barulah cocok dengan *Rule 4*. Namun Anda melihat bahwa *Paket 2* tersebut masih diteruskan lagi ke rule berikut (*Rule 5*) untuk menjalani pemeriksaan lagi. Untuk situasi tertentu, Anda dapat saja mengkonfigurasi agar sebuah paket dapat diperiksa oleh beberapa rule.
- ➡ **Packet 3**, paket ini merupakan paket dengan *header* yang tidak memiliki kecocokan dengan rule yang ada. Dengan kata lain, tidak ada rule yang berhasil memeriksa paket 3. Jika ini terjadi, maka paket tersebut akan diproses oleh default rule. Pada umumnya default rule ini tidak terlihat (*implicit*), Anda tidak akan menemukan rule tersebut pada firewall.
Untuk fitur filter pada Router MikroTik, default rule yang digunakan adalah *accept all*, artinya bila ada paket yang tidak cocok dengan rule yang ada sebelumnya, maka paket tersebut akan diterima atau diizinkan (*accept*). Begitu juga jika Anda tidak mengkonfigurasi apapun pada filter Router MikroTik, semua paket akan diterima.

Dari gambar dan pembahasan di atas, terlihat bahwa firewall akan menggunakan rule dari secara berurutan, dari *Rule 1* sampai dengan *Rule 5*, dari atas ke bawah (*top to bottom*). Itulah sebabnya urutan rule dalam firewall sangat penting. Urutan rule kurang tepat, maka firewall Anda akan melempem. Contoh urutan rule yang dilihat dengan menggunakan WinBox dari sebuah firewall adalah sebagai berikut:

#	Action	Chain	Src. Address	Dest. Address	Protocol	Src. Port	Dest. Port	In. Inter.	Out. Int.	Bytes	Packets
0	✓ allow	input	192.168.1.2		6 (tcp)		8291	ether2		15.7 KB	166
1	✗ drop	input			6 (tcp)		20	ether2		0 B	0
2	✗ drop	input			6 (tcp)		21	ether2		0 B	0
3	✗ drop	input			6 (tcp)		22	ether2		0 B	0
4	✗ drop	input			6 (tcp)		23	ether2		0 B	0
5	✗ drop	input			6 (tcp)		53	ether2		0 B	0
6	✗ drop	input			6 (tcp)		80	ether2		0 B	0
7	✗ drop	input			6 (tcp)		2000	ether2		0 B	0
8	✗ drop	input			6 (tcp)		8291	ether2		0 B	0
9	✓ allow	input			1 (icmp)			ether2		280 B	5
10	✗ drop	input			1 (icmp)			ether2		532 B	7

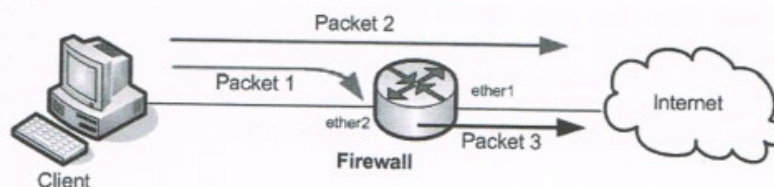
Gambar 5.2
Contoh rule pada sebuah firewall

Asal dan Tujuan Paket

5

Untuk mengkonfigurasi firewall, Anda juga harus mengetahui asal dari sebuah paket dan ke mana paket tersebut akan pergi. Kesalahan menentukan asal dan tujuan paket akan menyebabkan rule yang dibuat tidak akan dapat “menjaring” paket yang keluar dan masuk ke dalam jaringan. Gambar 5.3 memperlihatkan beberapa paket berdasarkan arah dan tujuan.

Firewall bekerja dengan menggunakan parameter chain yang nantinya akan digunakan saat akan membuat rule. Chain tersebut akan digunakan untuk melihat asal dan tujuan dari paket yang diterima oleh firewall. Pada Filter Router MikroTik, secara default terdapat 3 (tiga) chain, yaitu *input*, *forward* dan *output*.



Gambar 5.3
Paket berdasarkan asal dan tujuan

Mari kita lihat penjelasan untuk masing-masing paket yang ada pada gambar 5.3 :

- ➡ **Packet 1**, adalah paket yang berasal dari komputer client dan ditujukan bagi interface *ether2* dari Router MikroTik. Contoh paket ini misalnya pada saat Anda melakukan ping ke Router MikroTik, ataupun saat

melakukan konfigurasi baik dengan menggunakan WinBox maupun melalui WebFig, Telnet maupun SSH. Paket ini nantinya akan ditangani oleh *chain input*. Begitu juga jika ada paket yang nantinya berasal dari komputer client dan ditujukan bagi interface *ether1*, *chain input* juga yang akan melakukan *filtering*.

- **Packet 2**, adalah paket yang berasal dari komputer client dan menuju Internet. Paket ini sering disebut paket yang melintasi firewall. Paket ini nantinya akan ditangani oleh *chain forward*. Perlu Anda perhatikan, bahwa paket *forward* terlihat akan masuk juga melalui interface *ether2*.

Walaupun masuk melalui interface *ether2*, Router MikroTik tidak akan menggunakan *chain input* untuk memeriksa paket tersebut. Ini dikarenakan Router MikroTik tidak melihat bahwa paket ini ditujukan bagi interface-interface-nya, paket ini hanya melintasi atau melewati router. Nantinya akan ada dua jenis paket *forward* pada topologi tadi, yaitu dari komputer client ke Internet dan dari Internet ke komputer client. Contoh paket ini adalah pada saat Anda melakukan akses Internet dari komputer client.

- **Packet 3**, paket ini berasal dari router itu sendiri dan ditujukan bagi host yang ada di Internet, bisa juga merupakan paket dari router yang ditujukan bagi komputer client. Paket ini akan ditangani oleh *chain output*. Contoh paket ini adalah paket yang dihasilkan oleh Web Proxy jika Anda mengaktifkan Internal Web Proxy yang dimiliki oleh Router MikroTik. Jika Anda melakukan ping dari Router MikroTik, maka paket yang dihasilkan juga merupakan paket yang akan ditangani oleh *chain output*.

Pemahaman asal dan tujuan harus Anda kuasai dengan benar, begitu juga dengan konsep arah paket. Misalnya, Anda melakukan ping dari Router MikroTik kepada komputer client. *Ping request* dari Router akan ditangani oleh *chain output*, sedangkan *ping reply* yang keluar dari komputer client dan menuju kembali ke router akan ditangani oleh *chain input*.

Chain input

Chain input berperan untuk melakukan filter terhadap paket-paket yang ditujukan bagi interface-interface router. Ini berguna untuk membatasi akses terhadap Router MikroTik Anda, misalnya jika ingin dilakukan pembatasan akses konfigurasi. Dalam dunia *network security* sering diistilahkan dengan teknik membatasi akses terhadap port-port router yang dapat menimbulkan celah keamanan.

Secara default, baik Router MikroTik yang di-install pada personal computer maupun RouterBoard akan membuka port-port untuk keperluan konfigurasi. Misalnya port 8291 yang digunakan jika Anda ingin melakukan konfigurasi dengan WinBox maupun port 80 jika konfigurasi ingin dilakukan menggunakan web based (WebFig).

Nmap

Untuk mengetahui port-port apa saja yang terbuka dari Router MikroTik, aplikasi *nmap* sangat bermanfaat. Banyak para professional IT yang berkecimpung dalam bidang *network security* yang mengandalkan aplikasi ini. Anda dapat mendownloadnya pada link berikut <http://nmap.org/download.html>. Aplikasi ini dapat dijalankan pada mesin Windows, Linux, MAC maupun UNIX. Anda tinggal mendownload file instalasinya dan melakukan prosedur instalasi seperti umumnya pada aplikasi lain.

Contoh berikut memperlihatkan hasil *scanning* terhadap port apa saja yang terbuka pada interface sebuah Router MikroTik, dimana IP Address dari interface Router tersebut adalah 192.168.1.1. Perintah *nmap* yang dapat digunakan adalah *nmap -sS 192.168.1.1* seperti berikut.

```
C:\Users\Admin>nmap -sS 192.168.1.1

Starting Nmap 5.51 ( http://nmap.org ) at 2012-10-03 08:31 Taipei Standard Time
Nmap scan report for 192.168.1.1
Host is up (0.0033s latency).
Not shown: 992 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
53/tcp    open  domain
80/tcp    open  http
2000/tcp  open  cisco-sccp
8291/tcp  open  unknown
MAC Address: 00:0C:42:76:58:80 (Routerboard.com)

Nmap done: 1 IP address (1 host up) scanned in 3.80 seconds
```

Hasil pemeriksaan port di atas memperlihatkan sederatan port yang terbuka pada Router MikroTik. Perhatikanlah parameter *state* (status) dari hasil pengujian di atas, semuanya berstatus *open* (terbuka). Selain itu terdapat 992 port dalam kondisi tertutup (*not shown: 992 closed ports*).

melakukan konfigurasi baik dengan menggunakan WinBox maupun melalui WebFig, Telnet maupun SSH. Paket ini nantinya akan ditangani oleh *chain input*. Begitu juga jika ada paket yang nantinya berasal dari komputer client dan ditujukan bagi interface *ether1*, *chain input* juga yang akan melakukan *filtering*.

- **Packet 2**, adalah paket yang berasal dari komputer client dan menuju Internet. Paket ini sering disebut paket yang melintasi firewall. Paket ini nantinya akan ditangani oleh *chain forward*. Perlu Anda perhatikan, bahwa paket forward terlihat akan masuk juga melalui interface *ether2*. Walaupun masuk melalui interface *ether2*, Router MikroTik tidak akan menggunakan *chain input* untuk memeriksa paket tersebut. Ini dikarenakan Router MikroTik tidak melihat bahwa paket ini ditujukan bagi interface-interfacesnya, paket ini hanya melintasi atau melewati router. Nantinya akan ada dua jenis paket forward pada topologi tadi, yaitu dari komputer client ke Internet dan dari Internet ke komputer client. Contoh paket ini adalah pada saat Anda melakukan akses Internet dari komputer client.
- **Packet 3**, paket ini berasal dari router itu sendiri dan ditujukan bagi host yang ada di Internet, bisa juga merupakan paket dari router yang ditujukan bagi komputer client. Paket ini akan ditangani oleh *chain output*. Contoh paket ini adalah paket yang dihasilkan oleh Web Proxy jika Anda mengaktifkan Internal Web Proxy yang dimiliki oleh Router MikroTik. Jika Anda melakukan ping dari Router MikroTik, maka paket yang dihasilkan juga merupakan paket yang akan ditangani oleh *chain output*.

Pemahaman asal dan tujuan harus Anda kuasai dengan benar, begitu juga dengan konsep arah paket. Misalnya, Anda melakukan ping dari Router MikroTik kepada komputer client. *Ping request* dari Router akan ditangani oleh *chain output*, sedangkan *ping reply* yang keluar dari komputer client dan menuju kembali ke router akan ditangani oleh *chain input*.

Chain input

Chain input berperan untuk melakukan filter terhadap paket-paket yang ditujukan bagi interface-interface router. Ini berguna untuk membatasi akses terhadap Router MikroTik Anda, misalnya jika ingin dilakukan pembatasan akses konfigurasi. Dalam dunia *network security* sering diistilahkan dengan teknik membatasi akses terhadap port-port router yang dapat menimbulkan celah keamanan.

port 8291 tidak boleh Anda filter dengan *action drop*, karena sesuai skenario, port ini akan digunakan oleh WinBox.

Adapun perintah yang dapat Anda gunakan adalah sebagai berikut :

5

```
[admin@R1] > ip firewall filter add chain=input in-interface=ether2 \
    protocol=tcp dst-port=20 action=drop

[admin@R1] > ip firewall filter add chain=input in-interface=ether2 \
    protocol=tcp dst-port=21 action=drop

[admin@R1] > ip firewall filter add chain=input in-interface=ether2 \
    protocol=tcp dst-port=22 action=drop

[admin@R1] > ip firewall filter add chain=input in-interface=ether2 \
    protocol=tcp dst-port=23 action=drop

[admin@R1] > ip firewall filter add chain=input in-interface=ether2 \
    protocol=tcp dst-port=53 action=drop

[admin@R1] > ip firewall filter add chain=input in-interface=ether2 \
    protocol=tcp dst-port=80 action=drop

[admin@R1] > ip firewall filter add chain=input in-interface=ether2 \
    protocol=tcp dst-port=2000 action=drop

[admin@R1] > ip firewall filter print
Flags: X - disabled, I - invalid, D - dynamic
 0 chain=input action=drop protocol=tcp in-interface=ether2 dst-port=20
 1 chain=input action=drop protocol=tcp in-interface=ether2 dst-port=21
 2 chain=input action=drop protocol=tcp in-interface=ether2 dst-port=22
 3 chain=input action=drop protocol=tcp in-interface=ether2 dst-port=23
 4 chain=input action=drop protocol=tcp in-interface=ether2 dst-port=53
 5 chain=input action=drop protocol=tcp in-interface=ether2 dst-port=80
 6 chain=input action=drop protocol=tcp in-interface=ether2 dst-port=2000
```

Pada konfigurasi di atas, Anda tidak perlu menambahkan rule dengan konfigurasi *protocol=tcp dst-port=8291 action=accept* untuk mengizinkan akses konfigurasi dengan menggunakan WinBox. Bukankah secara default rule filter

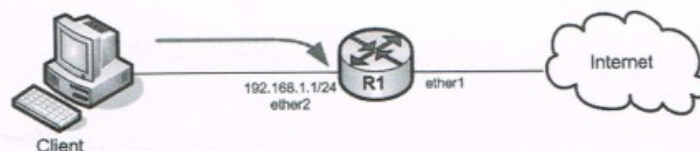
Firewall Tactic

Bagaimana menyusun konfigurasi rule-rule firewall dengan baik? Sangat sulit untuk menjawabnya, setiap Administrator jaringan mempunyai "taste" sendiri-sendiri untuk menyusun rule firewall. Walaupun tujuan akhir dari firewall tersebut adalah sama, tetap saja cara penyusunan rule-nya akan berbeda. Chef di restoran-restoran saja memiliki "taste" sendiri-sendiri, kenapa Anda tidak coba membuat "taste" sendiri pada susunan rule firewall.

Sebagai panduan untuk menyusun firewall, saya membagi dua pendekatan taktik dalam penyusunan rule firewall, yaitu :

- **Taktik 1**, Anda membuang (*drop*) beberapa paket yang tidak dibutuhkan, kemudian Anda menerima (*accept*) semua paket. Istilahnya "*drop some and accept all*"
- **Taktik 2**, Anda menerima (*accept*) beberapa paket yang Anda butuhkan, kemudian membuang (*drop*) semua paket. Istilahnya "*accept some and drop all*"

Bagaimana jika kedua taktik tersebut diterapkan dalam bentuk konfigurasi? Sebagai contoh implementasi, Anda dapat melihat gambar 5.4, dimana firewall dengan *chain input* akan diterapkan pada interface *ether2* (192.168.1.1). Dalam skenario ini diinginkan Router tersebut hanya dapat dikonfigurasi dengan menggunakan WinBox, tidak dengan cara-cara konfigurasi lain (misalnya Telnet, SSH maupun WebFig). Kesimpulannya, port yang akan dibuka hanyalah port 8291.



Gambar 5.4
Filter akan diterapkan pada interface ether2

Taktik 1

Jika taktik pertama yang akan Anda gunakan maka, Anda terlebih dahulu harus melakukan filter pada port-port yang terbuka dengan menerapkan *action drop* bagi paket yang ingin masuk melalui port tersebut. Berdasarkan hasil *scanning* dengan menggunakan *nmap* pada pembahasan sebelumnya maka port yang harus Anda filter adalah 20, 21, 22, 23, 53, 2000 dengan protokol TCP. Adapun

Perhatikanlah bahwa status dari beberapa port yang sudah diterapkan firewall dengan *action=drop* akan terlihat *filtered*, sedangkan port 8291 yang digunakan oleh WinBox masih dalam status *open*.

Taktik 2

Jika taktik kedua yang akan digunakan, maka Anda cukup memberikan akses kepada port 8291 dengan parameter *action=accept*. Jika Anda menjalankan DNS Server pada Router MikroTik, maka akses terhadap port 53 dengan protocol UDP juga harus dikonfigurasi dengan parameter *action=accept*. Kemudian dilanjutkan dengan melakukan filter terhadap semua port yang tidak dibutuhkan dengan parameter *action=drop*. Contoh perintah yang dapat digunakan adalah sebagai berikut.

5

```
[admin@R1] > ip firewall filter add chain=input in-interface=ether2 \
               protocol=tcp dst-port=8291 action=accept

[admin@R1] > ip firewall filter add chain=input in-interface=ether2 \
               protocol=udp dst-port=53 action=accept

[admin@R1] > ip firewall filter add chain=input in-interface=ether2 \
               action=drop

[admin@R1] > ip firewall filter print
Flags: X - disabled, I - invalid, D - dynamic
 0 chain=input action=accept protocol=tcp in-interface=ether2
   dst-port=8291
 1 chain=input action=accept protocol=udp in-interface=ether2
   dst-port=53
 2 chain=input action=drop in-interface=ether2
```

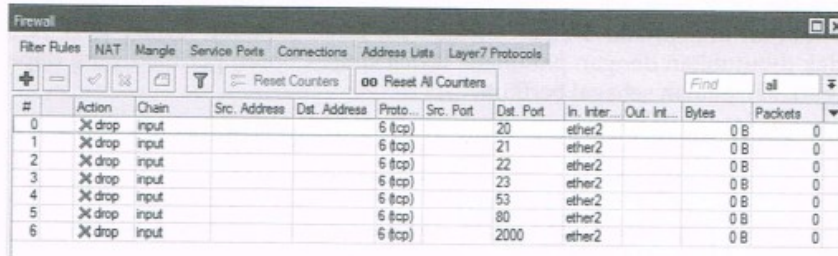
Adapun hasil dari konfigurasi ini akan sama dengan apa yang telah dilakukan pada taktik pertama tadi. Anda tinggal memilih taktik apa yang akan digunakan, atau dapat mencoba untuk menggabungkan kedua taktik tersebut. Namun yang perlu diperhatikan, bahwa semakin sedikit rule firewall yang dibuat, maka semakin baik firewall tersebut, tentunya tanpa mengurangi aspek keamanan yang ingin diterapkan.

Rule firewall yang banyak akan menguras memori router dan juga akan memperlambat proses pengolahan paket data, karena paket harus dicocokkan pada sederetan rule yang telah dibuat.

Jika dilihat dengan menggunakan WinBox, maka konfigurasi dengan menggunakan taktik 2 di atas akan terlihat seperti berikut:

pada Router MikroTik sudah mengizinkan semuanya (*accept all*). Jadi, jika rule untuk akses WinBox tidak dituliskan maka sebenarnya akses tersebut sudah diizinkan.

Saya tidak lagi menampilkan bagaimana mengkonfigurasi firewall dengan menggunakan WinBox. Jika sudah membaca buku *MikroTik Kung Fu Kitab 1*, tentunya Anda masih ingat menu mana yang dapat digunakan untuk mengkonfigurasi firewall. Namun untuk memperlihatkan hasil konfigurasi tadi, Anda dapat melihatnya pada gambar berikut ☺.



#	Action	Chain	Src. Address	Dst. Address	Proto	Src. Port	Dst. Port	In. Inter	Out. Int.	Bytes	Packets
0	drop	input			tcp		20	ether2		0	0
1	drop	input			tcp		21	ether2		0	0
2	drop	input			tcp		22	ether2		0	0
3	drop	input			tcp		23	ether2		0	0
4	drop	input			tcp		53	ether2		0	0
5	drop	input			tcp		80	ether2		0	0
6	drop	input			tcp		2000	ether2		0	0

Gambar 5.5
Susunan rule dengan "drop some and accept all"

5

Hasil pengujian terhadap konfigurasi di atas dengan menggunakan *nmap* dapat Anda lihat seperti berikut.

```
C:\Users\Admin> nmap -sS 192.168.1.1

Starting Nmap 5.51 ( http://nmap.org ) at 2012-10-04 11:51 Taipei Standard Time
Nmap scan report for 192.168.1.1
Host is up (0.0052s latency).
Not shown: 992 closed ports
PORT      STATE      SERVICE
20/tcp    filtered  ftp-data
21/tcp    filtered  ftp
22/tcp    filtered  ssh
23/tcp    filtered  telnet
53/tcp    filtered  domain
80/tcp    filtered  http
2000/tcp  filtered  cisco-sccp
8291/tcp  open      unknown
MAC Address: 00:0C:42:76:58:80 (Routerboard.com)

Nmap done: 1 IP address (1 host up) scanned in 20.66 seconds
```

Firewall Tactics

Untuk menyusun rule-rule firewall dengan menggunakan *chain forward*, Anda juga dapat menggunakan dua pendekatan firewall taktik yang sudah dibahas pada *chain input*. Sebagai contoh, marilah kita menyusun rule-rule firewall dengan hasil akhir yang ingin dicapai adalah mengizinkan komputer client mendapatkan layanan Internet browsing (HTTP/HTTPS). Komputer client juga diizinkan untuk melakukan ping (ICMP) ke Internet.

Taktik 1

Taktik firewall pertama yang dapat digunakan adalah *drop some and accept all*. Di sini Anda harus membuang *traffic-traffic* yang tidak dibutuhkan dengan menggunakan *action=drop*, misalnya paket-paket yang ditimbulkan oleh virus atau worm. Ada banyak paket-paket yang dihasilkan oleh virus/worm, namun saya hanya memberikan contoh beberapa saja.

Setelah tidak memberikan akses ke Internet terhadap paket-paket yang kemungkinan ditimbulkan oleh virus/worm (*drop some*), selanjutnya Anda baru memberikan akses seluas-luasnya kepada komputer client untuk mendapatkan layanan Internet (*accept all*), tentunya dengan menggunakan *action=accept*.

Adapun contoh konfigurasi yang dapat digunakan adalah sebagai berikut.

```
[admin@R1] > ip firewall filter add chain=forward protocol=tcp \
dst-port=135-139 action=drop

[admin@R1] > ip firewall filter add chain=forward protocol=tcp \
dst-port=1433-1434 action=drop

[admin@R1] > ip firewall filter add chain=forward protocol=tcp \
dst-port=275 action=drop

[admin@R1] > ip firewall filter add chain=forward protocol=tcp \
dst-port=2283 action=drop

[admin@R1] > ip firewall filter add chain=forward protocol=tcp \
dst-port=2535 action=drop

[admin@R1] > ip firewall filter add chain=forward protocol=tcp \
dst-port=4444 action=drop

[admin@R1] > ip firewall filter add chain=forward protocol=udp \
dst-port=4444 action=drop
```

#	Action	Chain	Src. Address	Dest. Address	Protocol	Src. Port	Dest. Port	In. Inter...	Out. Int...	Bytes	Packets
0	✓ acc	input			6 (tcp)		8291	ether2		0 B	0
1	✓ acc	input			17 (udp)		53	ether2		196 B	3
2	✗ drop	input						ether2		15.1 KB	197

Gambar 5.6
Susunan rule dengan "accept some and drop all"

Sedangkan pengujian dengan menggunakan *nmap* akan mendapatkan hasil yang sedikit berbeda, namun tujuan akhir firewall yang diinginkan sebenarnya relatif sama. Anda dapat melihat uraian berikut yang memperlihatkan status dari port 8291 adalah terbuka (*open*).

```
C:\Users\Admin> nmap -ss 192.168.1.1

Starting Nmap 5.51 ( http://nmap.org ) at 2012-10-04 12:09 Taipei Standard Time
Nmap scan report for 192.168.1.1
Host is up (0.00023s latency).
Not shown: 999 filtered ports
PORT      STATE SERVICE
8291/tcp  open  unknown
MAC Address: 00:0C:42:76:58:80 (Routerboard.com)

Nmap done: 1 IP address (1 host up) scanned in 26.33 seconds
```

5

Chain forward

Chain forward bertugas untuk melakukan filter terhadap paket yang akan melintasi router. Chain inilah yang sebenarnya dapat menentukan layanan Internet apa yang bisa dinikmati oleh komputer client pada suatu jaringan lokal (LAN), tentunya dapat juga digunakan untuk membatasi layanan tersebut.

Karena digunakan untuk melakukan filter terhadap paket yang akan melintasi router, maka harus diperhatikan arah dari paket tersebut. Apakah paket tersebut berasal dari komputer client menuju Internet, atau sebaliknya, dari Internet menuju komputer client.

Seperti yang kita ketahui bersama bahwa komunikasi jaringan komputer adalah komunikasi dua arah (*two ways*), ada *send* dan ada *receive*, ada *request* dan ada *reply*, ada *upload* dan ada *download*, ada *unggah* dan ada *unduh*, ada *siang* dan ada *malam*...loh !!! ☺

Contoh berikut memperlihatkan rule firewall yang hanya mengizinkan komputer client mendapatkan layanan HTTP (*dst-port=80*), HTTPS (*dst-port=443*) dan DNS (*dst-port=53*) ke Internet. Selain itu komputer client juga diizinkan melakukan ping (ICMP) ke Internet. Adapun perintah yang dapat digunakan adalah sebagai berikut.

```
[admin@R1] > ip firewall filter add chain=forward protocol=tcp \
dst-port=80 action=accept

[admin@R1] > ip firewall filter add chain=forward protocol=tcp \
src-port=80 action=accept

[admin@R1] > ip firewall filter add chain=forward protocol=tcp \
dst-port=443 action=accept

[admin@R1] > ip firewall filter add chain=forward protocol=tcp \
src-port=443 action=accept

[admin@R1] > ip firewall filter add chain=forward protocol=udp \
dst-port=53 action=accept

[admin@R1] > ip firewall filter add chain=forward protocol=udp \
src-port=53 action=accept

[admin@R1] > ip firewall filter add chain=forward protocol=icmp \
action=accept

[admin@R1] > ip firewall filter add chain=forward action=drop

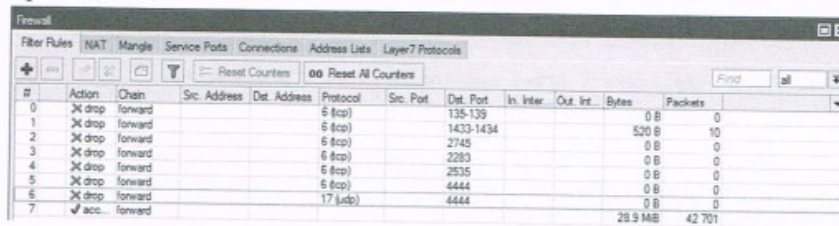
[admin@R1] > ip firewall filter print
Flags: X - disabled, I - invalid, D - dynamic
0 chain=forward action=accept protocol=tcp dst-port=80
1 chain=forward action=accept protocol=tcp src-port=80
2 chain=forward action=accept protocol=tcp dst-port=443
3 chain=forward action=accept protocol=tcp src-port=443
4 chain=forward action=accept protocol=udp dst-port=53
5 chain=forward action=accept protocol=udp src-port=53
6 chain=forward action=accept protocol=icmp
7 chain=forward action=drop
```

Jika dilihat dengan menggunakan WinBox, maka susunan rule firewall di atas akan terlihat seperti berikut.

```
[admin@R1] > ip firewall filter add chain=forward action=accept

[admin@R1] > ip firewall filter print
Flags: X - disabled, I - invalid, D - dynamic
 0 chain=forward action=drop protocol=tcp dst-port=135-139
 1 chain=forward action=drop protocol=tcp dst-port=1433-1434
 2 chain=forward action=drop protocol=tcp dst-port=2745
 3 chain=forward action=drop protocol=tcp dst-port=2283
 4 chain=forward action=drop protocol=tcp dst-port=2535
 5 chain=forward action=drop protocol=tcp dst-port=4444
 6 chain=forward action=drop protocol=udp dst-port=4444
 7 chain=forward action=accept
```

Jika dilihat dengan menggunakan WinBox, maka rule-rule di atas akan terlihat seperti berikut.



#	Action	Chain	Src. Address	Dest. Address	Protocol	Src. Port	Dest. Port	In. Inter.	Out. Inter.	Bytes	Packets
0	X drop	forward			6 (tcp)		135-139			0 B	0
1	X drop	forward			6 (tcp)		1433-1434			520 B	10
2	X drop	forward			6 (tcp)		2745			0 B	0
3	X drop	forward			6 (tcp)		2283			0 B	0
4	X drop	forward			6 (tcp)		2535			0 B	0
5	X drop	forward			6 (tcp)		4444			0 B	0
6	X drop	forward			17 (udp)		4444			0 B	0
7	✓ acc.	forward								28.9 MB	42 701

Gambar 5.7

Chain forward yang menganut aliran "drop some and accept all"

Taktik di atas memiliki keuntungan, Anda tidak perlu repot-repot untuk membuka akses ke Internet jika ternyata pengguna jaringan (user) ingin menjalankan berbagai macam aplikasi Internet, apalagi jika aplikasi tersebut merupakan aplikasi yang tidak umum digunakan. Namun taktik di atas juga memiliki kelemahan, Anda harus rajin-rajin memonitor aktifitas jaringan, siapa tahu ada virus/worm terbaru yang bekerja pada port baru pula, tidak seperti port yang sudah dikonfigurasi pada firewall.

Taktik 2

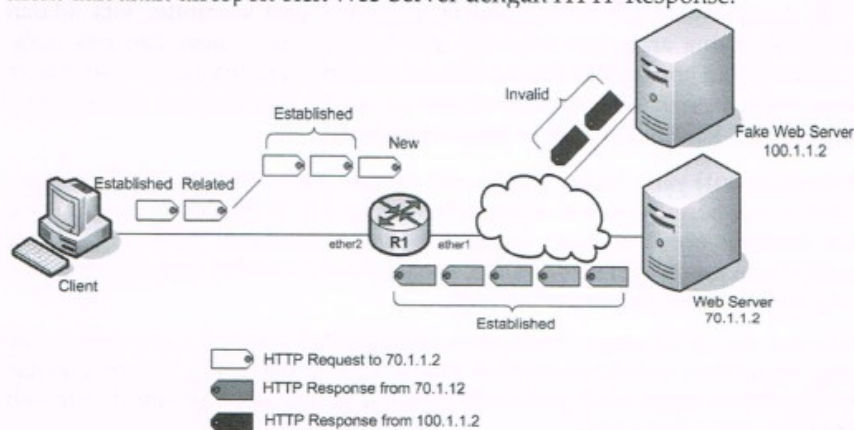
Jika taktik *accept some and drop all* yang akan digunakan, maka yang harus dilakukan terlebih dahulu adalah memberikan akses terhadap beberapa layanan Internet dengan menggunakan *action=accept*. Kemudian Anda lanjutkan dengan menutup semua akses ke Internet dengan menggunakan *action=drop*.

Connection State

Saat membangun firewall, Anda juga harus memperhatikan status koneksi (*connection state*) dari sebuah paket data. Ada 4 (empat) *connection state* yang dapat dimiliki sebuah paket, yaitu :

- *New*, paket dengan status ini menunjukkan bahwa paket tersebut merupakan paket pertama (*new*) dari sebuah koneksi. Kita ketahui bahwa suatu pengiriman data terdiri dari banyak paket, dan paket pertama dari serentetan paket yang akan dikirimkan inilah yang akan memiliki status *new*.
- *Established*, paket dengan status ini menunjukkan bahwa paket tersebut merupakan kelanjutan dari paket yang memiliki status *New* tadi.
- *Related*, paket ini merupakan paket baru (*new*), tapi sebenarnya merupakan kelanjutan (sambungan) dari koneksi yang telah ada sebelumnya.
- *Invalid*, paket ini adalah paket yang tidak memiliki hubungan dengan paket lain maupun koneksi lain.

Anda dapat memperhatikan gambar berikut untuk melihat *connection state* dari paket-paket yang dikirimkan oleh sebuah komputer client yang sedang browsing ke sebuah Web Server. HTTP Request akan dikirimkan oleh komputer client dan akan direspon oleh Web Server dengan HTTP Response.



Gambar 5.9
Berbagai macam status dari paket

#	Action	Chain	Src. Addr.	Dst. Addr.	Protocol	Src. Port	Dst. Port	In. Inter.	Out. Int.	Bytes	Packets
0	✓ acc.	forward			6 (tcp)		80			479.5 KB	8 163
1	✓ acc.	forward			6 (tcp)	80				19.9 MB	14 726
2	✓ acc.	forward			6 (tcp)		443			45.8 KB	237
3	✓ acc.	forward			6 (tcp)	443				117.7 KB	232
4	✓ acc.	forward			17 (udp)		53			1153 B	18
5	✓ acc.	forward			17 (udp)	53				6.5 KB	18
6	✓ acc.	forward			1 (icmp)					360 B	6
7	✗ drop	forward								123.9 KB	113

Gambar 5.8

Chain forward yang menganut aliran "accept some and drop all"

Perhatikanlah konfigurasi di atas, terutama konfigurasi rule pertama (*dst-port=80*) dan kedua (*src-port=80*). Kedua konfigurasi tersebut bertujuan memberikan layanan HTTP kepada komputer client. Yang menjadi pertanyaan, mengapa harus memakai dua baris konfigurasi? Tidakkah cukup hanya dengan menggunakan satu rule konfigurasi saja? Potongan konfigurasi di atas dapat Anda lihat sebagai berikut.

```
[admin@R1] > ip firewall filter add chain=forward protocol=tcp \
dst-port=80 action=accept

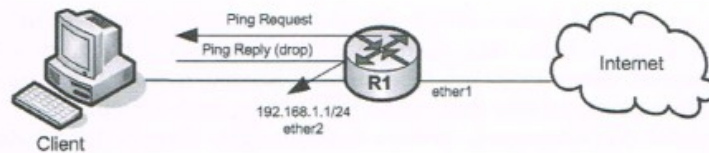
[admin@R1] > ip firewall filter add chain=forward protocol=tcp \
src-port=80 action=accept
```

Perlu Anda ingat, bahwa komunikasi pada jaringan komputer kita adalah komunikasi dua arah, ada *send* dan ada *received*, ada *request* dan ada *reply*. Penjelasan dari konfigurasi di atas adalah rule yang pertama dengan parameter *dst-port=80* dengan *action=accept* ditujukan untuk memberikan akses HTTP Request dari komputer client ke Internet (*upload*).

Sedangkan rule yang kedua adalah rule dengan parameter *src-port=80* dengan *action=accept* berguna untuk memberikan akses HTTP Response yang datang dari Internet untuk menuju komputer client (*download*). Anda juga dapat melihat rule untuk HTTPS dan DNS, yang semuanya terdiri dari sepasang rule yang berkebalikan parameter port-nya.

Pada sub bab berikutnya, Anda akan mempelajari paramater *connection-state* yang digunakan untuk memeriksa status dari setiap paket. Jika parameter ini digunakan maka tidak perlu lagi membuat konfigurasi sebanyak dua kali seperti contoh sebelumnya.

melakukan ping terhadap komputer client. Ini karena pada saat komputer client mengembalikan paket ping kepada Router, rule tersebut akan ikut membuang paket ping (ICMP) yang datang. Padahal rule tersebut semula ditujukan untuk membuang paket ICMP Request yang dikeluarkan oleh komputer client saja. Anda dapat melihat ilustrasinya pada gambar berikut.



Gambar 5.10
Ping Reply dari Client di drop oleh R1

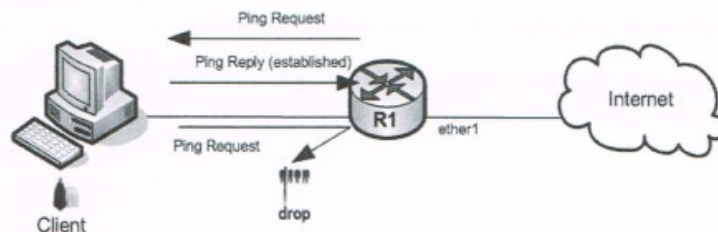
5

Untuk menjamin paket ICMP Reply yang merupakan kelanjutan (*established*) dari ICMP Request tadi bisa diterima oleh R1, maka Anda harus mengkonfigurasi rule dengan menggunakan parameter *connection-state=established* dengan *action=accept*. Maka seharusnya konfigurasi tadi dilengkapi dan hasilnya terlihat seperti berikut.

```
[admin@R1] > ip firewall filter add chain=input protocol=icmp \
connection-state=established action=accept \
[admin@R1] > ip firewall filter add chain=input in-interface=ether2 \
protocol=icmp action=drop
```

```
[admin@R1] > ip firewall filter print
Flags: X - disabled, I - invalid, D - dynamic
0 chain=input action=accept connection-state=established
  protocol=icmp
1 chain=input action=drop protocol=icmp in-interface=ether2
```

Jika sudah menerapkan parameter *connection state*, maka paket ICMP Reply akan diterima oleh R1, seperti terlihat pada gambar berikut :



Gambar 5.11
Ping Reply (*established*) akan diterima oleh R1

Pada gambar di atas, Anda dapat melihat bahwa komputer client membuat dua koneksi baru untuk meminta halaman HTTP pada Web Server. Koneksi yang pertama terdiri dari paket yang berstatus *new*, karena inilah paket yang pertama kali keluar dari komputer client. Kemudian diikuti dua paket lagi yang merupakan kelanjutan dari paket *new* tadi, sehingga status dari paket kedua dan ketiga disebut *established*.

Namun, untuk melakukan HTTP Request tersebut komputer client harus membuat koneksi baru lagi, Anda dapat melihatnya sebagai dua paket HTTP Request dibawah paket sebelumnya. Karena merupakan kelanjutan dari koneksi yang pertama tadi, maka paket yang baru tersebut akan diberi status *related* (berhubungan), artinya berhubungan dengan HTTP Request sebelumnya.

HTTP Request yang dilakukan oleh komputer client akan tiba pada Web Server dengan IP Address 70.1.1.2. Selanjutnya Web Server tersebut akan memberikan halaman HTTP yang diminta, ini dikenal sebagai HTTP Response. Karena HTTP Response ini merupakan hasil dari permintaan yang dilakukan komputer client tadi, maka paket HTTP Request dari Web Server 70.1.1.2 akan diberi status *established* juga.

Selain itu, Anda dapat melihat lagi sebuah Web Server palsu (*fake*), yang juga mencoba memberikan HTTP Response. Karena HTTP Response yang diberikan oleh Web Server 100.1.1.2 tidak diawali dengan HTTP Request kepada dirinya terlebih dahulu, maka paket HTTP Response-nya akan diberi status *invalid*. Inilah artinya, paket tersebut tidak memiliki hubungan apapun dari paket maupun koneksi sebelumnya.

Dalam menerapkan firewall, umumnya rule dengan *connection state* New, Established dan Related akan dikonfigurasi dengan *action=accept*. Sedangkan rule dengan *connection state* Invalid dikonfigurasi dengan *action=drop*.

Marilah kita lihat contoh penerapan *connection state* pada sebuah Router yang mengkonfigurasi filter terhadap protokol ICMP yang akan masuk (*input*) pada interface *ether2*. Ini dimaksudkan agar komputer client tidak dapat melakukan ping terhadap Router.

Contoh perintah yang digunakan adalah sebagai berikut.

```
[admin@R1] > ip firewall filter add chain=input in-interface=ether2 \
                protocol=icmp action=drop
```

Tidak ada yang salah dengan konfigurasi di atas, jika Anda mengujinya maka komputer client tidak akan bisa melakukan ping pada interface *ether2* (*request time out*). Namun konfigurasi di atas juga bisa berakibat Router tidak dapat


```

Flags: X - disabled, I - invalid, D - dynamic
0 chain=forward action=drop connection-state=invalid
1 chain=forward action=accept connection-state=established
2 chain=forward action=accept connection-state=related
3 chain=forward action=accept protocol=udp in-interface=ether2 dst-
  port=53
4 chain=forward action=accept protocol=tcp in-interface=ether2 dst-
  port=80
5 chain=forward action=accept protocol=tcp in-interface=ether2 dst-
  port=443
6 chain=forward action=accept protocol=icmp in-interface=ether2
7 chain=forward action=drop
  
```

Firewall												
Filter Rules		NAT	Mangle	Service Ports	Connections	Address Lists	Layer7 Protocols					
    				 Reset Counters		 Reset All Counters						
#	Action	Chain	S	D	Protocol	Src. Port	Dst. Port	In. Inter.	Out.	Connection State	Bytes	Packets
0	 drop	forward								invalid	0 B	0
1	 acc...	forward								established	1711.1 KB	2 694
2	 acc...	forward								related	268 B	3
3	 acc...	forward			17 (udp)		53	ether2			7.7 KB	124
4	 acc...	forward			6 (tcp)		80	ether2			5.4 KB	106
5	 acc...	forward			6 (tcp)		443	ether2			0 B	0
6	 acc...	forward			1 (icmp)			ether2			60 B	1
7	 drop	forward									7.0 KB	141

Gambar 5.13

Penggunaan connection state pada chain forward

chain tambahan

Secara default firewall filter memiliki 3 (tiga) chain yang sudah tersedia (*built-in*), chain-chain tersebut adalah *input*, *forward* dan *output*. Dalam kondisi-kondisi tertentu Anda dapat membuat chain tambahan yang sesuai dengan kebutuhan dan bebas memberikan nama terhadap chain tersebut.

Dengan menggunakan chain tambahan, Anda bisa menghemat baris-baris firewall. Dengan jumlah baris firewall yang lebih sedikit, maka paket data juga tidak akan melewati pemeriksaan yang begitu panjang. Firewall yang memiliki rule yang panjang dan bersifat *flat* (datar), akan menguras resource router pada saat akan memeriksa sebuah paket. Anda bisa melihat ilustrasi berikut yang memperlihatkan susunan rule firewall yang hanya menggunakan chain *built-in* dan bersifat *flat*.

Connection State pada chain input

Sebagai contoh, Anda dapat melihat susunan rule firewall berikut yang hanya mengizinkan akses WinBox terhadap interface *ether2*. Konfigurasi tersebut juga mengizinkan permintaan DNS Cache yang dilakukan oleh komputer client kepada Router MikroTik melalui interface *ether2*. Susunan rule tidak mengizinkan komputer client melakukan ping terhadap interface *ether2*. Rule-rule tersebut sudah menggunakan parameter *connection state* dan menganut aliran *accept some and drop all*, bukan firewall dengan aliran sesat ☺.

```
[admin@R1] > ip firewall filter print
Flags: X - disabled, I - invalid, D - dynamic
0 chain=input action=drop connection-state=invalid
1 chain=input action=accept connection-state=established
2 chain=input action=accept connection-state=related
3 chain=input action=accept protocol=tcp in-interface=ether2 dst-
  port=8291
4 chain=input action=accept protocol=udp in-interface=ether2 dst-
  port=53
5 chain=input action=drop
```

5

Firewall										
Filter Rules NAT Mangle Service Ports Connections Address Lists Layer7 Protocols										
#	Action	Chain	Protocol	Src. Port	Dest. Port	In. Inter.	Out. Int.	Connection State	Bytes	Packets
0	X drop	input						invalid	200 B	5
1	✓ accept	input						established	85.2 KB	788
2	✓ accept	input						related	337 B	3
3	✓ accept	input	6 (tcp)		8291	ether2			52 B	1
4	✓ accept	input	17 (udp)		53	ether2			670 B	11
5	X drop	input							805 B	7

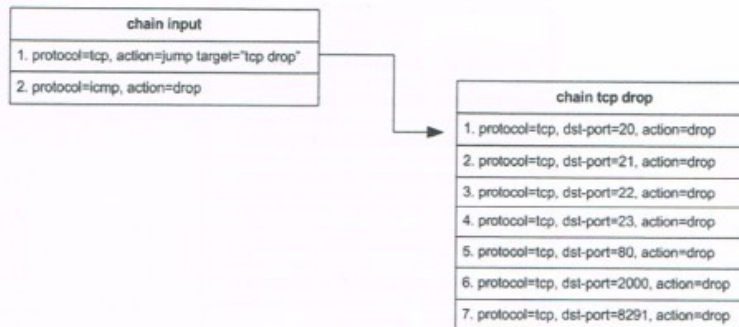
Gambar 5.12

Penggunaan connection state pada chain input

Connection State pada chain forward

Penggunaan connection state pada chain forward dapat Anda lihat pada susunan firewall di bawah ini. Rule-rule firewall tersebut hanya mengizinkan layanan Internet HTTP/HTTPS dan DNS dari jaringan lokal yang terhubung melalui interface *ether2*. Begitu juga dengan penggunaan ping yang diizinkan, jika dilakukan dari dalam jaringan lokal.

```
[admin@R1] > ip firewall filter print
```



Gambar 5.15
Rule firewall dengan chain tambahan

5

Perhatikanlah gambar di atas, Anda dapat melihat bila ada paket ping (ICMP) yang ditujukan bagi interface router, paket tersebut hanya akan melewati 1 (satu) baris. Tidak sama seperti contoh sebelumnya yang harus melewati 7 baris. Sedangkan bila ada paket data yang berisi protokol TCP dan ditujukan pada interface router, maka pemeriksaan paket akan diberikan kepada chain "tcp drop".

Untuk mewujudkan skenario firewall di atas yang menggunakan chain tambahan, marilah kita perhatikan tahapan-tahapan konfigurasi berikut. Yang paling pertama harus dilakukan adalah membuat chain tambahan tersebut dengan menggunakan perintah berikut.

```
[admin@R1] > ip firewall filter add chain="tcp drop" protocol=tcp \
dst-port=20 action=drop

[admin@R1] > ip firewall filter add chain="tcp drop" protocol=tcp \
dst-port=21 action=drop

[admin@R1] > ip firewall filter add chain="tcp drop" protocol=tcp \
dst-port=22 action=drop

[admin@R1] > ip firewall filter add chain="tcp drop" protocol=tcp \
dst-port=23 action=drop

[admin@R1] > ip firewall filter add chain="tcp drop" protocol=tcp \
dst-port=80 action=drop

[admin@R1] > ip firewall filter add chain="tcp drop" protocol=tcp \
dst-port=2000 action=drop
```


chain input
1. protocol=tcp, dst-port=20, action=drop
2. protocol=tcp, dst-port=21, action=drop
3. protocol=tcp, dst-port=22, action=drop
4. protocol=tcp, dst-port=23, action=drop
5. protocol=tcp, dst-port=80, action=drop
6. protocol=tcp, dst-port=2000, action=drop
7. protocol=tcp, dst-port=8291, action=drop
8. protocol=icmp, action=drop

Gambar 5.14
Firewall rule yang bersifat flat

Perhatikanlah dengan seksama gambar di atas, jika ada paket ping (protokol ICMP) yang ditujukan bagi salah satu interface router, maka paket tersebut akan melewati setidaknya 7 (tujuh) baris untuk sampai pada rule nomor (8). Pada saat paket tersebut tiba pada baris (8), barulah paket tersebut di-drop.

Bila ilustrasi pada gambar 5.14 diimplementasikan, maka rule-rule tersebut akan terlihat seperti berikut.

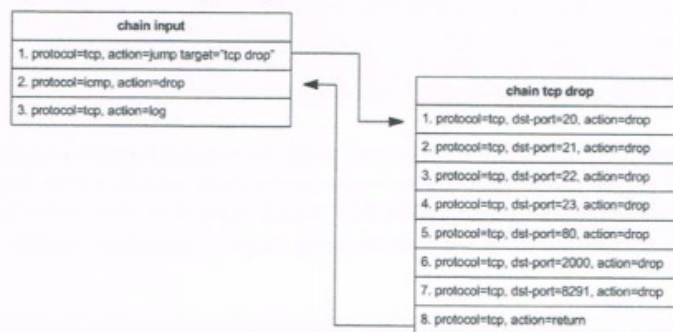
```
[admin@R1] > ip firewall filter print
Flags: X - disabled, I - invalid, D - dynamic
0 chain=input action=drop protocol=tcp dst-port=20
1 chain=input action=drop protocol=tcp dst-port=21
2 chain=input action=drop protocol=tcp dst-port=22
3 chain=input action=drop protocol=tcp dst-port=23
4 chain=input action=drop protocol=tcp dst-port=80
5 chain=input action=drop protocol=tcp dst-port=2000
6 chain=input action=drop protocol=tcp dst-port=8291
7 chain=input action=drop protocol=icmp
```

Untuk menyederhanakan rule firewall tadi maka Anda dapat membuat chain tambahan. Chain tambahan ini dapat berisi parameter *protocol=tcp* dengan *action=drop*. Sebagai contoh chain tambahan tersebut akan diberi nama "*tcp drop*". Dan untuk menghubungkan rule-rule pada *chain input* dengan rule yang ada pada chain "*tcp drop*" tadi, Anda dapat menggunakan parameter *action=jump*.

Untuk lebih memudahkan pemahaman Anda, gambar berikut merupakan ilustrasi rule firewall sebelumnya, namun sudah menggunakan chain tambahan.

action=return

Setelah paket data masuk ke dalam chain tambahan untuk diperiksa, maka Anda dapat mengembalikan paket tersebut ke chain input lagi bila ternyata tidak ada rule yang cocok pada chain "tcp drop" tadi. Untuk mengembalikan lagi pemeriksaan paket tadi, Anda harus membuat satu rule di dalam chain "tcp drop", di mana rule tersebut harus menggunakan parameter *action=return*. Ilustrasinya dapat dilihat pada gambar berikut.



Gambar 5.17
Penggunaan *action=return*

Dari gambar di atas, Anda dapat melihat jika ada paket data dengan protokol TCP dan ternyata *field* destination port-nya tidak ada yang cocok dengan parameter *dst-port* pada chain "tcp drop", maka paket tersebut akan dikembalikan lagi pemeriksaannya pada *chain input*.

Pada chain input, terdapat suatu baris rule firewall dengan *action=log*. Rule tersebut berfungsi hanya untuk melakukan pencatatan (logging) terhadap paket-paket TCP apa saja yang lolos dari chain "tcp drop".

Untuk mewujudkan skenario pada gambar 5.17, Anda dapat menggunakan perintah-perintah berikut ini.

```
[admin@R1] > ip firewall filter add chain="tcp drop" action=return

[admin@R1] > ip firewall filter add chain=input protocol=tcp \
               action=log log-prefix="paket-yg-lolos-dari-tcp-drop"
```

Dengan menggunakan WinBox, maka rule firewall dengan menggunakan *action=return* tadi, dapat dilihat pada gambar seperti berikut ini.

```
[admin@R1] > ip firewall filter add chain="tcp drop" protocol=tcp
dst-port=8291 action=drop

[admin@R1] > ip firewall filter print
Flags: X - disabled, I - invalid, D - dynamic
0 chain=tcp drop action=drop protocol=tcp dst-port=20
1 chain=tcp drop action=drop protocol=tcp dst-port=21
2 chain=tcp drop action=drop protocol=tcp dst-port=22
3 chain=tcp drop action=drop protocol=tcp dst-port=23
4 chain=tcp drop action=drop protocol=tcp dst-port=80
5 chain=tcp drop action=drop protocol=tcp dst-port=2000
6 chain=tcp drop action=drop protocol=tcp dst-port=8291
```

Selanjutnya Anda membuat rule firewall pada *chain input* dengan menggunakan parameter *action=jump*. Yang perlu Anda perhatikan adalah *action jump* yang berfungsi mengalihkan paket-paket TCP untuk diperiksa oleh *chain "tcp drop"* tadi (*jump-target="tcp drop"*). Perintah yang dapat digunakan adalah sebagai berikut.

```
[admin@R1] > ip firewall filter add chain=input protocol=tcp \
    jump-target="tcp drop"

[admin@R1] > ip firewall filter add chain=input protocol=icmp action=drop
```

Aturlah urutan rule firewall Anda, sehingga jika dilihat dengan menggunakan WinBox akan terlihat seperti gambar berikut.

Firewall												
Filter Rules		NAT	Mangle	Service Ports	Connections	Address Lists	Layer7 Protocols					
		Reset Counters		00 Reset All Counters								
#	Action	Chain	Src...	Dest...	Protocol	Src	Dest. Port	In. Inter.	Out. Int.	Bytes	Packets	
0	jump	input			6 (tcp)					0 B	0	
1	drop	input			1 (icmp)					90 B	1	
2	drop	tcp drop			6 (tcp)		20			0 B	0	
3	drop	tcp drop			6 (tcp)		21			0 B	0	
4	drop	tcp drop			6 (tcp)		22			0 B	0	
5	drop	tcp drop			6 (tcp)		23			0 B	0	
6	drop	tcp drop			6 (tcp)		80			0 B	0	
7	drop	tcp drop			6 (tcp)		2000			0 B	0	
8	drop	tcp drop			6 (tcp)		8291			0 B	0	

Gambar 5.16
Penggunaan action jump dilihat dengan WinBox

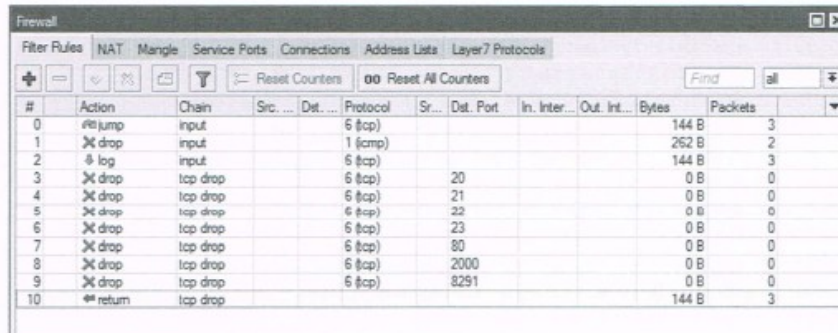

```
[admin@R1] > log print
00:00:14 system,info router rebooted
00:00:22 interface,info ether2 link up (speed 100M, full duplex)
00:00:22 wireless,info 00:0C:42:6A:E7:6E@wlan1: connected
00:00:23 interface,info ether1 link up (speed 100M, full duplex)
00:00:38 system,info,account user admin logged in via winbox
00:00:54 system,info filter rule removed by admin
00:01:02 system,info filter rule removed by admin
00:01:03 system,info filter rule removed by admin
00:01:05 system,info filter rule moved by admin
00:01:12 system,info filter rule changed by admin
00:01:15 system,info filter rule changed by admin
00:14:49 system,info,account user admin logged in via local
00:15:27 system,info filter rule added by admin
00:16:25 system,info filter rule added by admin
00:16:37 system,info filter rule moved by admin
00:16:58 firewall,info paket-yg-lolos-dari-tcp-drop in: in:ether2
out:(none), src-mac 00:1e:68:55:a9:d8, proto TCP (SYN),
192.168.10.20:8208->192.168.10.1:24, len 48
00:16:58 firewall,info paket-yg-lolos-dari-tcp-drop in: in:ether2
out:(none), src-mac 00:1e:68:55:a9:d8, proto TCP (SYN),
192.168.10.20:8208->192.168.10.1:24, len 48
00:16:59 firewall,info paket-yg-lolos-dari-tcp-drop in: in:ether2
out:(none), src-mac 00:1e:68:55:a9:d8, proto TCP (SYN),
192.168.10.20:8208->192.168.10.1:24, len 48
00:17:17 system,info filter rule moved by admin
00:20:09 system,info filter rule moved by admin
00:26:20 system,info,account user admin logged in via local
```

chain=input, chain=forward dan chain tambahan

Penggunaan chain tambahan juga akan menghindarkan Anda dari pekerjaan menulis konfigurasi firewall berulang-ulang. Misalnya, pada saat Anda ingin menerapkan pembatasan (blok) *traffic* TCP dengan parameter protokol dan destination port yang sama, namun untuk *chain input* dan *chain forward*. Jika tidak menggunakan chain tambahan, maka Anda akan menuliskan rule firewall tersebut berulang kali, seperti pada ilustrasi berikut ini.

chain input	chain forward
1. protocol=tcp, dst-port=20, action=drop	1. protocol=tcp, dst-port=20, action=drop
2. protocol=tcp, dst-port=21, action=drop	2. protocol=tcp, dst-port=21, action=drop
3. protocol=tcp, dst-port=22, action=drop	3. protocol=tcp, dst-port=22, action=drop
4. protocol=tcp, dst-port=23, action=drop	4. protocol=tcp, dst-port=23, action=drop
5. protocol=tcp, dst-port=80, action=drop	5. protocol=tcp, dst-port=80, action=drop
6. protocol=tcp, dst-port=2000, action=drop	6. protocol=tcp, dst-port=2000, action=drop
7. protocol=tcp, dst-port=8291, action=drop	7. protocol=tcp, dst-port=8291, action=drop
8. protocol=icmp, action=drop	8. protocol=icmp, action=drop

Gambar 5.20
Rule Firewall tanpa chain tambahan

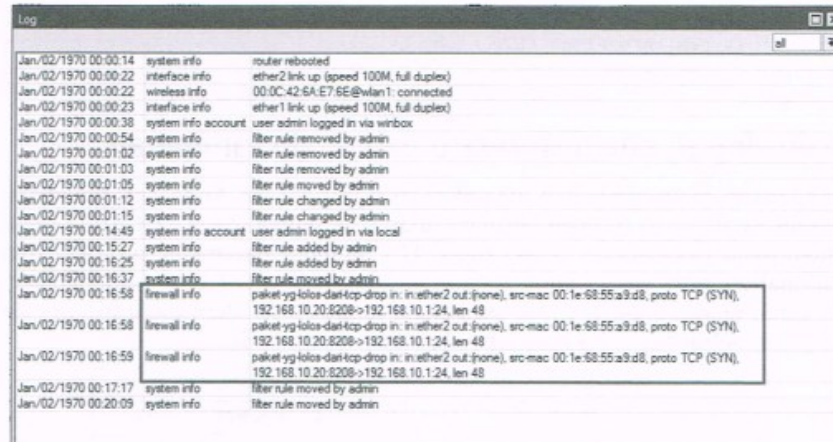


#	Action	Chain	Src...	Dest...	Protocol	Sr...	Dest. Port	In. Inter...	Out. Int...	Bytes	Packets
0	jump	input			6 (tcp)					144 B	3
1	drop	input			1 (icmp)					262 B	2
2	log	input			6 (tcp)					144 B	3
3	drop	tcp drop			6 (tcp)		20			0 B	0
4	drop	tcp drop			6 (tcp)		21			0 B	0
5	drop	tcp drop			6 (tcp)		22			0 B	0
6	drop	tcp drop			6 (tcp)		23			0 B	0
7	drop	tcp drop			6 (tcp)		80			0 B	0
8	drop	tcp drop			6 (tcp)		2000			0 B	0
9	drop	tcp drop			6 (tcp)		8291			0 B	0
10	return	tcp drop								144 B	3

Gambar 5.18
Penggunaan action=return dilihat dengan WinBox

Action *log* pada konfigurasi tersebut akan melakukan pencatatan bila ada paket data dengan *field* destination port selain yang tertera pada chain "tcp drop". Untuk melihat hasil kinerja dari rule firewall dengan *action=log*, Anda dapat menggunakan menu *Log* pada WinBox. Gambar berikut memperlihatkan pencatatan pada saat ada komputer client yang mengakses port 24 dari Router MikroTik Anda.

5



Time	System Info	Firewall Info
Jan/02/1970 00:00:14	system info	router rebooted
Jan/02/1970 00:00:22	interface info	ether2 link up (speed 100M, full duplex)
Jan/02/1970 00:00:22	wireless info	00:0C:42:6A:E7:6E@wan1: connected
Jan/02/1970 00:00:23	interface info	ether1 link up (speed 100M, full duplex)
Jan/02/1970 00:00:38	system info account	user admin logged in via winbox
Jan/02/1970 00:00:54	system info	filter rule removed by admin
Jan/02/1970 00:01:02	system info	filter rule removed by admin
Jan/02/1970 00:01:03	system info	filter rule removed by admin
Jan/02/1970 00:01:05	system info	filter rule moved by admin
Jan/02/1970 00:01:12	system info	filter rule changed by admin
Jan/02/1970 00:01:15	system info	filter rule changed by admin
Jan/02/1970 00:14:49	system info account	user admin logged in via local
Jan/02/1970 00:15:27	system info	filter rule added by admin
Jan/02/1970 00:16:25	system info	filter rule added by admin
Jan/02/1970 00:16:37	system info	filter rule moved by admin
Jan/02/1970 00:16:58	firewall info	packet-yg-kolot-dari-tcp-drop in: in:ether2 out:(none), src-mac 00:1e:68:55:a3:d8, proto TCP (SYN), 192.168.10.20:82208->192.168.10.1:24, len 48
Jan/02/1970 00:16:58	firewall info	packet-yg-kolot-dari-tcp-drop in: in:ether2 out:(none), src-mac 00:1e:68:55:a3:d8, proto TCP (SYN), 192.168.10.20:82208->192.168.10.1:24, len 48
Jan/02/1970 00:16:59	firewall info	packet-yg-kolot-dari-tcp-drop in: in:ether2 out:(none), src-mac 00:1e:68:55:a3:d8, proto TCP (SYN), 192.168.10.20:82208->192.168.10.1:24, len 48
Jan/02/1970 00:17:17	system info	filter rule moved by admin
Jan/02/1970 00:20:09	system info	filter rule moved by admin

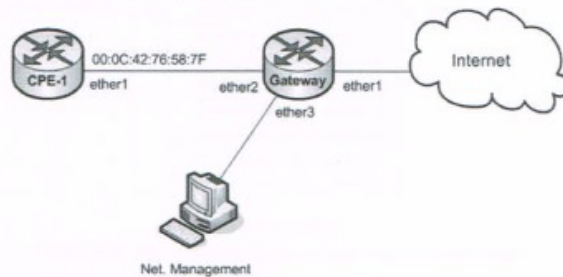
Gambar 5.19
Hasil dari rule firewall dengan action=log

Anda juga dapat menggunakan perintah berikut untuk melihat log dari Router MikroTik Anda.

MAC Server

Router MikroTik yang belum memiliki IP Address pun dapat dikonfigurasi baik dengan menggunakan WinBox maupun MAC Telnet. Anda cukup menggunakan MAC Address dari interface router untuk login dan kemudian melakukan konfigurasi. Namun untuk melakukan konfigurasi melalui MAC Address, Anda harus terhubung langsung pada interface router, minimal laptop yang akan digunakan berada satu network dengan router tersebut.

Anda juga dapat melakukan konfigurasi suatu router melalui router lain dengan menggunakan tool MAC Telnet. Perhatikanlah gambar berikut yang memperlihatkan router CPE yang belum memiliki IP Address.



Gambar 5.23

Router CPE-1 tidak memiliki IP Address

Bila Anda sudah berhasil login ke Router Gateway, maka pekerjaan konfigurasi pada Router CPE-1 dapat dilakukan walaupun router tersebut belum memiliki IP Address. Yang perlu dilakukan adalah menjalankan MAC Telnet pada Router Gateway dengan menggunakan perintah seperti berikut ini.

```
[admin@Gateway] > tool mac-telnet 00:0C:42:76:58:7F
```

Bagaimana jika ternyata Anda tidak mengetahui MAC Address interface *ether1* dari router CPE-1? Solusi yang bisa diambil adalah menjalankan MAC Scan untuk mengetahuinya, seperti perintah berikut ini.

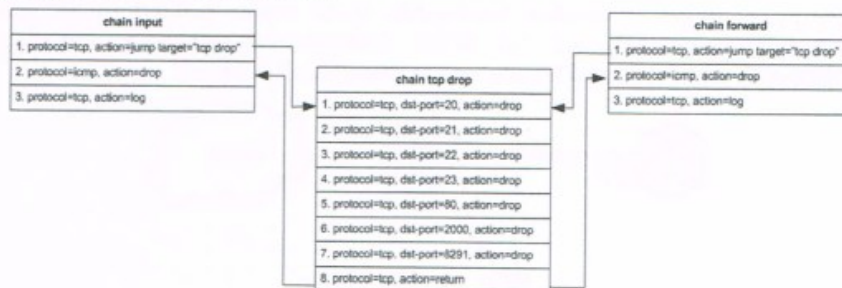
```
[admin@Gateway] > tool mac-scan interface=ether2
```

MAC-ADDRESS	ADDRESS	AGE
00:0C:42:76:58:7F		5
00:0C:42:5D:0F:A4	255.255.255.255	5

```
-- [Q quit|D dump|C-z pause]
```


Jika Anda ingin mengimplementasikan skenario firewall pada gambar di atas, maka setidaknya Anda akan membuat 16 (enam belas) rule firewall, masing-masing 8 (delapan) untuk *chain input* dan 8 (delapan) lagi untuk *chain forward*. Belum lagi jika Anda masih ingin menambahkan rule dengan *action=log* pada masing-masing chain, maka total rule firewall akan menjadi 18 (delapan belas) baris.

Dengan membuat chain tambahan, maka jumlah baris rule firewall dapat dikurangi. Anda dapat melihat gambar berikut yang memperlihatkan penggunaan chain tambahan.



Gambar 5.21
Chain forward dan input menggunakan chain "tcp drop"

Jika ilustrasi pada gambar 5.21, diwujudkan dalam bentuk konfigurasi maka hasil akan terlihat seperti berikut ini.

Firewall												
Filter Rules												
#	Action	Chain	Src.	Dest.	Protocol	Sr.	Dest. Port	In. Inter.	Out. Int.	Bytes	Packets	
0	jump	input			tcp					144 B	3	
1	drop	input			icmp					803 B	6	
2	log	input			tcp					144 B	3	
3	jump	forward			tcp					5.8 KiB	99	
4	drop	forward			icmp					0 B	0	
5	log	forward			tcp					0 B	0	
6	drop	tcp drop			tcp		20			0 B	0	
7	drop	tcp drop			tcp		21			0 B	0	
8	drop	tcp drop			tcp		22			0 B	0	
9	drop	tcp drop			tcp		23			96 B	2	
10	drop	tcp drop			tcp		80			4360 B	92	
11	drop	tcp drop			tcp		2000			0 B	0	
12	drop	tcp drop			tcp		8291			0 B	0	
13	return	tcp drop			tcp					1580 B	8	

Gambar 5.22
Chain forward dan input menggunakan chain "tcp drop"

seperti berikut ini:

```
[admin@CPE-1] > tool mac-server mac-winbox print
Flags: X - disabled, * - default
#   INTERFACE
0  * all
```

Contoh berikut memperlihatkan *session* login ke dalam router yang sedang aktif.

```
[admin@CPE-1] > tool mac-server sessions print
# INTERFACE SRC-ADDRESS UPTIME
0 ether1 00:0C:42:5D:0F:A4 26s
```

Bila Anda ingin hanya ingin meng-aktifkan konfigurasi WinBox melalui MAC Address pada interface *ether1* saja, maka perintah yang dapat digunakan adalah sebagai berikut.

```
[admin@CPE-1] > tool mac-server mac-winbox add interface=ether1 \
                 disable=no

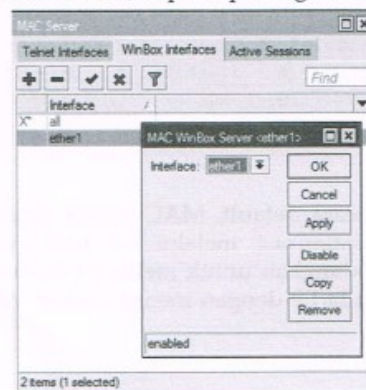
[admin@CPE-1] > tool mac-server mac-winbox disable 0

[admin@CPE-1] > tool mac-server mac-winbox print
Flags: X - disabled, * - default
#   INTERFACE
0 X* all
1   ether1
```

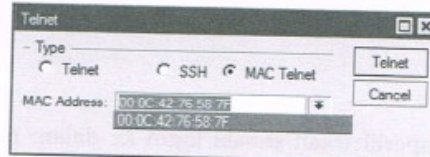
Dengan menggunakan WinBox, Anda dapat menggunakan menu *Tools* → *MAC Server* → tab *WinBox Interfaces* → tombol *Add*, seperti pada gambar berikut ini.

Gambar 5.25
MAC Server untuk login WinBox

Bila Anda ingin hanya ingin meng-aktifkan konfigurasi MAC Telnet melalui MAC Address pada interface *ether1* saja, maka perintah yang dapat digunakan adalah



Bila ingin menjalankan MAC Telnet dengan menggunakan WinBox, maka Anda dapat menggunakan menu *Tools* → *Telnet* → pilih *MAC Telnet* → tombol *Telnet*, seperti pada gambar berikut ini.



Gambar 5.24
MAC Telnet

Mungkin Anda bertanya-tanya, mengapa saya menuliskan sub bab MAC Server seperti ini? Pada sub bab sebelumnya Anda sudah melihat bagaimana konfigurasi-konfigurasi yang bertujuan membatasi akses login ke router MikroTik, tentu dengan menggunakan *chain input*. Dan Anda juga melihat bahwa pembatasan tersebut dilakukan dengan parameter protokol maupun destination port. Jika dihubungkan dengan OSI Model, apa yang dilakukan pada konfigurasi-konfigurasi tersebut adalah pembatasan pada Layer 3 (Network).

Yang menjadi dilema adalah Router MikroTik secara default menjalankan MAC Server yang memungkinkan login ke dalam router hanya dengan menggunakan MAC Address. Jika dihubungkan dengan OSI Model lagi, maka login dengan menggunakan MAC Address ini terjadi pada Layer 2 (Data Link). Sehingga untuk benar-benar mengamankan router Anda, pembatasan harus dilakukan pada kedua layer (baik L3 dan L2). Dan untuk membatasi akses yang terjadi pada Layer 2, Anda harus melakukan konfigurasi pada MAC Server. Tidak cukup hanya dengan mengandalkan fitur firewall pada sub bab sebelumnya.

Anda dapat melihat parameter-parameter MAC Server pada Router MikroTik dengan menggunakan perintah seperti berikut ini.

```
[admin@Gateway] > tool mac-server print
Flags: X - disabled, * - default
#   INTERFACE
0   * all
```

Secara default, MAC Server akan aktif di semua interface router, sehingga konfigurasi melalui L2 tadi, dapat dilakukan dari interface manapun. Sedangkan untuk melihat interface mana saja yang dapat menerima koneksi pada L2 dengan menggunakan WinBox, Anda dapat menggunakan perintah



Jangan Hanya Membayangkan, Wujudkan
Pengetahuan Anda menjadi tulisan.

Kirimkan Naskah Anda ke :

Admin@Jasakom.com

* Info lebih lanjut : www.jasakom.com

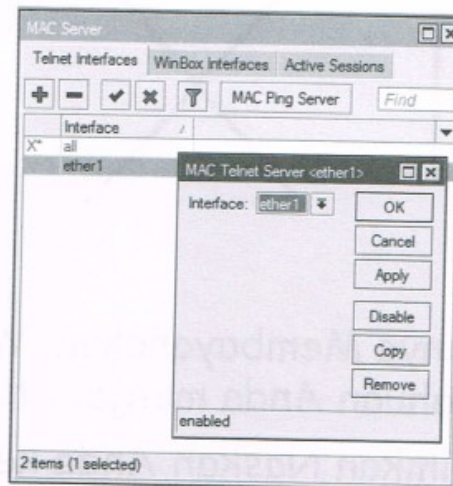
sebagai berikut:

```
[admin@CPE-1] > tool mac-server add interface=ether1 disabled=no

[admin@CPE-1] > tool mac-server disable 0

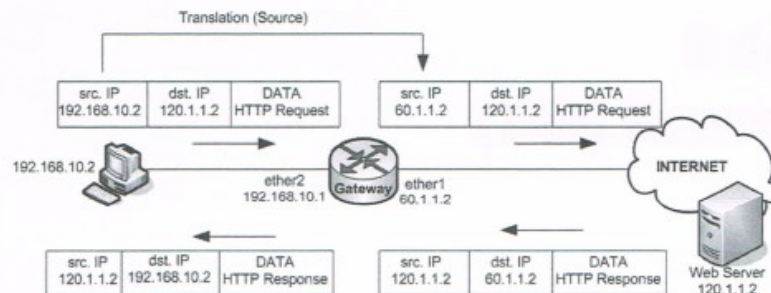
[admin@CPE-1] > tool mac-server print
Flags: X - disabled, * - default
#      INTERFACE
0 X*   all
1      ether1
```

Dengan menggunakan WinBox, konfigurasi dapat dilakukan melalui menu *Tools* → *MAC Server* → tab *Telnet Interfaces* → tombol *Add*, seperti pada gambar berikut ini.



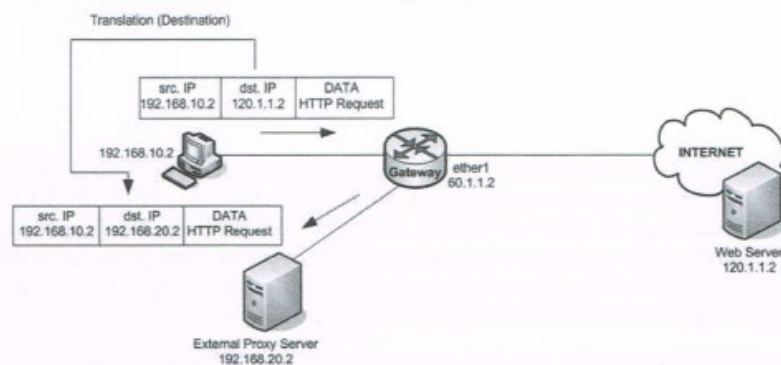
Gambar 5.26
MAC Server untuk login MAC Telnet

---END OF CHAPTER---



Gambar 6.1
Chain srcnat pada sebuah router

Sedangkan pada gambar 6.2, Anda dapat melihat bagaimana *chain dstnat* melakukan perubahan *field* IP Address pada paket.



Gambar 6.2
Chain dstnat pada sebuah router

Karena umumnya digunakan pada router yang menghubungkan beberapa jaringan, maka penggunaan NAT erat kaitannya dengan *chain forward* pada firewall filter.. Anda dapat melihat gambar berikut untuk melihat kaitan antara *chain srcnat* maupun *chain dstnat* dengan *chain forward* pada firewall filter.

Saya tidak akan menggambarkan diagram lengkap yang lebih kompleks tentang Packet Flow pada sebuah router. Saya menginginkan Anda dapat memahami konsep firewall secara sederhana terlebih dahulu untuk kemudian ke tahap yang lebih rumit dan kompleks. Itulah sebabnya pada gambar tersebut saya hanya memperlihatkan tiga buah chain. Namun jika Anda ingin sedikit berpusing ria, silahkan menuju URL <http://wiki.mikrotik.com/wiki/>

Bab 6

Firewall (NAT)

6

Network Address Translation (NAT) adalah salah satu fitur pada firewall yang digunakan untuk melakukan perubahan *field* IP Address, baik IP Address pengirim maupun IP Address Tujuan. Selain *field* IP Address, NAT juga dapat melakukan perubahan *field* port pengirim maupun port tujuan pada paket data. NAT lebih banyak diimplementasikan pada router-router yang ditempatkan di antara jaringan lokal (*private*) dengan jaringan publik (*Internet*).

Firewall NAT juga terdiri dari susunan rule-rule, seperti yang pernah ada pada firewall filter di Bab 5 sebelumnya. Rule-rule pada Firewall NAT ini juga akan dibaca dari atas ke bawah (*top to bottom*). Jika pada bab sebelumnya Anda juga sudah melihat penggunaan chain tambahan dengan memanfaatkan *action jump* dan *action return*, maka penggunaan chain tambahan tersebut dapat juga Anda lakukan pada Firewall NAT ini.

Terdapat dua chain pada firewall NAT, yaitu :

- **Chain source NAT (*srcnat*)**, digunakan untuk merubah *field* IP Address pengirim maupun port pengirim yang ada pada paket.
- **Chain destination NAT (*dstnat*)**, digunakan untuk merubah *field* IP Address tujuan maupun port tujuan yang ada pada paket.

Anda dapat memperhatikan gambar 6.1 untuk melihat bagaimana *chain srcnat* melakukan perubahan *field* IP Address.

kondisi di mana router MikroTik memiliki koneksi ke Internet melalui beberapa ISP. Dengan hadirnya beberapa koneksi Internet, maka Anda dapat menerapkan teknik *load balancing*.

Teknik ini akan membagi beban (*traffic*) dari jaringan lokal ke beberapa koneksi ISP yang dimiliki. Ini dapat mencegah menumpuknya *traffic* kepada satu koneksi ISP, sehingga semua koneksi Internet yang ada dapat digunakan secara seimbang (*balance*). Teknik *load balancing* yang digunakan adalah metode *Equal Cost Multi Path (ECMP)*, yang merupakan teknik yang sederhana, namun menjadi dasar bagi Anda untuk membuat teknik yang lebih rumit, tentunya dengan parameter konfigurasi yang lebih banyak.

Sedangkan *action* yang digunakan juga tidak hanya *action masquerade* namun akan dibahas penggunaan *action srcnat*. Selain kedua *action* tersebut (*masquerade* dan *srcnat*), Anda dapat saja menggunakan *action-action* yang lain, yang sifatnya sama dengan *action* yang ada pada firewall filter sebelumnya.

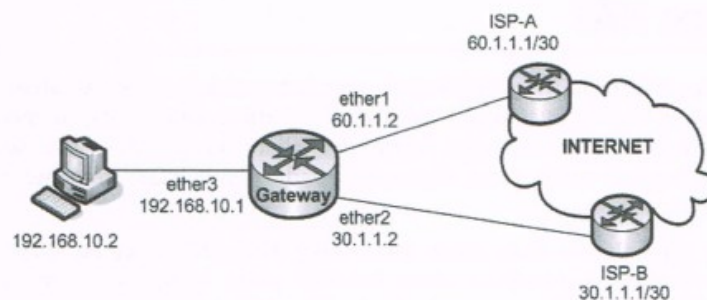
Misalnya, jika Anda ingin melakukan logging terhadap beberapa paket yang akan diproses oleh *chain srcnat*, maka Anda bisa saja memilih *action=log*. Begitu juga dengan *action-action* yang lain seperti penggunaan *action accept*, *add-src-to-src-address-list*, *add-dst-to-dst-address-list* maupun *action* yang lainnya.

6

action=masquerade

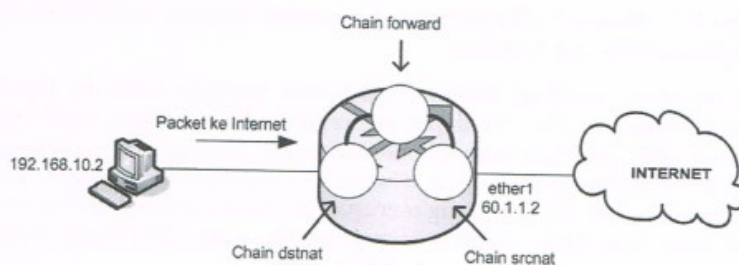
Skenario 1

Pada skenario ini, Router Gateway memiliki koneksi Internet dari dua ISP yang berbeda. Koneksi Internet tersebut terdiri dari upload maupun download sebesar 512 kbps dari ISP-A dan 512 kbps dari ISP-B. ISP-A terhubung pada interface *ether1* Router Gateway sedangkan ISP-B terhubung ke *ether2*. Yang diinginkan adalah memanfaatkan kedua koneksi Internet seperti terlihat pada gambar berikut ini.



Gambar 6.4
Koneksi Internet melalui dua ISP

Manual:Packet_Flow.



Gambar 6.3
Hubungan antara chain srcnat, dstnat dan forward

Pada gambar di atas, Anda dapat melihat posisi dari setiap chain pada sebuah router. Terlihat bahwa jika ada sebuah paket data yang ingin melintasi router, maka chain pertama yang dapat digunakan adalah *chain srcnat*. Setelah paket data melewati *chain dstnat*, maka paket akan diberikan ke *chain forward*. *Chain forward* ini dapat memutuskan apakah paket dapat diteruskan (*accept*) atau paket harus dibuang (*drop*).

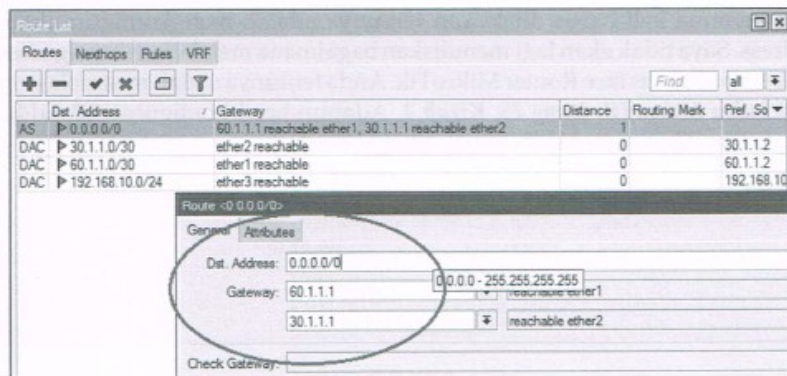
Jika pada *chain dstnat* sebelumnya sudah dilakukan perubahan IP Address tujuan, maka *field* IP Address tujuan yang baru inilah yang akan digunakan *chain forward* sebagai pedoman untuk meneruskan paket. Setelah melewati *chain dstnat* dan *chain forward*, maka selanjutnya paket akan diberikan kepada *chain srcnat* yang dapat melakukan perubahan *field* IP Address pengirim maupun port pengirim.

Dengan memahami hubungan ketiga chain tersebut, Anda dapat membuat susunan rule firewall sesuai kebutuhan, lebih efisien, lebih efektif dan tentunya dengan "taste" Anda sendiri.

Source NAT

Pada buku *MikroTik Kung Fu Kitab 1*, penggunaan *chain srcnat* sudah dibahas. Penggunaan *chain srcnat* pada buku tersebut ditujukan untuk memberikan akses Internet pada beberapa komputer client yang ada pada jaringan lokal. Sedangkan *action* yang digunakan pada pembahasan tersebut adalah *masquerade*.

Adapun pembahasan *chain srcnat* pada buku *MikroTik Kung Fu Kitab 2* ini, tetap bertujuan memberikan akses Internet pada komputer client namun dengan beberapa skenario yang berbeda. Skenario yang akan dibahas adalah



Gambar 6.5
Dua gateway untuk menuju Internet

Dengan memiliki dua koneksi Internet, sebaiknya Router Gateway juga menggunakan konfigurasi DNS Server dari kedua ISP tersebut. Pada skenario, ISP-A menjalankan DNS Servernya pada IP Address 60.1.1.1 sedangkan ISP-B pada 30.1.1.1. Sehingga perintah konfigurasi DNS Server yang dapat dilakukan pada Router Gateway adalah sebagai berikut.

6

```
[admin@Gateway] > ip dns set servers=60.1.1.1,30.1.1.1 \
allow-remote-request=yes

[admin@Gateway] > ip dns print
servers: 60.1.1.1,30.1.1.1
dynamic-servers:
allow-remote-requests: yes
max-udp-packet-size: 4096
cache-size: 2048KiB
cache-max-ttl: 1w
cache-used: 47KiB
```

Tahap terakhir adalah mengkonfigurasi Firewall NAT yang akan melakukan perubahan *field* IP Address pengirim lokal menjadi IP Address publik yang ada pada interface *ether1* maupun *ether2*. Karena memiliki dua gateway dan dua IP Address publik di masing-masing interface tersebut, maka harus dilakukan dua konfigurasi Firewall NAT, masing-masing menggunakan parameter *out-interface=ether1* dan *out-interface=ether2*. Perintah yang dapat digunakan adalah sebagai berikut.

Yang pertama kali harus dilakukan tentunya adalah mengkonfigurasi IP Address. Saya tidak akan lagi menuliskan bagaimana melakukan konfigurasi IP Address pada interface Router MikroTik. Anda tentunya sudah mempelajarinya pada buku *MikroTik Kung Fu Kitab 1*. Adapun hasil konfigurasi IP Address yang harus dilakukan dapat dilihat sebagai berikut.

```
[admin@Gateway] > ip address print
Flags: X - disabled, I - invalid, D - dynamic
# ADDRESS NETWORK INTERFACE
0 60.1.1.2/30 60.1.1.0 ether1
1 30.1.1.2/30 30.1.1.0 ether2
2 192.168.10.1/24 192.168.10.0 ether3
```

Karena memiliki dua ISP, maka tentunya Router Gateway akan memiliki dua gateway. Dan karena kedua ISP tersebut akan digunakan bersama-sama maka pada saat mengkonfigurasi default gateway menuju Internet, Anda harus memasukkan dua IP Address dari router ISP secara bersamaan. Perintah yang dapat digunakan adalah sebagai berikut.

```
[admin@Gateway] > ip route add dst-address=0.0.0.0/0 \
gateway=60.1.1.1,30.1.1.1

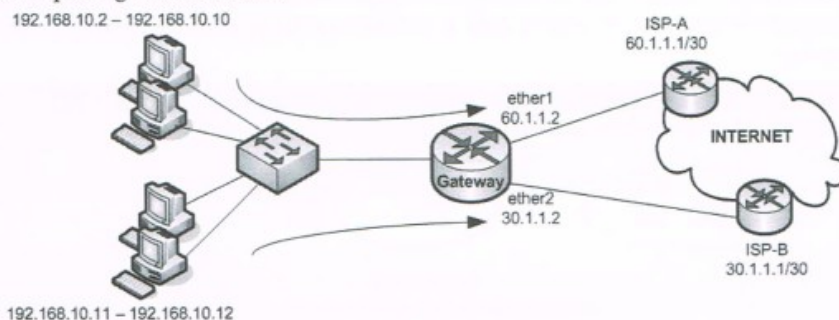
[admin@Gateway] > ip route print
Flags: X - disabled, A - active, D - dynamic,
C - connect, S - static, r - rip, b - bgp, o - ospf, m - mme,
B - blackhole, U - unreachable, P - prohibit
# DST-ADDRESS PREF-SRC GATEWAY DISTANCE
0 A S 0.0.0.0/0 30.1.1.1 1
60.1.1.1
1 ADC 30.1.1.0/30 30.1.1.2 ether2 0
2 ADC 60.1.1.0/30 60.1.1.2 ether1 0
3 ADC 192.168.10.0/24 192.168.10.1 ether3 0
```

6

Jika menggunakan WinBox, menu yang dapat digunakan adalah *IP → Routes*, seperti terlihat pada gambar berikut.

Skenario 2

Jika Anda ingin memisahkan *traffic* dari beberapa komputer ke ISP tertentu maka Anda dapat memanfaatkan parameter *src-address*. Pada skenario ini, diinginkan komputer client dengan IP Address 192.168.10.2 s/d 192.168.10.10 akan menggunakan koneksi Internet melalui ISP-A. Sedangkan komputer client dengan IP Address 192.168.10.11 s/d 192.168.10.20 akan menggunakan koneksi Internet melalui ISP-B. Sehingga akan dihasilkan dua group dalam jaringan lokal Anda, dan masing-masing group akan mendapatkan jatah bandwidth download dan upload sebesar 512 kbps. Ilustrasi dari skenario ini dapat Anda lihat pada gambar berikut.



Gambar 6.8

Load Balance dengan menggunakan parameter *src-address*

Untuk menjalankan skenario ini, harus dilakukan dua konfigurasi rule Firewall NAT dengan memanfaatkan parameter *src-address*. Rule yang pertama menggunakan parameter *src-address=192.168.10.2-192.168.10.10 out-interface=ether1* sedangkan rule yang kedua menggunakan parameter *src-address=192.168.10.11-192.168.10.20 out-interface=ether2*. Secara lengkap perintah yang dapat digunakan adalah sebagai berikut.

```
[admin@Gateway] > ip firewall nat add chain=srcnat \
src-address=192.168.10.2-192.168.10.10 out-interface=ether1 \
action=masquerade

[admin@Gateway] > ip firewall nat add chain=srcnat \
src-address=192.168.10.11-192.168.10.20 out-interface=ether2 \
action=masquerade

[admin@Gateway] > ip firewall nat print
Flags: X - disabled, I - invalid, D - dynamic
```



```
[admin@Gateway] > ip firewall nat add chain=srcnat
out-interface=ether1 action=masquerade

[admin@Gateway] > ip firewall nat add chain=srcnat
out-interface=ether2 action=masquerade

[admin@Gateway] > ip firewall nat print
Flags: X - disabled, I - invalid, D - dynamic
0 chain=srcnat action=masquerade out-interface=ether1
1 chain=srcnat action=masquerade out-interface=ether2
```

Jika menggunakan WinBox, konfigurasi dapat dilakukan melalui menu **IP** → **Firewall** → tab **NAT** → tombol **Add**, dan hasilnya dapat dilihat seperti gambar berikut.

#	Action	Chain	Proto	In. Interface	Out. Interface	Bytes	Packets
0	masquerade	srcnat			ether1	1080 B	12
1	masquerade	srcnat			ether2	1455 B	17

Gambar 6.6
Masquerade dengan dua out-interface

6

Hasil dari konfigurasi *load balancing* di atas akan bersifat *round robin*, artinya kedua *link* akan digunakan secara acak. Bila mana koneksi ISP-A sudah terbebani secara penuh, maka *traffic* selanjutnya baru akan dialihkan ke ISP-B.

Jika beban sudah terbagi bagi pada kedua ISP, maka kondisi seimbang (*balance*) akan tercapai, Anda dapat melihat berapa bandwidth yang digunakan melalui menu **Interfaces**, seperti pada gambar berikut. Perhatikanlah bagian RX pada interface *ether1* dan *ether2* yang memperlihatkan besarnya *traffic* download yang didapat dari kedua ISP.

Name	Type	L2 MTU	Tx	Rx	Tx Pac...	Rx Pac...	Tx Drops	Rx Drops	Tx Errors	Rx Errors
R ether1	Ethernet	1526	13.8 kbps	514.6 kbps	28	45	0	0	0	0
R ether2	Ethernet	1522	14.0 kbps	510.0 kbps	25	42	0	0	0	0
R ether3	Ethernet	1522	1099.5 kbps	29.8 kbps	93	55	0	0	0	0
ether4	Ethernet	1522	0 bps	0 bps	0	0	0	0	0	0
ether5	Ethernet	1522	0 bps	0 bps	0	0	0	0	0	0

Gambar 6.7
Beban yang seimbang (*load balance*) pada *ether1* dan *ether2*

action=srcnat

Selain menggunakan *action masquerade* untuk menyembunyikan IP Address privat sehingga tidak terlihat oleh server-server di Internet, Anda juga dapat menggunakan *action src-nat*. Jika *action masquerade* dapat membuat komputer-komputer client dapat mengakses Internet, begitu juga dengan *action src-nat*. Yang menjadi pertanyaan adalah kapan Anda menggunakan *action masquerade* dan kapan Anda harus menggunakan *action src-nat*.

Action masquerade digunakan untuk melakukan perubahan secara otomatis *field* IP Address yang ada pada paket data menjadi IP Address publik yang ada pada interface publik.

Bila Anda mengambil contoh jaringan pada gambar 6.4, maka jika ada paket yang keluar ke Internet melalui interface *ether1*, maka *field* IP Address pengirim yang ada pada paket tersebut akan dirubah menjadi IP Address 60.1.1.2. Sedangkan jika ada paket data yang keluar ke Internet melalui interface *ether2*, maka *field* IP Address pengirim pada paket tersebut akan dirubah menjadi 30.1.1.2.

Action src-nat juga akan melakukan hal yang sama seperti *action masquerade*. Namun Anda memiliki kebebasan menentukan IP Address publik mana yang akan digunakan oleh paket data pada saat akan keluar menuju Internet. Action ini sering digunakan bila mana pada satu interface publik terdapat beberapa IP Address.

Misalnya, Anda berlangganan Internet dari beberapa ISP, sedangkan router yang Anda miliki hanya memiliki sebuah interface publik. Marilah kita lihat dua contoh skenario yang menerapkan *action src-nat* berikut.

Skenario 1

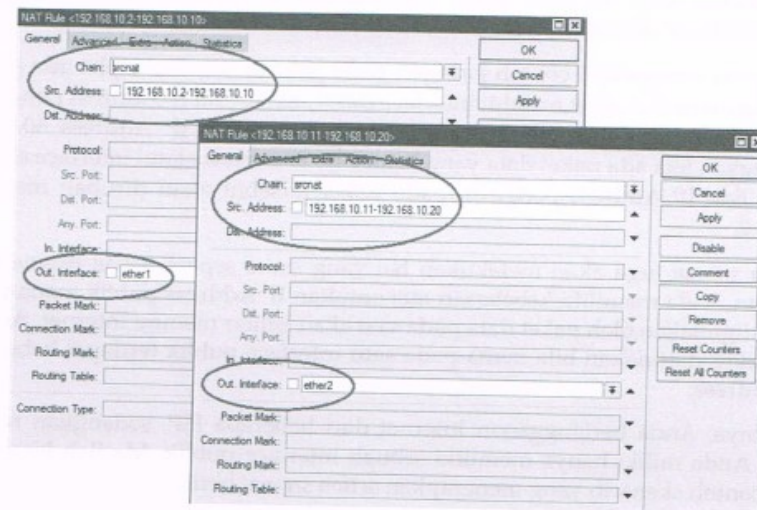
Skenario ini sebenarnya sama dengan skenario yang ada pada gambar 6.8. Router Gateway terhubung ke ISP-A melalui *ether1* dan terhubung ke ISP-B melalui *ether2*. Yang diinginkan juga adalah komputer client dengan IP Address 192.168.10.2 s/d 192.168.10.10 akan menggunakan koneksi Internet dari ISP-A, sedangkan komputer client dengan IP Address 192.168.10.11 s/d 192.168.10.20 akan menggunakan koneksi Internet dari ISP-B.

Sama seperti pada skenario sebelumnya, harus dikonfigurasi IP Address pada semua interface, baik *ether1*, *ether2* maupun *ether3*. Kemudian harus juga dikonfigurasi DNS Server yang didapat dari kedua ISP. Adapun untuk konfigurasi default gateway, Anda harus mengkonfigurasi dua gateway (60.1.1.1 dan 30.1.1.1) untuk menuju Internet, sama seperti konfigurasi default gateway pada skenario-skenario *action masquerade* sebelumnya.

```
0 chain=srcnat action=masquerade
  src-address=192.168.10.2-192.168.10.10 out-interface=ether1

1 chain=srcnat action=masquerade
  src-address=192.168.10.11-192.168.10.20
  out-interface=ether2
```

Jika menggunakan WinBox, maka konfigurasi dapat dilakukan melalui menu **IP → Firewall → tab NAT → tombol Add**, sama seperti konfigurasi sebelumnya, hanya saja kali ini Anda harus mengisi parameter *Src. Address*, seperti gambar berikut.



Gambar 6.9
Penggunaan *Src. Address* melalui WinBox

Sedangkan hasil akhir konfigurasi jika dilihat dengan menggunakan WinBox, akan terlihat seperti gambar berikut.

#	Action	Chain	Src. Address	Dst. Address	In. Interface	Out. Int.	Bytes	Packets
0	=> masquerade	srcnat	192.168.10.2-192.168.10.10			ether1	7.2 KiB	154
1	=> masquerade	srcnat	192.168.10.11-192.168.10.20			ether2	576 B	12

Gambar 6.10
Hasil akhir konfigurasi load balance dengan parameter *Src. Address*

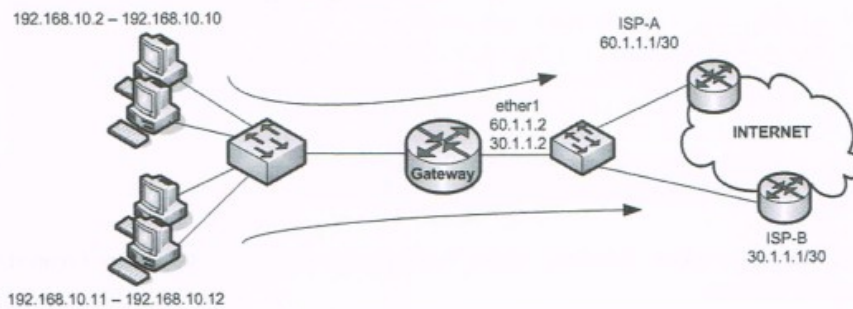
Quick Set	Filter Rules	NAT	Mangle	Service Ports	Connections	Address Lists	Layer7 Protocols
Interfaces							
Wireless							
Bridge							
PPP							
Switch							
Mesh							

#	Action	Chain	Src. Address	Dest. Address	Out. Interface	To Addresses	Bytes	Packets
0	src-nat	srcnat	192.168.10.2-192.168.10.10			60.1.1.2	47.9 KB	1 021
1	src-nat	srcnat	192.168.10.11-192.168.10.20			30.1.1.2	25.5 KB	545

Gambar 6.12
Hasil Konfigurasi dengan menggunakan action src-nat

Skenario 2

Bagaimana jika ternyata Anda berlangganan Internet ke beberapa ISP, namun Router yang ada hanya memiliki sebuah interface publik? Dengan bantuan sebuah Switch Anda tetap akan dapat memaksimalkan kedua koneksi Internet tersebut seperti terlihat pada gambar berikut.



Gambar 6.13
Satu interfae untuk dua koneksi Internet

Pada skenario di atas, tetap diinginkan komputer client dengan IP Address 192.168.10.2 s/d 192.168.10.10 akan menggunakan koneksi Internet dari ISP-A dan komputer client dengan IP Address 192.168.10.11 s/d 192.168.10.20 akan menggunakan koneksi dari ISP-B.

Untuk melakukan konfigurasi IP Address pada interface publik (dalam skenario ini adalah *ether1*), perintah yang dapat Anda gunakan adalah sebagai berikut.

```
[admin@Gateway] > ip address add address=60.1.1.2/30 interface=ether1
[admin@Gateway] > ip address add address=30.1.1.2/30 interface=ether1
```

Yang berbeda adalah konfigurasi rule firewall NAT untuk merubah IP Address pengirim pada paket menjadi IP Address publik dengan menggunakan *action src-nat*. Konfigurasi pertama yang harus Anda lakukan adalah melakukan perubahan IP Address pengirim 192.168.10.2 s/d 192.168.10.20 menjadi IP Address 60.1.1.2. Sedangkan konfigurasi kedua adalah melakukan perubahan IP Address pengirim 192.168.10.11 s/d 192.168.10.20 menjadi IP Address 30.1.1.2. Perintah yang dapat digunakan adalah sebagai berikut.

```
[admin@Gateway] > ip firewall nat add chain=srcnat \
src-address=192.168.10.2-192.168.10.10 action=src-nat \
to-addresses=60.1.1.2

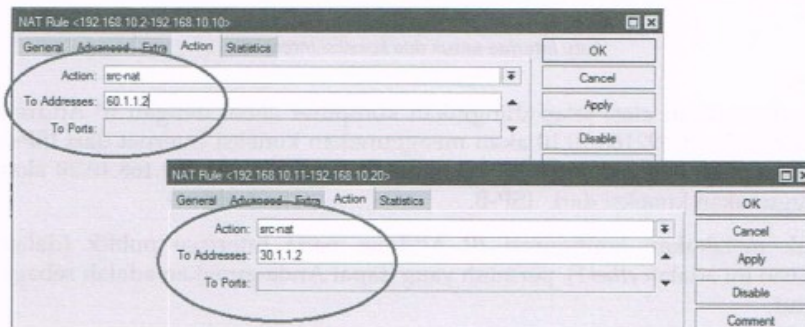
[admin@Gateway] > ip firewall nat add chain=srcnat \
src-address=192.168.10.11-192.168.10.20 action=src-nat \
to-addresses=30.1.1.2

[admin@Gateway] > ip firewall nat print
Flags: X - disabled, I - invalid, D - dynamic
0 chain=srcnat action=src-nat to-addresses=60.1.1.2
src-address=192.168.10.2-192.168.10.10

1 chain=srcnat action=src-nat to-addresses=30.1.1.2
src-address=192.168.10.11-192.168.10.20
```

Jika menggunakan WinBox, maka konfigurasi *action src-nat* akan terlihat seperti berikut.

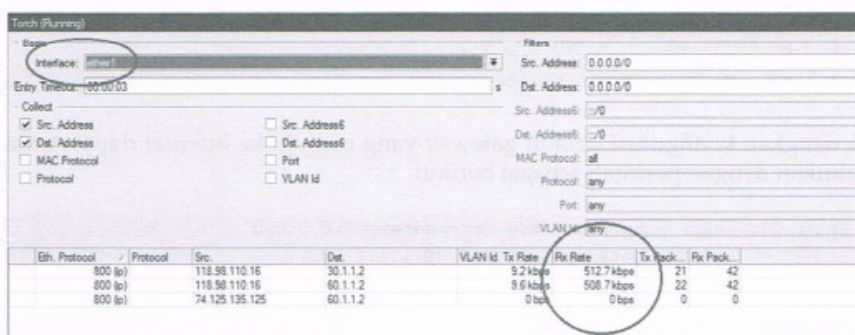
6



Gambar 6.11
Action src-nat pada WinBox

Sedangkan hasil konfigurasi tersebut jika dilihat dengan menggunakan WinBox akan terlihat seperti berikut.

Jika kedua group komputer client menggunakan seluruh alokasi bandwidth yang diberikan oleh masing-masing ISP, maka pemakaian bandwidth pada interface *ether1* yang dimonitor dengan menggunakan *torch* akan terlihat seperti gambar berikut.



Gambar 6.14

Traffic pada interface *ether1* yang terhubung ke dua ISP

6

Parameter lain

Contoh-contoh skenario yang diberikan pada sub bab Source NAT tadi lebih banyak memperlihatkan jaringan yang menggunakan dua koneksi Internet. Dan konfigurasinya juga lebih merujuk kepada teknik membagi beban melalui beberapa koneksi Internet yang tersedia (*load balance*). Namun, untuk benar-benar mewujudkan teknik *load balance* yang benar-benar handal dan efisien, Anda tidak bisa hanya mengandalkan parameter-parameter seperti *src-nat*, *out-interface*, *masquerade*.

Akan banyak parameter dan konfigurasi tambahan yang nantinya dibutuhkan seperti fitur *check-gateway* maupun *routing-mark* pada saat Anda akan mengkonfigurasi default gateway.

Tujuan dari sub bab ini adalah mematangkan “kuda-kuda” Anda dalam mengkonfigurasi firewall, khususnya Source NAT. Dengan memahami benar apa tujuan dan hasil konfigurasi dari action *masquerade* maupun *src-nat*, maka diharapkan Anda tidak akan kesulitan lagi pada saat akan menerapkan teknik *load balance* yang rumit sekalipun.


```
[admin@Gateway] > ip address add address=192.168.10.1/24
interface=ether2
```

```
[admin@Gateway] > ip address print
Flags: X - disabled, I - invalid, D - dynamic
# ADDRESS NETWORK INTERFACE
0 60.1.1.2/30 60.1.1.0 ether1
1 30.1.1.2/30 30.1.1.0 ether1
2 192.168.10.1/24 192.168.10.0 ether2
```

Sedangkan konfigurasi default gateway yang menuju ke Internet dapat Anda lakukan dengan perintah sebagai berikut

```
[admin@Gateway] > ip route add dst-address=0.0.0.0/0 \
gateway=60.1.1.1,30.1.1.1
```

```
[admin@R1] > ip route print
Flags: X - disabled, A - active, D - dynamic, C - connect, S - static,
r - rip, b - bgp, o - ospf, m - mme, B - blackhole, U - unreachable, P -
prohibit
# DST-ADDRESS PREF-SRC GATEWAY DISTANCE
0 A S 0.0.0.0/0 60.1.1.1 1
30.1.1.1
1 ADC 30.1.1.0/30 30.1.1.2 ether1 0
2 ADC 60.1.1.0/30 60.1.1.2 ether1 0
3 ADC 192.168.10.0/24 192.168.10.1 ether2 0
```

Adapun konfigurasi Firewall NAT dengan parameter *action=src-nat* pada interface *ether1*, dapat Anda lakukan dengan menggunakan perintah sebagai berikut.

```
[admin@Gateway] > ip firewall nat add chain=srcnat \
src-address=192.168.10.2-192.168.10.10 action=src-nat \
to-addresses=60.1.1.2

[admin@Gateway] > ip firewall nat add chain=srcnat \
src-address=192.168.10.11-192.168.10.20 action=src-nat \
to-addresses=30.1.1.2

[admin@Gateway] > ip firewall nat print
Flags: X - disabled, I - invalid, D - dynamic
0 chain=srcnat action=src-nat to-addresses=60.1.1.2
src-address=192.168.10.2-192.168.10.10

1 chain=srcnat action=src-nat to-addresses=30.1.1.2
src-address=192.168.10.11-192.168.10.20
```

Menjalankan Web Server pada LAN

Untuk menjalankan Web Server pada jaringan lokal sesuai skenario pada Gambar 6.15, maka harus dipastikan Web Server tersebut menggunakan IP Address 192.168.20.3/24 dengan gateway 192.168.20.1. Pastikan juga Web Server tersebut mengkonfigurasi DNS Server ke IP Address 192.168.20.1 atau DNS dari ISP dengan IP Address 60.1.1.1.

Saya tidak akan membahas bagaimana mengkonfigurasi parameter-parameter tersebut pada Web Server, umumnya Web Server dibangun dengan menggunakan mesin Linux, dan ini kan buku MikroTik, akan merusak “suasana” jika kita membahasnya. Namun Anda tidak perlu khawatir, pada lain kesempatan saya akan membahasnya, bagaimana membangun Web Server yang baik, benar dan sehat ☺.

Setelah Anda mengkonfigurasi IP Address pada Web Server, konfigurasi IP Address 192.168.20.1/24 pada interface *ether2* Router MikroTik. Kemudian pastikan network 192.168.10.0/24 dan 192.168.20.0/24 dapat mengakses Internet dengan menggunakan perintah konfigurasi firewall NAT seperti berikut ini.

```
[admin@Gateway] > ip firewall nat add chain=srcnat \
out-interface=ether1 action=masquerade

[admin@Gateway] > ip firewall nat print
Flags: X - disabled, I - invalid, D - dynamic
0 chain=srcnat action=masquerade out-interface=ether1
```

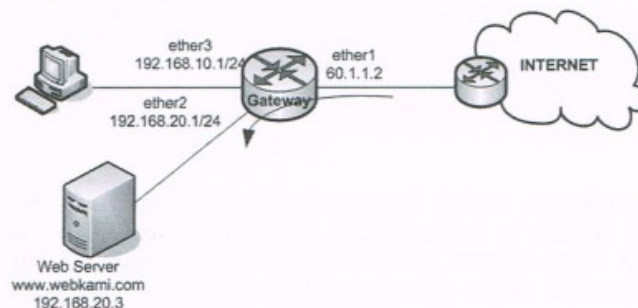
Langkah selanjutnya adalah bagaimana memasukkan *traffic* HTTP yang berasal dari Internet ke dalam jaringan lokal, untuk kemudian menuju Web Server Anda. Perlu diingat, Web Server tersebut tersembunyi dari Internet karena sebelumnya sudah dikonfigurasi *action masquerade* pada interface *ether1*. Pengguna yang ada di Internet hanya akan melihat interface *ether1* dari Router MikroTik dengan IP Address 60.1.1.2. Singkatnya, mereka hanya bisa melakukan ping ke IP Address 60.1.1.2 namun tidak dapat melakukan ping ke IP Address 192.168.20.3.

Jika ternyata konfigurasi Destination NAT dilakukan dengan tidak benar, maka pada saat pengguna di Internet membuka domain www.webkami.com, yang terlihat di browser mereka bukanlah tampilan dari Web Server, melainkan tampilan dari WebFig dari Router MikroTik Anda. Bukankah domain www.webkami.com di Internet dikenal sebagai 60.1.1.2 yang merupakan IP Address *ether1* dari Router MikroTik. Kesalahan konfigurasi tersebut akan mengakibatkan tampilan seperti gambar berikut:

Destination NAT

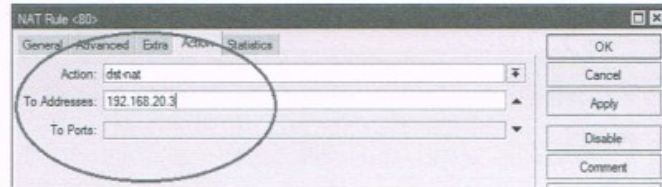
Pada buku *MikroTik Kung Fu Kitab 1*, Destination NAT ditujukan untuk melakukan pengalihan *traffic* browsing (paket HTTP) yang menuju Internet untuk belok ke Internal Proxy maupun Eksternal Proxy. Untuk melakukan pembelokan (*redirect*) ke Internal Proxy yang dilakukan adalah merubah *dst-port* 80 (HTTP) menjadi *dst-port* 8080 yang merupakan port dimana Internal Proxy bekerja. Sedangkan untuk melakukan *redirect* ke External Proxy, yang dilakukan adalah merubah IP Address tujuan (*dst-address*) yang ada paket data menjadi IP Address dari External Proxy, begitu juga port tujuan 80 dirubah menjadi port 8080, port di mana External Proxy bekerja.

Adapun penggunaan Destination NAT pada buku ini adalah konfigurasi yang diperlukan pada saat Anda menjalankan server-server Internet, namun server tersebut menggunakan IP Address privat. Sebagai contoh skenario, Anda ingin menjalankan Web Server, namun hanya memiliki sebuah IP Address publik yang sudah digunakan oleh interface *ether1* Router MikroTik. Akhirnya diambil keputusan bahwa Web Server tersebut cukup menggunakan IP Address privat, namun harus tetap dapat diakses dari Internet. Contoh skenario ini dapat dilihat pada gambar berikut.



Gambar 6.15
Web Server dengan IP Address Private

Pada gambar di atas, Web Server menggunakan IP Address 192.168.20.3 dan terhubung ke interface *ether2*. Web Server tersebut juga sudah memiliki domain, sebagai contoh *www.webkami.com*. Karena Web Server tersebut akan diakses dari Internet dan IP Address 192.168.20.3 tidak dikenal di Internet (karena merupakan IP Address privat), maka domain *www.webkami.com* akan dipetakan ke IP Address 60.1.1.2.



Gambar 6.18
Parameter action dst-nat pada WinBox

Jika Anda menginginkan user di Internet mengakses layanan FTP di Web Server, maka harus dilakukan penambahan konfigurasi Destination NAT dengan parameter *destination port* 20 dan 21. Web Server yang baik dan sehat juga harus menjalankan FTP Server yang sehat pula. Web Server yang tidak menjalankan FTP Server dengan baik umumnya akan bermasalah dengan beberapa Content Management System (CMS), terutama pada saat akan melakukan upload gambar 😊. Mudah-mudahan Web Server Anda tidak seperti itu. Baiklah, konfigurasi yang harus ditambahkan adalah sebagai berikut.

```
[admin@Gateway] > ip firewall nat add chain=dstnat \
in-interface=ether1 dst-address=60.1.1.2 protocol=tcp \
dst-port=20,21 action=dst-nat to-addresses=192.168.20.3
```

6

Secara lengkap konfigurasinya dapat dilihat seperti berikut.

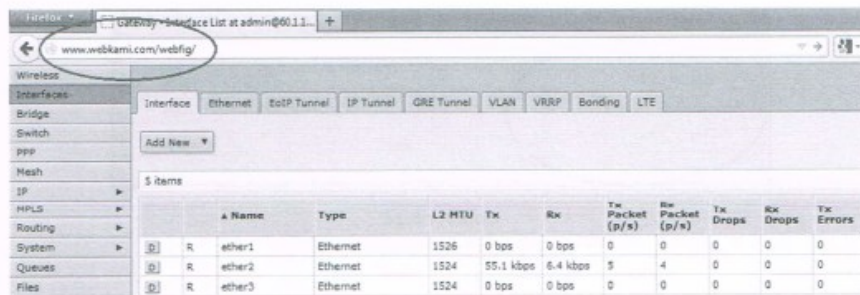
```
[admin@R1] > ip firewall nat print
Flags: X - disabled, I - invalid, D - dynamic
0 chain=srcnat action=masquerade out-interface=ether1

1 chain=dstnat action=dst-nat to-addresses=192.168.20.3
  protocol=tcp dst-address=60.1.1.2 in-interface=ether1
  dst-port=80

2 chain=dstnat action=dst-nat to-addresses=192.168.20.3
  protocol=tcp dst-address=60.1.1.2 in-interface=ether1
  dst-port=20,21
```

Sampai pada tahapan ini, pekerjaan memasukkan traffic HTTP dari Internet ke Web Server yang ada di dalam jaringan lokal telah selesai. Pengujian dapat dilakukan dengan mengakses www.webkami.com dari Internet.

Yang menjadi masalah lagi adalah bagaimana jika komputer-komputer client akan mengakses web www.webkami.com? Dengan hanya menggunakan



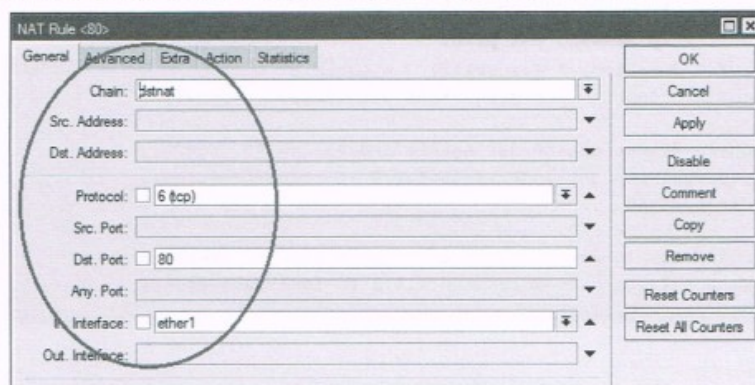
	Name	Type	L2 MTU	Tx	Rx	Tx Packet (p/s)	Rx Packet (p/s)	Tx Drops	Rx Drops	Tx Errors
☒	R ether1	Ethernet	1526	0 bps	0 bps	0	0	0	0	0
☒	R ether2	Ethernet	1524	55.1 kbps	6.4 kbps	5	4	0	0	0
☒	R ether3	Ethernet	1524	0 bps	0 bps	0	0	0	0	0

Gambar 6.16
Domain www.webkani.com menampilkan WebFig

Untuk mengkonfigurasi Destination NAT sehingga pengguna Internet dapat mengakses Web Server yang ada di jaringan lokal, maka perintah yang dapat digunakan adalah sebagai berikut.

```
[admin@Gateway] > ip firewall nat add chain=dstnat \
in-interface=ether1 dst-address=60.1.1.2 protocol=tcp \
dst-port=80 action=dst-nat to-addresses=192.168.20.3
```

Dengan menggunakan WinBox, maka parameter *chain dstnat*, *protocol* dan *in-interface* yang digunakan akan terlihat seperti gambar berikut.



Gambar 6.17
Chain *dstnat*, *protocol* dan *in-interface* WinBox

Sedangkan parameter action yang harus dikonfigurasi jika menggunakan WinBox akan terlihat seperti gambar berikut:

Dan dengan menggunakan WinBox, rule-rule firewall tersebut akan terlihat seperti berikut:

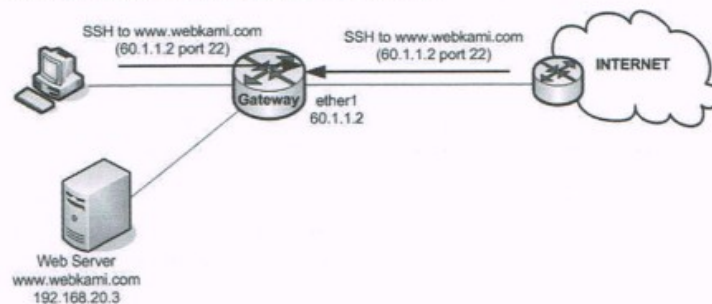
Firewall											
Filter Rules		NAT	Mangle	Service Ports	Connections	Address Lists	Layer7 Protocols				
+ -		[icon] [icon]		[icon] [icon]		[icon] [icon]		Reset Counters			
						oo		Reset All Counters			
#	Action	Chain	Out. Address	Protocol	Out. Port	In. Inter.	Out. Interf.	To Addresses	To Ports	Bytes	Packets
0	masquerade	srcnat					ether1			85.1 KB	1520
1	dst-nat	dstnat	60.1.1.2	tcp	80	ether1		192.168.20.3		1716 B	33
2	dst-nat	dstnat	60.1.1.2	tcp	20,21	ether1		192.168.20.3		52 B	1
3	dst-nat	dstnat	60.1.1.2	tcp	80	ether3		192.168.20.3		432 B	9
4	dst-nat	dstnat	60.1.1.2	tcp	20,21	ether3		192.168.20.3		48 B	1

Gambar 6.19
Rule Firewall untuk Web Server Lokal

Akses SSH pada Web Server (Port Translation)

Bagaimana jika Anda menginginkan adanya akses SSH (*destination port 22*) terhadap Web Server? Ini juga akan sedikit menjadi masalah bila mana disini lain Anda tetap ingin menjalankan akses SSH pada Router Gateway. Kesalahan konfigurasi dapat membuat akses SSH yang ditujukan kepada Web Server akan tiba di Router Gateway, ataupun sebaliknya, pada saat akan me-remote Router Gateway dengan menggunakan SSH, Anda malah tiba di Web Server.

Untuk menjalankan skenario ini, sebenarnya bisa saja diterapkan seperti konfigurasi tadi, meneruskan akses SSH yang ditujukan untuk IP Address 60.1.1.2 ke Web Server 192.168.20.3. Namun ini akan mengakibatkan akses SSH tidak dapat lagi dilakukan pada Router MikroTik. Kekacauan traffic SSH ini dapat Anda perhatikan pada gambar berikut.



Gambar 6.20
Router MikroTik dan Web Server menjalankan SSH

Jika Anda ingin mendapatkan akses SSH baik pada Router Gateway maupun

konfigurasi-konfigurasi tadi, maka yang akan terlihat pada browser komputer client juga adalah WebFig dari Router MikroTik. Mengapa ini masih terjadi? Apakah konfigurasi tadi kurang cukup?

Konfigurasi tadi hanya mengalihkan traffic HTTP dan FTP yang menuju IP Address ke 60.1.1.2 ke Web Server 192.168.20.3. Yang perlu Anda perhatikan adanya parameter *in-interface=ether1* pada konfigurasi tersebut. Ini menunjukkan yang diproses hanyalah *traffic* yang berasal dari Internet (*traffic* dari Internet akan menuju Web Server melalui *ether1*).

Konfigurasi tadi tidak mengatur bagaimana jika *traffic* HTTP/FTP berasal dari komputer client yang harus menggunakan interface *ether3* untuk menuju Web Server. Untuk mengatasi masalah ini, konfigurasi yang perlu ditambahkan lagi adalah sebagai berikut.

```
[admin@Gateway] > ip firewall nat add chain=dstnat \
in-interface=ether3 dst-address=60.1.1.2 protocol=tcp \
dst-port=80 action=dst-nat to-addresses=192.168.20.3

[admin@Gateway] > ip firewall nat add chain=dstnat \
in-interface=ether3 dst-address=60.1.1.2 protocol=tcp \
dst-port=20,21 action=dst-nat to-addresses=192.168.20.3
```

Sehingga hasil akhir konfigurasi dapat dilihat seperti berikut.

```
[admin@R1] > ip firewall nat print
Flags: X - disabled, I - invalid, D - dynamic
0 chain=srcnat action=masquerade out-interface=ether1

1 chain=dstnat action=dst-nat to-addresses=192.168.20.3
protocol=tcp dst-address=60.1.1.2 in-interface=ether1
dst-port=80

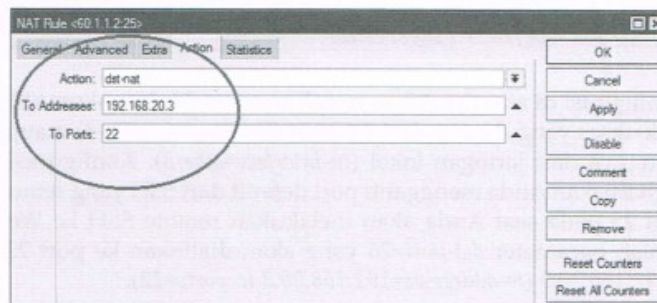
2 chain=dstnat action=dst-nat to-addresses=192.168.20.3
protocol=tcp dst-address=60.1.1.2 in-interface=ether1
dst-port=20,21

3 chain=dstnat action=dst-nat to-addresses=192.168.20.3
protocol=tcp dst-address=60.1.1.2 in-interface=ether3
dst-port=80

4 chain=dstnat action=dst-nat to-addresses=192.168.20.3
protocol=tcp dst-address=60.1.1.2 in-interface=ether3
dst-port=20,21
```

Dengan sedikit eksperimen, kondisi tersebut tentu saja dapat saja dibuat sebaliknya. Jika ingin melakukan remote ke Web Server cukup gunakan port 22 sedangkan jika ingin me-remote Router Gateway port yang digunakan adalah 25. Silahkan berimajinasi dengan port-port tersebut atau dengan nomor port lain J. Bila “kuda-kuda” firewall Anda matang, kondisi skenario jaringan apapun dapat dengan mudah dapat Anda perankan dan wujudkan. Seperti bintang film dengan kemampuan akting mumpuni akan mampu memerankan skenario apapun.

Dengan menggunakan WinBox, maka konfigurasi yang harus Anda lakukan dapat dilihat pada gambar berikut:



Gambar 6.22

Action dst-nat untuk akses SSH Web Server

Secara keseluruhan, penggunaan rule Firewall NAT dengan chain dst-nat baik dari skenario awal Web Server sampai dengan Akses SSH tadi, akan terlihat seperti gambar berikut.

#	Action	Chain	Src. Address	Dest. Address	Protocol	Out. Port	In. Interface	Out. Interface	To Addresses	To Ports	Bytes	Packets
0	masquerade	srcnat									30.3 K8	570
1	dst-nat	dstnat	60.1.1.2	60.1.1.2	6 tcp	80	ether1	ether1	192.168.20.3		0 B	0
2	dst-nat	dstnat	60.1.1.2	60.1.1.2	6 tcp	20.21	ether1		192.168.20.3		0 B	0
3	dst-nat	dstnat	60.1.1.2	60.1.1.2	6 tcp	80	ether3		192.168.20.3		0 B	0
4	dst-nat	dstnat	60.1.1.2	60.1.1.2	6 tcp	20.21	ether3		192.168.20.3		0 B	0
5	dst-nat	dstnat	60.1.1.2	60.1.1.2	6 tcp	25	ether1		192.168.20.3	22	0 B	0
6	dst-nat	dstnat	60.1.1.2	60.1.1.2	6 tcp	25	ether3		192.168.20.3	22	0 B	0

Gambar 6.23

Rule Firewall lengkap untuk Destination NAT

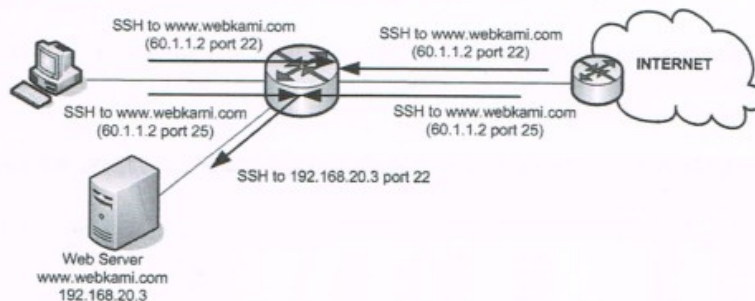
pada Web Server, maka salah satu solusinya adalah menerapkan konfigurasi Destination NAT dengan melakukan perubahan *field* port address tujuan (*dst-port*). Contoh perintah yang dapat digunakan adalah sebagai berikut.

```
[admin@Gateway] > ip firewall nat add chain=dstnat \
dst-address=60.1.1.2 protocol=tcp dst-port=25 \
in-interface=ether1 action=dst-nat \
to-addresses=192.168.20.3 to-ports=22

[admin@Gateway] > ip firewall nat add chain=dstnat \
dst-address=60.1.1.2 protocol=tcp dst-port=25 \
in-interface=ether3 action=dst-nat \
to-addresses=192.168.20.3 to-ports=22
```

Kedua konfigurasi di atas ditujukan untuk memungkinkan akses SSH ke Web Server baik akses yang datang dari Internet (*in-interface=ether1*) maupun akses yang datang dari jaringan lokal (*in-interface=ether3*). Konfigurasi tersebut juga mengharuskan Anda mengganti port default dari SSH yang semula adalah 22 menjadi 25 pada saat Anda akan melakukan remote SSH ke Web Server. Perhatikanlah parameter *dst-port=25* yang akan dialihkan ke port 22 pada IP Address 192.168.20.3 (*to-addresses=192.168.20.3 to-ports=22*).

Jika digambarkan kembali maka implementasi perubahan destination port dengan menggunakan *action dst-nat*, maka akan terlihat seperti berikut.



Gambar 6.21
Implementasi port translation

Bila pada saat melakukan akses SSH, Anda tidak mengganti port default dari akses SSH, maka akses tersebut akan tiba di Router MikroTik. Kesimpulannya, Anda harus menggunakan port 22 untuk melakukan remote Router Gateway dan port 25 pada saat akan meremote Web Server.

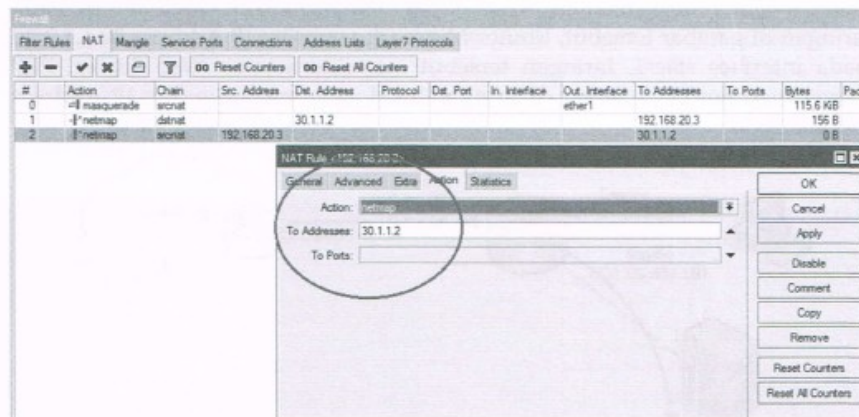

```
src-address=192.168.20.3 action=netmap
to-addresses=30.1.1.2

[admin@Gateway] > ip firewall nat print
Flags: X - disabled, I - invalid, D - dynamic
0 chain=srcnat action=masquerade out-interface=ether1

1 chain=dstnat action=netmap to-addresses=192.168.20.3 dst-ad-
dress=30.1.1.2

2 chain=srcnat action=netmap to-addresses=30.1.1.2 src-ad-
dress=192.168.20.3
```

Bila menggunakan WinBox, maka contoh parameter *action* yang harus dilakukan dapat dilihat pada gambar berikut.



Gambar 6.25
Konfigurasi action netmap menggunakan WinBox

Dengan konfigurasi seperti di atas tadi, maka IP Address 30.1.1.2 sebenarnya adalah Web Server dengan IP Address 192.168.20.3. Untuk mengujinya, cobalah men-shutdown Web Server, kemudian lakukanlah ping dari Internet terhadap IP Address 30.1.1.2. Hasil yang didapatkan adalah *Request Time Out*, padahal IP Address 30.1.1.2 tidak pernah *down*. Yang sebenarnya *down* adalah 192.168.20.3, namun karena sudah dipetakan menjadi 30.1.1.2. Maka pada saat Web Server *down*, IP Address 30.1.1.2 akan terlihat *down* pula.

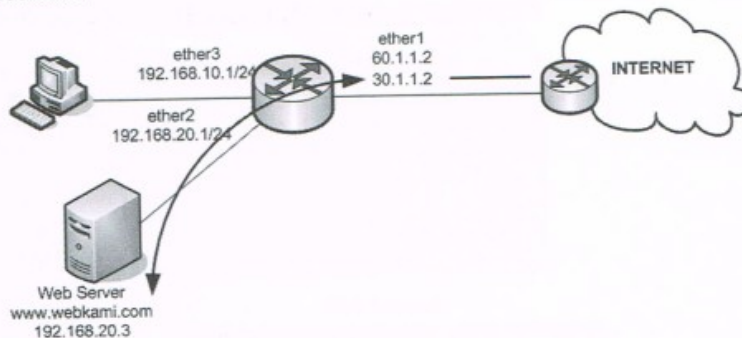
---END OF CHAPTER---

1 : 1 Mapping

Jika Anda terlalu pusing dengan urusan *srcnat* dan *dstnat* untuk memasukkan *traffic* dari Internet ke Web Server pada jaringan lokal, maka Anda dapat memanfaatkan *action netmap*. Hanya saja, Anda harus memiliki satu lagi IP Address publik yang akan digunakan oleh *netmap*.

Netmap dapat digunakan untuk membuat pemetaan langsung IP Address public kepada IP Address privat yang ada di jaringan lokal, sering disebut dengan *1 : 1 Mapping*. Pemetaan ini dapat diterapkan kepentingan Destination NAT maupun Source NAT. Juga dapat dilakukan untuk pemetaan sebuah IP Address publik yang kepada IP Address privat, maupun sekelompok IP Address publik ke sekelompok IP Address privat.

Sebagai contoh implementasi, Anda dapat memperhatikan Gambar 6.24. Pada jaringan di gambar tersebut, Router MikroTik memiliki IP Address 30.1.1.2/30 pada interface *ether1*. Jaringan tersebut memiliki sebuah Web Server, FTP Server maupun DNS Server di jaringan lokal yang menggunakan IP Address 192.168.20.3.



Gambar 6.24

Pemetaan 1:1 dari IP Address 30.1.1.2 ke 192.168.20.3

Skenario jaringan pada gambar di atas menginginkan IP Address 30.1.1.2 dipetakan langsung ke 192.168.20.3 maka perintah yang dapat digunakan adalah sebagai berikut.

```
[admin@Gateway] > ip firewall nat add chain=dstnat \
dst-address=30.1.1.2 action=netmap to-addresses=192.168.20.3
[admin@Gateway] > ip firewall nat add chain=srcnat \
```

Ada tiga jenis *marking* yang dapat digunakan, yaitu :

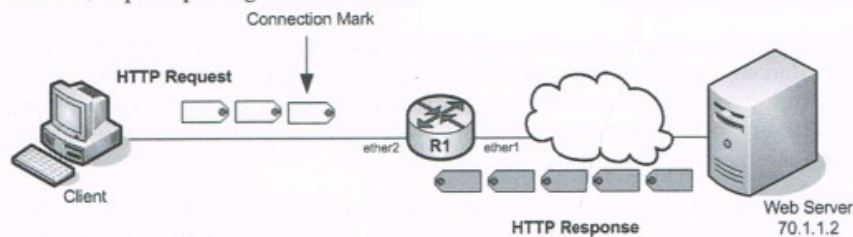
- Connection Mark
- Packet Mark.
- Route Mark.

Ketiga jenis *marking* tersebut akan dibahas pada sub bab terpisah. Saya ingin Anda dapat memiliki sedikit pemahaman tentang konsep penggunaan mangle. Dengan tujuan agar tidak terjebak dengan teknik copy paste konfigurasi mangle, yang belum tentu sesuai dan efisien jika diterapkan pada jaringan Anda. Dengan memahami konsep *marking* paket dengan baik, maka Anda dapat menyusun sendiri rule-rule *marking* yang dibutuhkan. Ini akan dapat menghasilkan firewall mangle yang memperlihatkan "*taste*" Anda ☺.

Connection Mark

Connection Mark merupakan jenis *marking* yang digunakan untuk menandai adanya suatu koneksi. Kita sudah mengetahui, bahwa sebuah komputer yang akan melakukan komunikasi akan mengeluarkan serangkaian paket (*data stream*). Nah, Connection Mark ini digunakan untuk memberikan tanda untuk paket yang pertama kali keluar dari komputer tersebut.

Connection Mark dapat digunakan untuk memberikan tanda (*marking*) pada paket pertama, baik paket pertama yang merupakan request dari client, maupun paket pertama yang merupakan response dari server. Sebelum kita mencoba beberapa konfigurasi *marking* koneksi, ada baiknya Anda melihat ilustrasi sebuah client yang sedang browsing ke salah satu Web Server di Internet, seperti pada gambar berikut.



Gambar 7.1
Connection Mark pada HTTP Request

Pada gambar di atas terlihat sebuah client yang melakukan browsing ke sebuah Web Server di Internet. Yang pertama dikeluarkan oleh client adalah

Bab 7

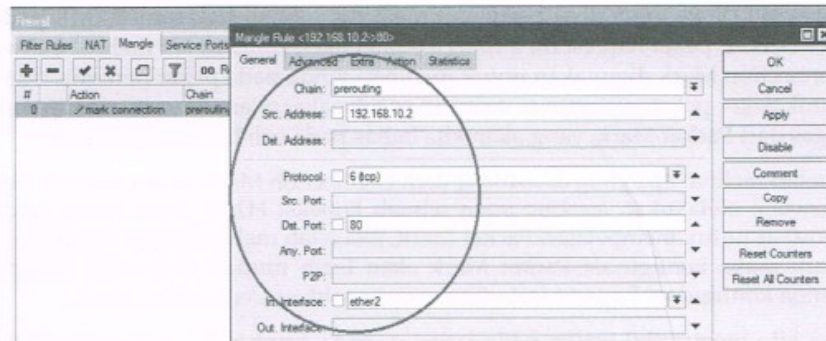
Firewall (Mangle)

Mangle merupakan salah satu fitur pada firewall Router MikroTik yang digunakan untuk memberi tanda (*mark*) pada paket data. Kadang pekerjaan memberi tanda ini disebut *marking*, dan pekerjaan ini wajib Anda kuasai dengan benar, selain tentunya *routing* maupun *bridging*. Tujuan memberikan tanda ini dimaksudkan agar paket tersebut lebih mudah dikenali lagi, yang pada akhirnya akan mempermudah Anda menerapkan filter, masquerade, routing maupun pada saat akan melakukan manajemen bandwidth.

Sebenarnya, Anda bisa saja tidak menggunakan *marking* untuk mengenali sebuah paket data, bukankah dapat saja menggunakan parameter-parameter *source/destination address*, *source/destination port*, *content*, *L7 protocol* pada firewall filter dan NAT? Dengan hanya menggunakan filter dan NAT itu akan sangat merepotkan dan terkadang tidak akurat. Inilah yang mendasari penggunaan mangle dalam manajemen jaringan, untuk mempermudah Anda dalam mengelola paket.

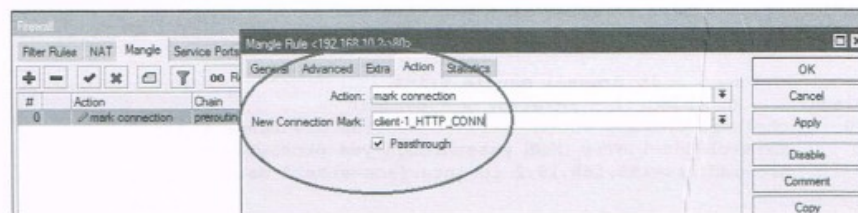
Untuk dapat menggunakan fitur mangle dengan tepat, Anda harus mengetahui arah dan tujuan paket data, jenis-jenis *protocol*, penggunaan *source/destination port*, maupun *connection state* dari suatu paket data. Singkatnya, modal yang digunakan pada saat menerapkan firewall filter, akan digunakan juga pada saat akan menerapkan firewall mangle.

Tanda atau *marking* yang diberikan kepada paket data, hanya dibaca dan digunakan pada router yang bersangkutan. *Marking* tersebut akan dilepas pada saat paket akan meninggalkan router. Sehingga *marking* pada suatu router tidak dapat digunakan pada router lain. Firewall mangle juga nantinya akan terdiri dari susunan rule-rule, sama seperti pada firewall filter maupun NAT. Rule-rule tersebut juga akan dibaca secara berurutan dari atas ke bawah (*top to bottom*).



Gambar 7.2
Connection Mark untuk traffic HTTP

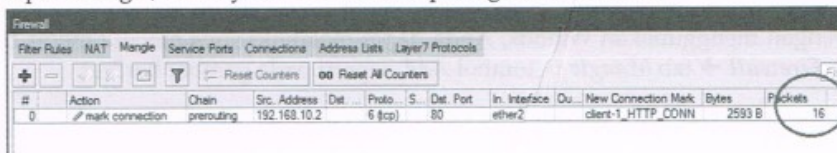
Sedangkan untuk *action*, dapat dilakukan pada tab *Action*, seperti pada gambar berikut.



Gambar 7.3
Action mark connection

7

Untuk mengujinya, cobalah melakukan browsing ke salah satu situs di Internet, misalnya www.google.com, kemudian perhatikanlah *Counter Packets* yang ada di tampilan oleh WinBox. Walaupun hanya membuka halaman depan dari Google, akan terlihat bahwa komputer client Anda sebenarnya membuat beberapa koneksi sekaligus (simultan). Beberapa koneksi ini bisa saja digunakan untuk membuka beberapa *link* maupun *images* yang ada di halaman depan Google, hasilnya akan terlihat seperti gambar berikut.



Gambar 7.4
Counter Packet untuk Connection Mark

paket HTTP Request yang bertujuan meminta sebuah halaman HTML pada Web Server, paket request ini terdiri 3 buah paket. Anda dapat melihat bahwa Connection Mark digunakan untuk memberi tanda pada paket pertama, tidak untuk paket-paket selanjutnya. Adapun untuk paket selanjutnya itu merupakan tugas dari Packet Mark, yang akan kita bahas pada sub bab selanjutnya J.

Setelah melihat apa yang di-marking oleh Connection Mark, maka marilah kita mencoba melakukan *marking* pada sebuah koneksi HTTP. Saya harap Anda bersabar untuk mempelajari Packet Mark, jika telah mahir dengan Connection Mark maka konfigurasi Packet Mark akan lebih mudah. Nantinya, hampir semua konfigurasi Packet Mark akan membutuhkan Connection Mark.

Jika kita mengambil acuan pada jaringan pada gambar 7.1, komputer client adalah 192.168.10.2 dan untuk melakukan *marking* koneksi terhadap aktifitas browsing (HTTP), maka perintah yang dapat digunakan adalah sebagai berikut.

```
[admin@Gateway] > ip firewall mangle add chain=prerouting \
src-address=192.168.10.2 protocol=tcp dst-port=80 \
in-interface=ether2 action=mark-connection \
new-connection-mark=client-1_HTTP_CONN
```

```
[admin@Gateway] > ip firewall mangle print
Flags: X - disabled, I - invalid, D - dynamic
0 chain=prerouting action=mark-connection new-connection-
mark=client-1_HTTP_CONN passthrough=yes protocol=tcp
src-address=192.168.10.2 in-interface=ether2 dst-port=80
```

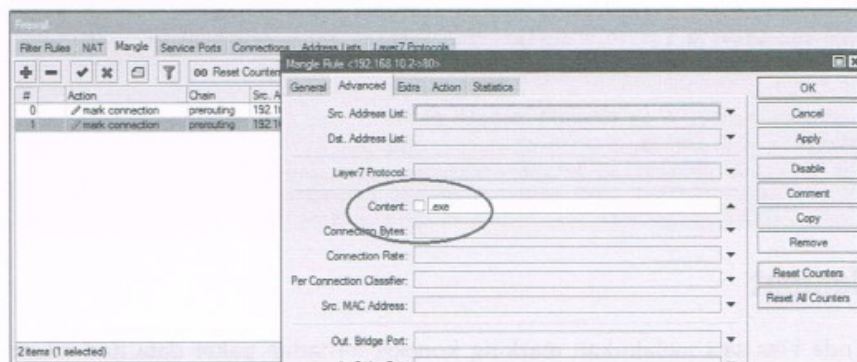
Parameter yang perlu Anda perhatikan adalah *in-interface* yang menggunakan *ether2*, ini karena HTTP Request yang keluar dari client yang menuju Internet akan masuk pertama kali ke Router pada interface *ether2*. Parameter *passthrough* yang digunakan juga adalah *yes*.

Begitu juga dengan parameter *chain prerouting*, chain ini dipilih karena merupakan chain yang digunakan untuk melakukan *marking* bagi paket yang ditujukan untuk router atau paket yang akan melintasi router. *Chain prerouting* juga merupakan chain yang mengizinkan penggunaan parameter *in-interface*. Anda tidak bisa menggunakan *chain postrouting*, karena *chain postrouting* tidak mengizinkan Anda memilih parameter *in-interface*.

Dengan menggunakan WinBox, Anda dapat melakukannya melalui menu **IP** → **Firewall** → tab **Mangle** → tombol **Add**, seperti pada gambar berikut:


```
1 chain=prerouting action=mark-connection new-connection-
mark=client-1_HTTP_exe_CONN passthrough=yes
protocol=tcp src-address=192.168.10.2 in-interface=ether2 dst-
port=80 content=.exe
```

Parameter *content* dapat Anda konfigurasi pada tab *Advanced*, seperti pada gambar berikut.



Gambar 7.5
Penggunaan parameter *content*

Yang perlu diperhatikan adalah parameter *passthrough* yang menentukan apakah paket data akan diteruskan pada rule berikutnya. Harus diingat bahwa pada firewall, paket akan dicocokkan pada baris-baris rule secara berurutan dari atas ke bawah.

Anda bisa melihat susunan *marking* koneksi pada gambar 7.6, dimana opsi pada *marking* koneksi HTTP menggunakan parameter *passthrough=no*. Sedangkan di bawahnya ada sebuah rule *marking* koneksi yang ditujukan untuk koneksi HTTP dengan ekstensi file *.exe*.

#	Action	Chain	Src. Address	Dest. Address	Proto.	S.	Dst. Port	In. Intef.	Out. Intef.	Content	Passthrough	New Connection Mark	Bytes	Packets
0	mark connection	prerouting	192.168.10.2	8.8.8.8	tcp	80	ether2			no	no	client-1_HTTP_CONN	222.5 KB	4 007
1	mark connection	prerouting	192.168.10.2	8.8.8.8	tcp	80	ether2			.exe	yes	client-1_HTTP_exe_CONN	8 B	0

Gambar 7.6
Passthrough pada baris pertama Connection Mark

Karena baris pertama tidak akan meneruskan paket ke baris kedua, maka pada saat komputer client mendownload file *.exe*, rule *marking* koneksi yang kedua tidak dapat "menangkap" paket tersebut, terlihat bahwa *Counter*

Contoh konfigurasi Connection Mark di atas adalah konfigurasi yang ditujukan untuk menandai koneksi HTTP yang dilakukan oleh komputer client 192.168.10.2. Jika ingin melakukan *marking* pada semua jenis koneksi dari client tersebut, tanpa melihat jenis koneksi Internet apa yang dilakukannya, maka Anda tidak perlu memasukkan paramater *protocol* dan *dst-port*, seperti terlihat berikut ini.

```
[admin@Gateway] > ip firewall mangle add chain=prerouting \
src-address=192.168.10.2 in-interface=ether2 action=mark-connection \
new-connection-mark=client-1_CONN

[admin@Gateway] > ip firewall mangle print
Flags: X - disabled, I - invalid, D - dynamic
0 chain=prerouting action=mark-connection new-connection-
mark=client-1_CONN passthrough=yes src-address=192.168.10.2 in-
interface=ether2
```

Content

Anda bisa saja melakukan marking koneksi terhadap paket data dari client yang melakukan download jenis-jenis file tertentu. Ini nantinya akan berguna saat akan melakukan pembatasan bandwidth pada aktifitas download.

Parameter yang dapat dimanfaatkan untuk *marking* koneksi berdasarkan ekstensi file adalah parameter *content*. Anda juga dapat memanfaatkan fitur *Layer 7 Protocol* untuk konfigurasi ini. Namun untuk pembahasan buku ini, parameter yang digunakan adalah *content*, yang mempunyai kelebihan tidak membutuhkan resource router yang besar dibandingkan jika Anda menggunakan parameter *L7 Protocol*. Penggunaan *L7 Protocol* akan dibahas pada buku MikroTik Kung Fu selanjutnya

Sebagai contoh, jika ingin melakukan *marking* koneksi terhadap aktifitas download untuk file dengan ekstensi *.exe*, perintah yang dapat digunakan adalah sebagai berikut:

```
[admin@Gateway] > ip firewall mangle add chain=prerouting \
src-address=192.168.10.2 protocol=tcp dst-port=80 content=.exe \
in-interface=ether2 action=mark-connection \
new-connection-mark=client-1_HTTP_exe_CONN

[admin@Gateway] > ip firewall mangle print
Flags: X - disabled, I - invalid, D - dynamic
0 chain=prerouting action=mark-connection new-connection-
mark=client-1_HTTP_CONN passthrough=yes protocol=tcp
src-address=192.168.10.2 in-interface=ether2 dst-port=80
```


Namun, kadang akan terlihat bahwa jumlah koneksi tersebut ada kalanya melebihi jumlah koneksi yang ada di Download Manager. Ini dikarenakan Download Manager akan membuat koneksi baru lagi pada saat satu sesi koneksi telah selesai melakukan download. Misalnya pada gambar 7.7 ada koneksi nomor 1, 2, 3 dan 4. Jika koneksi nomor 3 yang ternyata lebih dahulu selesai melakukan download, maka Download Manager akan membuka koneksi baru lagi, sehingga *Counter Packet* akan berjumlah lebih dari 4 (empat).

Connection Mark vs Internal Proxy

Bagaimana jika Router MikroTik Anda menjalankan proxy? Bagaimanakah cara melakukan *marking* terhadap koneksi yang dilakukan oleh proxy? Seperti sudah yang sudah di bahas pada *MikroTik Kung Fu Kitab 1*, jika Router MikroTik menjalankan proxy, maka yang akan melakukan koneksi HTTP ke Internet bukanlah komputer client, melainkan proxy itu sendiri.

Pada saat proxy melakukan koneksi ke Internet, koneksi tersebut dianggap berasal dari Router MikroTik itu sendiri. Oleh karena itu chain yang harus digunakan untuk melakukan *marking* terhadap koneksi proxy tersebut adalah *chain output*. Perintah yang dapat digunakan untuk melakukan *marking* bagi koneksi yang dilakukan oleh proxy ke Internet adalah sebagai berikut.

```
[admin@Gateway] > ip firewall mangle add chain=output \
out-interface=ether1 action=mark-connection \
new-connection-mark=proxy_CONN
```

7

Anda juga dapat menambahkan beberapa variasi konfigurasi, misalnya jika ingin melakukan *marking* terhadap koneksi yang dilakukan proxy, namun hanya bagi koneksi-koneksi yang melakukan download file dengan ekstensi .exe. Untuk melakukan itu, Anda cukup menambahkan parameter *content=.exe* pada contoh perintah di atas tadi.

Packet Mark

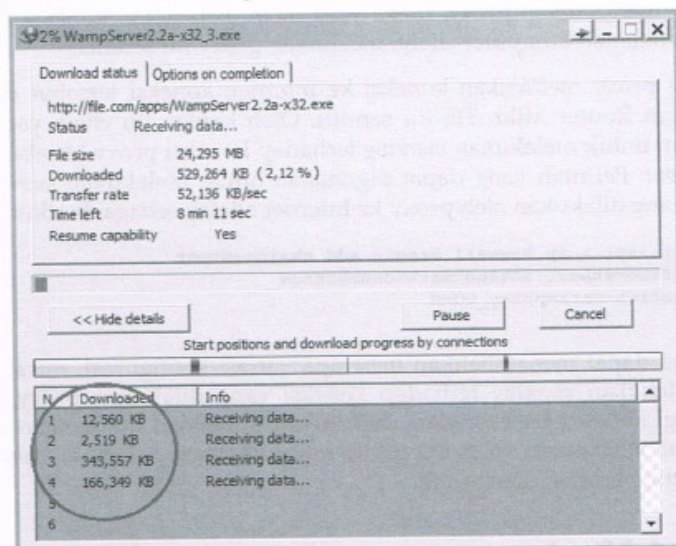
Pada sub bab sebelumnya sudah dibahas tujuan dan kegunaan dari Connection Mark. Selanjutnya Anda harus melakukan konfigurasi-konfigurasi Packet Mark, sehingga dapat melakukan *marking* pada keseluruhan paket data yang keluar dan masuk dari sebuah komputer client.

Bukankah Connection Mark hanya melakukan *marking* untuk paket yang “pertama” keluar dari komputer client? Bagaimana dengan paket-paket selanjutnya yang merupakan kelanjutan dari paket pertama tadi? Inilah fungsi

Packet-nya bernilai 0. Anda dapat saja mengurutkan kembali rule-rule firewall tersebut. Atau dapat juga membuat rule bekerja kembali dengan menggunakan parameter *passthrough=yes* pada baris pertama.

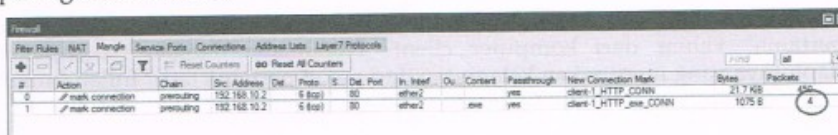
Connection Mark vs Download Manager

Kebanyakan komputer client akan menggunakan Download Manager untuk melakukan aktifitas download. Aplikasi Download Manager akan membuat beberapa koneksi secara bersamaan untuk pada saat akan mendownload. Ini akan menyebabkan *Counter Packet* pada Connection Mark juga akan bertambah sesuai dengan jumlah koneksi yang dibuat oleh Download Manager. Contoh suatu aplikasi Download Manager yang membuat 4 (empat) koneksi pada saat akan mendownload file .exe dapat dilihat pada gambar berikut.



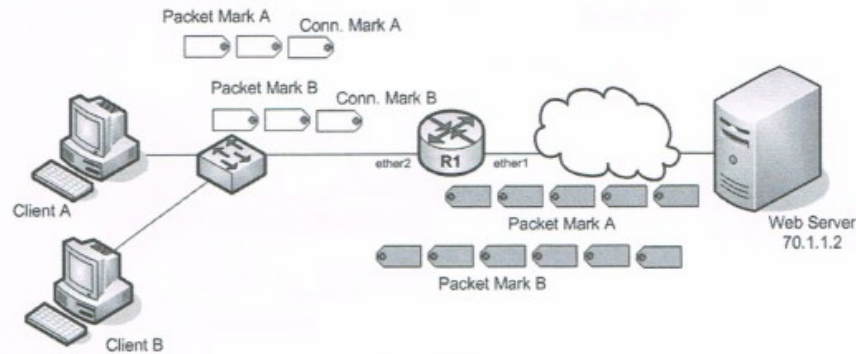
Gambar 7.7
Jumlah koneksi pada Download Manager

Pada saat Download Manager membuat 4 (empat) koneksi, maka pada Connection Mark akan terlihat *Counter Packet* sebanyak 4 (empat) juga, seperti pada gambar berikut.



Gambar 7.8
Jumlah Packet Koneksi yang dibuat oleh Download Manager

kelanjutan dari Connection Mark A, dan paket B merupakan kelanjutan dari Connection Mark B. Jangan sampai konfigurasi malah membuat Paket A dianggap kelanjutan dari Connection Mark B. Anda dapat melihat ilustrasinya seperti berikut.



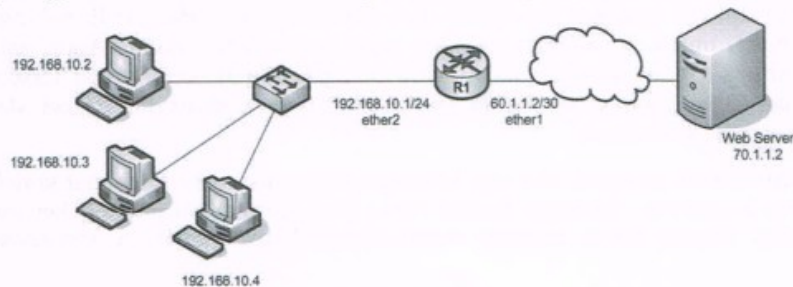
Gambar 7.10
Packet Mark A dan B dalam satu jaringan

Sehingga untuk menjaga agar *marking* paket berfungsi dengan benar, pada saat akan membuat konfigurasi Packet Mark harus diperhatikan parameter *connection-mark* yang merupakan nama dari Connection Mark yang sudah dibuat sebelumnya. Jadi, jika Connection Mark untuk client A diberi nama *client-A_CONN*, maka pada saat Anda akan membuat konfigurasi Packet Mark untuk client A, konfigurasi tersebut harus berisi parameter *connection-mark=client-A_CONN*. Anda akan melihat contoh-contohnya pada penjelasan berikutnya.

7

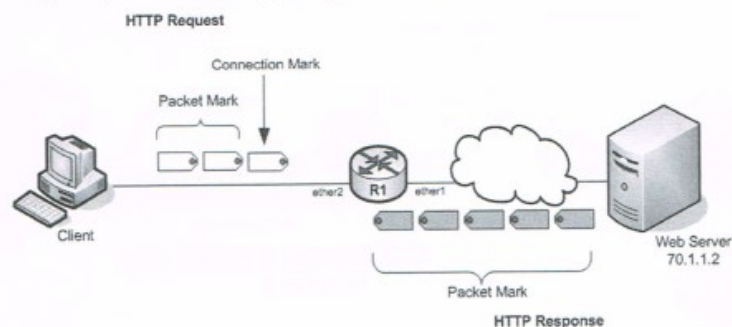
Skenario 1

Pada skenario 1 ini, Anda diharuskan membuat *marking* bagi keseluruhan *traffic* (baik upload maupun download) dari semua komputer client yang ada di jaringan 192.168.10.0/24. Anda dapat melihat topologinya seperti berikut.



Gambar 7.11
Marking untuk semua jenis koneksi

dari Packet Mark, melakukan *marking* pada paket-paket lanjutan setelah paket pertama tadi. Tetaplah ingat, bahwa dalam komunikasi TCP/IP, data yang dikirimkan akan dipecah-pecah sehingga akan membentuk *stream data*. Ilustrasinya dapat Anda lihat pada gambar berikut.



Gambar 7.9
Keseluruhan stream data HTTP dari komputer Client

Perhatikanlah gambar di atas, yang merupakan gambar yang memperlihatkan sebuah komputer client yang sedang melakukan aktifitas download dari suatu Web Server di Internet. Karena komunikasi TCP/IP umumnya adalah 2 arah, maka Anda melihat adanya HTTP Request yang berisi permintaan download ke Web Server dan HTTP Response yang merupakan data atau file yang diberikan Web Server ke komputer client. Itulah sebabnya dalam ilustrasi tersebut jumlah paket response lebih banyak (5 *packet*) dari jumlah paket request (3 *packet*). Bukankah pada umumnya *traffic* download lebih besar dari *traffic* upload, kecuali Anda termasuk pengguna yang suka meng-upload sesuatu ke Internet ☺.

Masih dari gambar 7.9, Anda dapat melihat bahwa Connection Mark digunakan untuk melakukan *marking* pada paket yang pertama. Sedangkan Packet Mark digunakan untuk melakukan *marking* pada paket-paket selanjutnya. Terlihat juga bahwa Packet Mark dapat digunakan untuk melakukan *marking* pada *traffic* upload maupun download dari koneksi yang terjadi tadi. Sehingga nantinya, untuk melakukan *marking* pada keseluruhan *traffic*, Anda harus membuat 3 buah konfigurasi firewall mangle. Ketiga konfigurasi tersebut adalah Connection Mark, Packet Mark untuk *traffic* upload dan Packet Mark untuk *traffic* download.

Dalam sebuah jaringan, bisa saja beberapa komputer client membuat koneksi secara bersamaan. Sehingga Router harus lebih jeli melihat paket-paket yang datang. Router harus mampu membedakan bahwa paket A merupakan


```

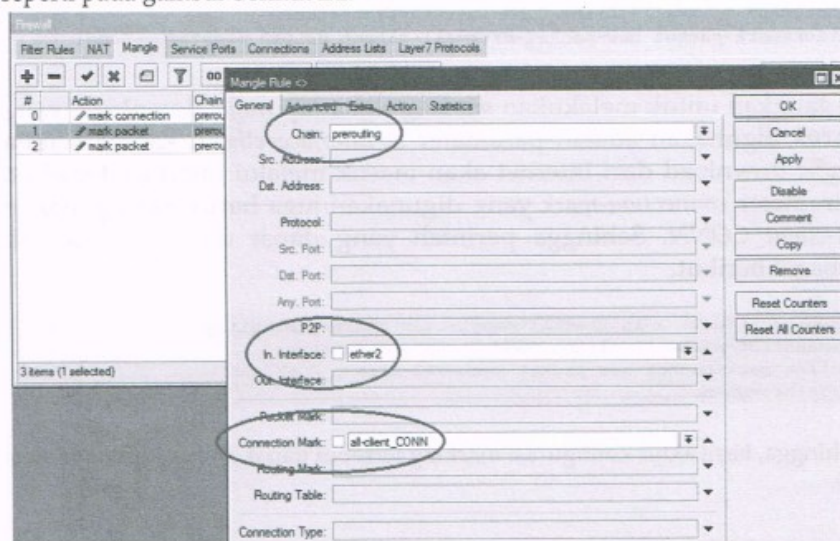
1 chain=prerouting action=mark-packet new-packet-mark=all-client_UPLOAD passthrough=no in-interface=ether2 connection-mark=all-client_CONN

2 chain=prerouting action=mark-packet new-packet-mark=all-client_DOWNLOAD passthrough=no in-interface=ether1 connection-mark=all-client_CONN

```

Jika melihat hasil konfigurasi *marking* di atas, maka keseluruhan *traffic* upload dari semua komputer client di jaringan 192.168.10.0/24 akan dikenal sebagai paket dengan nama *all-client_UPLOAD*. Untuk *traffic* downloadnya, akan dikenal sebagai paket dengan nama *all-client_DOWNLOAD*. Perhatikan juga parameter *passthrough=no* yang digunakan pada kedua konfigurasi Packet Mark.

Dengan menggunakan WinBox, maka menu yang dapat digunakan untuk membuat Packet Mark adalah *IP* → *Firewall* → tab *Mangle* → tombol *Add*, seperti pada gambar berikut ini.



Gambar 7.12
Konfigurasi Packet Mark melalui WinBox.

Sedangkan untuk mengkonfigurasi parameter *action* melalui WinBox, dapat dilakukan pada tab *Action*, seperti pada gambar berikut.