

Load Balance menggunakan Metode PCC

Load balance pada mikrotik adalah teknik untuk mendistribusikan beban trafik pada dua atau lebih jalur koneksi secara seimbang, agar trafik dapat berjalan optimal, memaksimalkan throughput, memperkecil waktu tanggap dan menghindari overload pada salah satu jalur koneksi.

Selama ini banyak dari kita yang beranggapan salah, bahwa dengan menggunakan loadbalance dua jalur koneksi, maka besar bandwidth yang akan kita dapatkan menjadi dua kali lipat dari bandwidth sebelum menggunakan loadbalance (akumulasi dari kedua bandwidth tersebut). Hal ini perlu kita perjelas dahulu, bahwa loadbalance tidak akan menambah besar bandwidth yang kita peroleh, tetapi hanya bertugas untuk membagi trafik dari kedua bandwidth tersebut agar dapat terpakai secara seimbang.

Dengan artikel ini, kita akan membuktikan bahwa dalam penggunaan loadbalancing tidak seperti rumus matematika $512 + 256 = 768$, akan tetapi $512 + 256 = 512 + 256$, atau $512 + 256 = 256 + 256 + 256$.

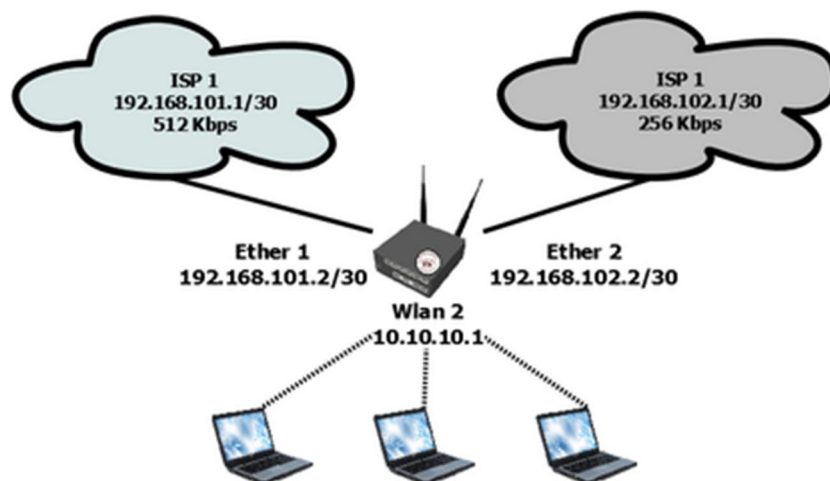
Pada artikel ini kami menggunakan RB433UAH dengan kondisi sebagai berikut :

1. Ether1 dan Ether2 terhubung pada ISP yang berbeda dengan besar bandwidth yang berbeda. ISP1 sebesar 512kbps dan ISP2 sebesar 256kbps.
2. Kita akan menggunakan web-proxy internal dan menggunakan openDNS.
3. Mikrotik RouterOS anda menggunakan versi 4.5 karena fitur PCC mulai dikenal pada versi 3.24.

Jika pada kondisi diatas berbeda dengan kondisi jaringan ditempat anda, maka konfigurasi yang akan kita jabarkan disini harus anda sesuaikan dengan konfigurasi untuk jaringan ditempat anda.

Konfigurasi Dasar

Berikut ini adalah Topologi Jaringan dan IP address yang akan kita gunakan



```
/ip address
```

```
add address=192.168.101.2/30 interface=ether1
```

```
add address=192.168.102.2/30 interface=ether2
```

```
add address=10.10.10.1/24 interface=wlan2
```

```
/ip dns
```

```
set allow-remote-requests=yes primary-dns=208.67.222.222 secondary-dns=208.67.220.220
```

Untuk koneksi client, kita menggunakan koneksi wireless pada wlan2 dengan range IP client 10.10.10.2 s/d 10.10.10.254 netmask 255.255.255.0, dimana IP 10.10.10.1 yang dipasangkan pada wlan2 berfungsi sebagai gateway dan dns server dari client. Jika anda menggunakan DNS dari salah satu isp anda, maka akan ada tambahan mangle yang akan kami berikan tanda tebal

Setelah pengkonfigurasi IP dan DNS sudah benar, kita harus memasang default route ke masing-masing IP gateway ISP kita agar router meneruskan semua trafik yang tidak terhubung padanya ke gateway tersebut. Disini kita menggunakan fitur check-gateway berguna jika salah satu gateway kita putus, maka koneksi akan dibelokkan ke gateway lainnya.

```
/ip route
add dst-address=0.0.0.0/0 gateway=192.168.101.1 distance=1 check-gateway=ping
add dst-address=0.0.0.0/0 gateway=192.168.102.1 distance=2 check-gateway=ping
```

Untuk pengaturan Access Point sehingga PC client dapat terhubung dengan wireless kita, kita menggunakan perintah

```
/interface wireless
set wlan2 mode=ap-bridge band=2.4ghz-b/g ssid=Mikrotik disabled=no
```

Agar pc client dapat melakukan koneksi ke internet, kita juga harus merubah IP privat client ke IP publik yang ada di interface publik kita yaitu ether1 dan ether2.

```
/ip firewall nat
add action=masquerade chain=srcnat out-interface=ether1
add action=masquerade chain=srcnat out-interface=ether2
```

Sampai langkah ini, router dan pc client sudah dapat melakukan koneksi internet. Lakukan ping baik dari router ataupun pc client ke internet. Jika belum berhasil, cek sekali lagi konfigurasi anda.

Webproxy Internal

Pada routerboard tertentu, seperti RB450G, RB433AH, RB433UAH, RB800 dan RB1100 mempunyai expansion slot (USB, MicroSD, CompactFlash) untuk storage tambahan. Pada contoh berikut, kita akan menggunakan usb flashdisk yang dipasangkan pada slot USB. Untuk pertama kali pemasangan, storage tambahan ini akan terbaca statusnya invalid di /system store. Agar dapat digunakan sebagai media penyimpanan cache, maka storage harus diformat dahulu dan diaktifkan Nantinya kita tinggal mengaktifkan webproxy dan set cache-on-disk=yes untuk menggunakan media storage kita. Jangan lupa untuk membelokkan trafik HTTP (tcp port 80) kedalam webproxy kita.

```
/store disk format-drive usb1

/store
add disk=usb1 name=cache-usb type=web-proxy
activate cache-usb
```

```
/ip proxy
set cache-on-disk=yes enabled=yes max-cache-size=200000KiB port=8080

/ip firewall nat
add chain=dstnat protocol=tcp dst-port=80 in-interface=wlan2 action=redirect to-ports=8080
```

Pengaturan Mangle

Pada loadbalancing kali ini kita akan menggunakan fitur yang disebut PCC (Per Connection Classifier). Dengan PCC kita bisa mengelompokkan trafik koneksi yang melalui atau keluar masuk router menjadi beberapa kelompok. Pengelompokan ini bisa dibedakan berdasarkan src-address, dst-address, src-port dan atau dst-port. Router akan mengingat-ingat jalur gateway yang dilewati diawal trafik koneksi, sehingga pada paket-paket selanjutnya yang masih berkaitan dengan koneksi awalnya akan dilewatkan pada jalur gateway yang sama juga. Kelebihan dari PCC ini yang menjawab banyaknya keluhan sering putusnya koneksi pada teknik loadbalancing lainnya sebelum adanya PCC karena perpindahan gateway..

Sebelum membuat mangle loadbalance, untuk mencegah terjadinya loop routing pada trafik, maka semua trafik client yang menuju network yang terhubung langsung dengan router, harus kita bypass dari loadbalancing. Kita bisa membuat daftar IP yang masih dalam satu network router dan memasang mangle pertama kali sebagai berikut

```
/ip firewall address-list
add address=192.168.101.0/30 list=lokal
add address=192.168.102.0/30 list=lokal
add address=10.10.10.0/24 list=lokal

/ip firewall mangle
add action=accept chain=prerouting dst-address-list=lokal in-interface=wlan2 comment="trafik lokal"
add action=accept chain=output dst-address-list=lokal
```

Pada kasus tertentu, trafik pertama bisa berasal dari Internet, seperti penggunaan remote winbox atau telnet dari internet dan sebagainya, oleh karena itu kita juga memerlukan mark-connection untuk menandai trafik tersebut agar trafik baliknya juga bisa melewati interface dimana trafik itu masuk

```
/ip firewall mangle
add action=mark-connection chain=prerouting connection-mark=no-mark in-interface=ether1 new-connection-mark=con-from-isp1 passthrough=yes comment="trafik dari isp1"
add action=mark-connection chain=prerouting connection-mark=no-mark in-interface=ether2 new-connection-mark=con-from-isp2 passthrough=yes comment="trafik dari isp2"
```

Umumnya, sebuah ISP akan membatasi akses DNS servernya dari IP yang hanya dikenalnya, jadi jika anda menggunakan DNS dari salah satu ISP anda, anda harus menambahkan mangle agar trafik DNS tersebut melalui gateway ISP yang bersangkutan bukan melalui gateway ISP lainnya. Disini kami berikan mangle DNS ISP1 yang melalui gateway ISP1. Jika anda menggunakan publik DNS independent, seperti.opendns, anda tidak memerlukan mangle dibawah ini.

```
/ip firewall mangle
add action=mark-connection chain=output comment=dns dst-address=202.65.112.21 dst-port=53
```

```
new-connection-mark=dns passthrough=yes protocol=tcp comment="trafik DNS citra.net.id"
add action=mark-connection chain=output dst-address=202.65.112.21 dst-port=53 new-connection-
mark=dns passthrough=yes protocol=udp
add action=mark-routing chain=output connection-mark=dns new-routing-mark=route-to-isp1
passthrough=no
```

Karena kita menggunakan webproxy pada router, maka trafik yang perlu kita loadbalance ada 2 jenis. Yang pertama adalah trafik dari client menuju internet (non HTTP), dan trafik dari webproxy menuju internet. Agar lebih terstruktur dan mudah dalam pembacaannya, kita akan menggunakan custom-chain sebagai berikut :

```
/ip firewall mangle
add action=jump chain=prerouting comment="lompat ke client-lb" connection-mark=no-mark in-
interface=wlan2 jump-target=client-lb
add action=jump chain=output comment="lompat ke lb-proxy" connection-mark=no-mark out-
interface=!wlan2 jump-target=lb-proxy
```

Pada mangle diatas, untuk trafik loadbalance client pastikan parameter in-interface adalah interface yang terhubung dengan client, dan untuk trafik loadbalance webproxy, kita menggunakan chain output dengan parameter out-interface yang bukan terhubung ke interface client. Setelah custom chain untuk loadbalancing dibuat, kita bisa membuat mangle di custom chain tersebut sebagai berikut

```
/ip firewall mangle
add action=mark-connection chain=client-lb dst-address-type=!local new-connection-mark=to-isp1
passthrough=yes per-connection-classifier=both-addresses: 3/0 comment="awal loadbalancing klien"
add action=mark-connection chain=client-lb dst-address-type=!local new-connection-mark=to-isp1
passthrough=yes per-connection-classifier=both-addresses: 3/1
add action=mark-connection chain=client-lb dst-address-type=!local new-connection-mark=to-isp2
passthrough=yes per-connection-classifier=both-addresses: 3/2
add action=return chain=client-lb comment="akhir dari loadbalancing"
```

```
/ip firewall mangle
add action=mark-connection chain=lb-proxy dst-address-type=!local new-connection-mark=con-from-
isp1 passthrough=yes per-connection-classifier=both-addresses: 3/0 comment="awal load balancing
proxy"
add action=mark-connection chain=lb-proxy dst-address-type=!local new-connection-mark=con-from-
isp1 passthrough=yes per-connection-classifier=both-addresses: 3/1
add action=mark-connection chain=lb-proxy dst-address-type=!local new-connection-mark=con-from-
isp2 passthrough=yes per-connection-classifier=both-addresses: 3/2
add action=return chain=lb-proxy comment="akhir dari loadbalancing"
```

Untuk contoh diatas, pada loadbalancing client dan webproxy menggunakan parameter pemisahan trafik pcc yang sama, yaitu both-address, sehingga router akan mengingat-ingat berdasarkan src-address dan dst-address dari sebuah koneksi. Karena trafik ISP kita yang berbeda (512kbps dan 256kbps), kita membagi beban trafiknya menjadi 3 bagian. 2 bagian pertama akan melewati gateway ISP1, dan 1 bagian terakhir akan melewati gateway ISP2. Jika masing-masing trafik dari client dan proxy sudah ditandai, langkah berikutnya kita tinggal membuat mangle mark-route yang akan digunakan dalam proses routing nantinya

```
/ip firewall mangle
```

```

add action=jump chain=prerouting comment="marking route client" connection-mark=!no-mark in-
interface=wlan2 jump-target=route-client
add action=mark-routing chain=route-client connection-mark=to-isp1 new-routing-mark=route-to-isp1
passthrough=no
add action=mark-routing chain=route-client connection-mark=to-isp2 new-routing-mark=route-to-isp2
passthrough=no
add action=mark-routing chain=route-client connection-mark=con-from-isp1 new-routing-mark=route-
to-isp1 passthrough=no
add action=mark-routing chain=route-client connection-mark=con-from-isp2 new-routing-mark=route-
to-isp2 passthrough=no
add action=return chain=route-client disabled=no

/ip firewall mangle
add action=mark-routing chain=output comment="marking route proxy" connection-mark=con-from-
isp1 new-routing-mark=route-to-isp1 out-interface=!wlan2 passthrough=no
add action=mark-routing chain=output connection-mark=con-from-isp2 new-routing-mark=route-to-
isp2 out-interface=!wlan2 passthrough=no

```

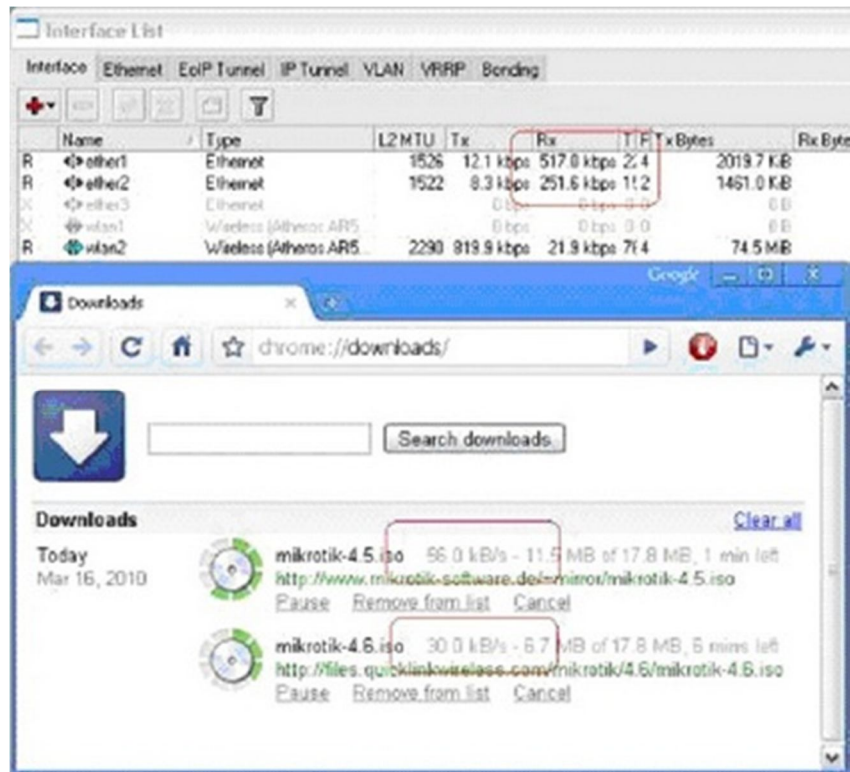
Pengaturan Routing

Pengaturan mangle diatas tidak akan berguna jika anda belum membuat routing berdasar mark-route yang sudah kita buat. Disini kita juga akan membuat routing backup, sehingga apabila sebuah gateway terputus, maka semua koneksi akan melewati gateway yang masing terhubung

```

/ip route
add check-gateway=ping dst-address=0.0.0.0/0 gateway=192.168.101.1 routing-mark=route-to-isp1
distance=1
add check-gateway=ping dst-address=0.0.0.0/0 gateway=192.168.102.1 routing-mark=route-to-isp1
distance=2
add check-gateway=ping dst-address=0.0.0.0/0 gateway=192.168.102.1 routing-mark=route-to-isp2
distance=1
add check-gateway=ping dst-address=0.0.0.0/0 gateway=192.168.101.1 routing-mark=route-to-isp2
distance=2

```



Dari gambar terlihat, bahwa hanya dengan melakukan 1 file download (1 koneksi), kita hanya mendapatkan speed 56kBps (448kbps) karena pada saat itu melewati gateway ISP1, sedangkan jika kita mendownload file (membuka koneksi baru) lagi pada web lain, akan mendapatkan 30kBps (240kbps). Dari pengujian ini terlihat dapat disimpulkan bahwa

$$512\text{kbps} + 256\text{kbps} \neq 768\text{kbps}$$

Catatan :

- * Loadbalancing menggunakan teknik pcc ini akan berjalan efektif dan mendekati seimbang jika semakin banyak koneksi (dari client) yang terjadi.
- * Gunakan ISP yang memiliki bandwidth FIX bukan Share untuk mendapatkan hasil yang lebih optimal.
- * Load Balance menggunakan PCC ini bukan selamanya dan sepenuhnya sebuah solusi yang pasti berhasil baik di semua jenis network, karena proses penyeimbangan dari traffic adalah berdasarkan logika probabilitas.