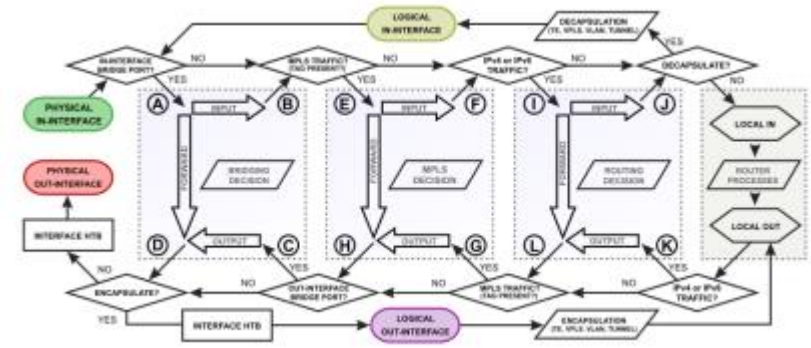
Penggunaan Custom Chain pada Firewall MikroTik

Pada RouterOS MikroTik terdapat sebuah fitur yang disebut dengan '**Firewall**'. Fitur ini biasanya banyak digunakan untuk melakukan filtering akses ***(Filter Rule)***, *Forwarding (***NAT***)*, dan juga untuk menandai koneksi maupun paket dari trafik data yang melewati router (***Mangle***). Supaya fungsi dari fitur firewall ini dapat berjalan dengan baik, kita harus menambahkan rule-rule yang sesuai. Terdapat sebuah parameter utama pada rule di fitur firewall ini yaitu '**Chain**'. Parameter ini memiliki kegunaan untuk menetukan jenis trafik yang akan di-manage pada fitur firewall dan setiap fungsi pada firewall seperti **Filter Rule**, **NAT**, **Mangle** memiliki opsi chain yang berbeda.

Pengisian parameter chain pada dasarnya mengacu pada skema '***Traffic Flow***' dari Router. Jadi kita harus mengenali terlebih dahulu jenis trafik yang akan kita *manage* menggunakan firewall. chain bisa dianaloginkan sebagai tempat admin mencegat sebuah trafik, kemudian melakukan firewalling sesuai kebutuhan.



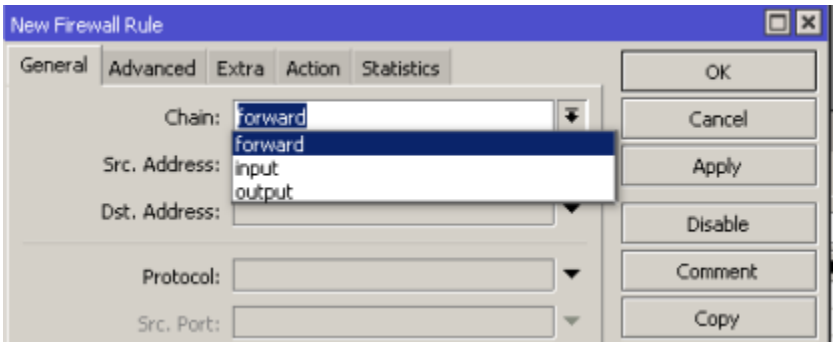
FILTER RULES

Filter rule biasanya digunakan untuk melakukan kebijakan boleh atau tidaknya sebuah trafik ada dalam jaringan, identik dengan accept atau drop. Pada menu **Firewall → Filter Rules** terdapat 3 macam chain yang tersedia. Chain tersebut antara lain adalah ***Forward***, ***Input***, ***Output***. Adapun fungsi dari masing-masing chain tersebut adalah sebagai berikut:

- ***Forward*** :
Digunakan untuk memproses trafik paket data yang hanya melewati router. Misalnya trafik dari jaringan public ke local atau sebaliknya dari jaringan local ke public, contoh kasus seperti pada saat kita melakukan browsing. Trafik laptop browsing ke internet dapat dimanage oleh firewall dengan menggunakan chain forward.

- ***Input*** :
Digunakan untuk memproses trafik paket data yang masuk ke dalam router melalui interface yang ada di router dan memiliki tujuan IP Address berupa ip yang terdapat pada router. Jenis trafik ini bisa berasal dari jaringan public maupun dari jaringan lokal dengan tujuan router itu sendiri. Contoh: Mengakses router menggunakan winbox, webfig, telnet baik dari Public maupun Local.

- ***Output*** :
Digunakan untuk memproses trafik paket data yang keluar dari router. Dengan kata lain merupakan kebalikan dari 'Input'. Jadi trafik yang berasal dari dalam router itu sendiri dengan tujuan jaringan Public maupun jaringan Local.Misal dari new terminal winbox, kita ping ke ip google. Maka trafik ini bisa ditangkap dichain output.

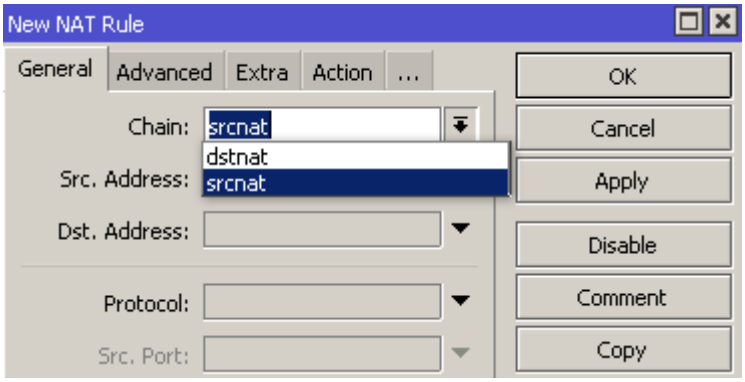


NAT (Network Address Translation)

Pada menu **Firewall** → **NAT** terdapat 2 macam opsi chain yang tersedia, yaitu ***dst-nat*** dan ***src-nat***. Dan fungsi dari NAT sendiri adalah untuk melakukan pengubahan *Source Address* maupun *Destination Address*. Kemudian fungsi dari masing-masing chain tersebut adalah sebagai berikut:

- ***dstnat*** :
Memiliki fungsi untuk mengubah destination address pada sebuah paket data. Biasa digunakan untuk membuat host dalam jaringan lokal dapat diakses dari luar jaringan (internet) dengan cara NAT akan mengganti alamat IP tujuan paket dengan alamat IP lokal. Jadi kesimpulan fungsi dari chain ini adalah untuk mengubah/mengganti IP Address tujuan pada sebuah paket data.

-***srcnat*** :
Memiliki fungsi untuk mengubah source address dari sebuah paket data. Sebagai contoh kasus fungsi dari chain ini banyak digunakan ketika kita melakukan akses website dari jaringan LAN. Secara aturan untuk IP Address local tidak diperbolehkan untuk masuk ke jaringan WAN, maka diperlukan konfigurasi 'srcnat' ini. Sehingga IP Address lokal akan disembunyikan dan diganti dengan IP Address public yang terpasang pada router.



MANGLE

Pada menu **Firewall → Mangle** terdapat 4 macam pilihan untuk chain, yaitu ***Forward***, ***Input***, ***Output***, ***Prerouting***, dan ***Postrouting***. Mangle sendiri memiliki fungsi untuk menandai sebuah koneksi atau paket data, yang melewati route, masuk ke router, ataupun yang keluar dari router. Pada implementasinya Mangle sering dikombinasikan dengan fitur lain seperti *Management Bandwith*, *Routing policy*, dll. Adapun fungsi dari masing-masing chain yang ada pada mangle adalah sebagai berikut:

- ***Forward, Input, Output*** :

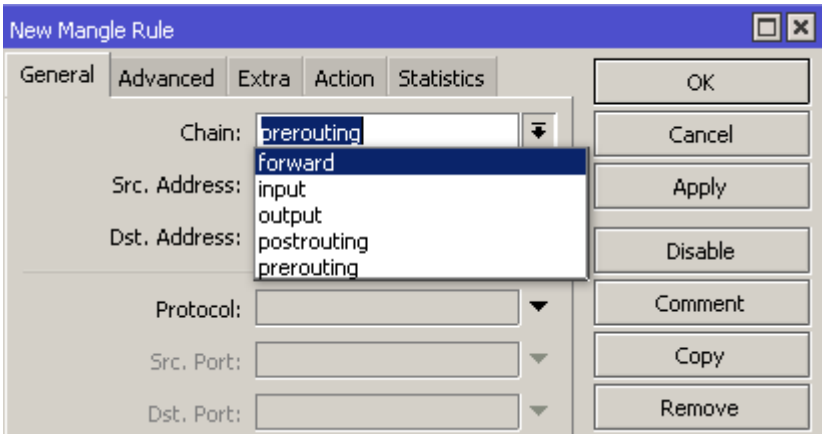
Untuk penjelasan mengenai Forward, Input, dan Output sebenarnya tidak jauh berbeda dengan apa yang telah diuraikan pada Filter rules diatas. Namun pada Mangle, semua jenis trafik paket data forward, input, dan output bisa ditandai berdasarkan koneksi atau paket atau paket data.

- ***Prerouting*** :

Merupakan sebuah koneksi yang akan masuk kedalam router dan melewati router. Berbeda dengan input yang mana hanya akan menangkap trafik yang masuk ke router. Trafik yang melewat router dan trafik yang masuk kedalam router dapat ditangkap di chain prerouting.

- ***Postrouting*** :

Kebalikan dari prerouting, postrouting merupakan koneksi yang akan keluar dari router, baik untuk trafik yang melewati router ataupun yang keluar dari router.



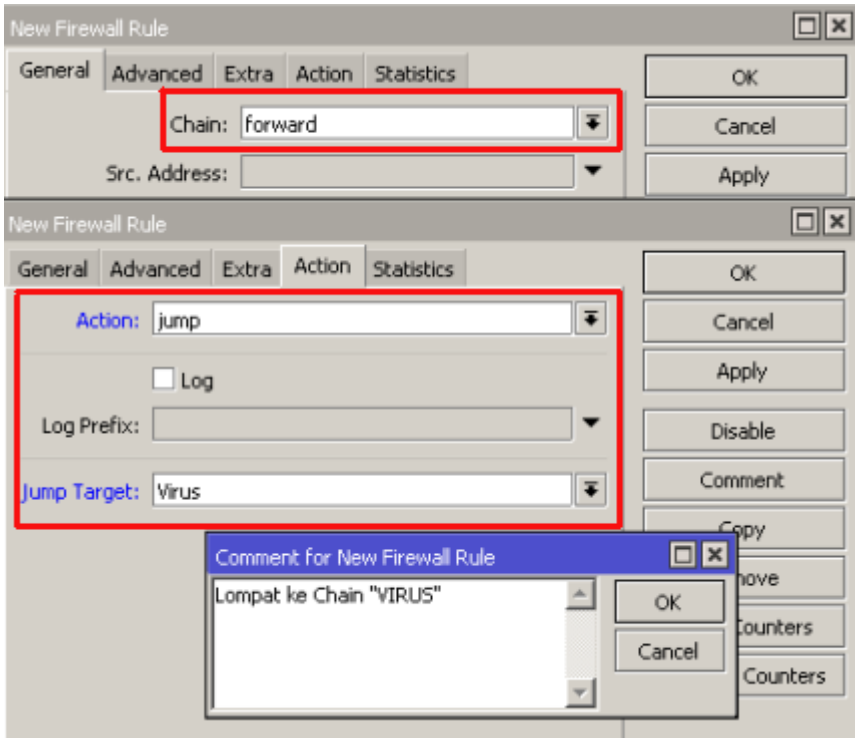
CUSTOM CHAIN

Nah, selain jenis chain yang telah diuraikan diatas, sebenarnya ada jenis chain yang lain dimana kita bisa menambahkan atau menentukan sendiri nama dari chain tersebut selain dari forward, input, output dll. Nama chain tersebut dapat kita tentukan sendiri, namun pada prinsipnya tetap mnegacu pada chain utama yang tersedia di Firewall. Biasanya custom chain digunakan untuk menghemat resource router dan mempermudah admin jaringan dalam membaca rule firewall. By default router akan membaca rule firewall secara berurutan sesuai nomor urut rule firewall. Namun dengan fitur jump ini, admin jaringan dapat menentukan pembacaan rule firewall yang lebih efisien.

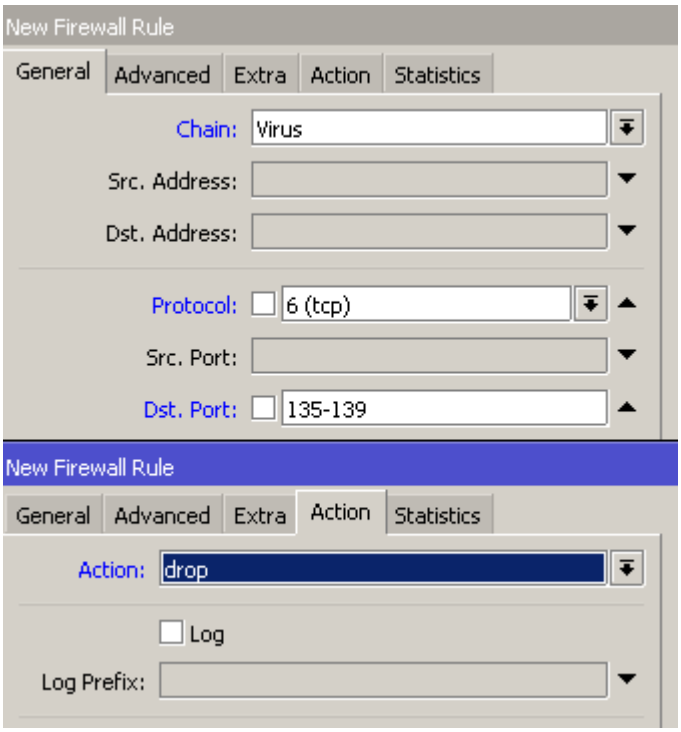
Untuk membuat *custom chain* tersebut kita memerlukan sebuah '**Action**' yaitu **Jump**. Jump sendiri berfungsi untuk melompat ke chain lain yang telah didefiniskan pada paramater ***jump-target***. Sehingga kita bisa menempatkan rule dari *custom chain* yang telah kita buat pada urutan paling bawah. Ini dimaksudkan untuk mempermudah dalam pengelolaan rule-rule firewall, terlebih lagi jika kita memeiliki rule-rule yang banyak. Adapun langkah-langkah pembuatan ***Custom Chain*** adalah sebagai berikut.

Pada contoh kasus kali ini kita akan membuat sebuah rule yang mana akan melindungi perangkat client dari trafik yang mengandung virus. Untuk itu agar lebih mudah dalam pengelolaan kita akan membuat sebuah chain baru yang bernama **Virus** dengan jenis trafik **Forward**.

Pertama, pilih menu **Firewall** → **Filter** Rules. Kemudian isikan parameter sesuai dengan tampilan gambar dibawah ini.



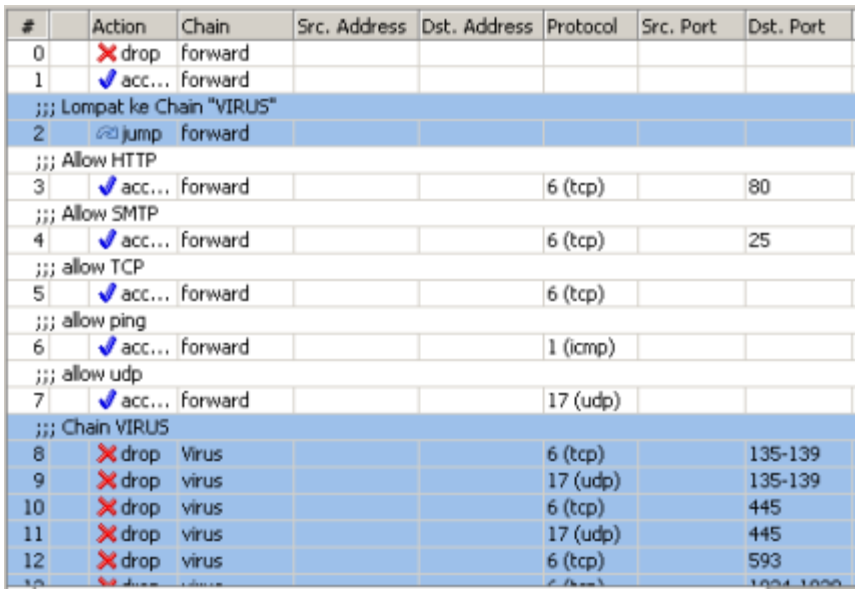
Selanjutnya kita akan menentukan rule untuk custom chain virus yang sudah dibuat. Di rule ini kita berikan tambahan parameter yang lebih spesifik.



Untuk lebih lengkap tentang rule dengan protokol dan juga port yang digunakan oleh virus bisa diisikan dengan script berikut.

```
/ip firewall filter
add chain=virus protocol=udp dst-port=135-139 action=drop
add chain=virus protocol=tcp dst-port=445 action=drop
add chain=virus protocol=udp dst-port=445 action=drop
add chain=virus protocol=tcp dst-port=593 action=drop
add chain=virus protocol=tcp dst-port=1024-1030 action=drop
add chain=virus protocol=tcp dst-port=1080 action=drop
add chain=virus protocol=tcp dst-port=1214 action=drop
add chain=virus protocol=tcp dst-port=1363 action=drop
add chain=virus protocol=tcp dst-port=1364 action=drop
add chain=virus protocol=tcp dst-port=1368 action=drop
add chain=virus protocol=tcp dst-port=1373 action=drop
add chain=virus protocol=tcp dst-port=1377 action=drop
add chain=virus protocol=tcp dst-port=1433-1434 action=drop
add chain=virus protocol=tcp dst-port=2745 action=drop
add chain=virus protocol=tcp dst-port=2283 action=drop
add chain=virus protocol=tcp dst-port=2535 action=drop
add chain=virus protocol=tcp dst-port=2745 action=drop
add chain=virus protocol=tcp dst-port=3127-3128 action=drop
add chain=virus protocol=tcp dst-port=3410 action=drop
add chain=virus protocol=tcp dst-port=4444 action=drop
add chain=virus protocol=udp dst-port=4444 action=drop
add chain=virus protocol=tcp dst-port=5554 action=drop
add chain=virus protocol=tcp dst-port=8866 action=drop
add chain=virus protocol=tcp dst-port=9898 action=drop
add chain=virus protocol=tcp dst-port=10000 action=drop
add chain=virus protocol=tcp dst-port=10080 action=drop
add chain=virus protocol=tcp dst-port=12345 action=drop
add chain=virus protocol=tcp dst-port=17300 action=drop
add chain=virus protocol=tcp dst-port=27374 action=drop
add chain=virus protocol=tcp dst-port=65506 action=drop
```

Kemudian apabila kita telah selesai mengisikan parameter-parameter diatas maka akan tampil pada list firewall filter sebagai berikut.



#		Action	Chain	Src. Address	Dst. Address	Protocol	Src. Port	Dst. Port
0		drop	forward					
1		acc...	forward					
;;; Lompat ke Chain "VIRUS"								
2		jump	forward					
;;; Allow HTTP								
3		acc...	forward			6 (tcp)		80
;;; Allow SMTP								
4		acc...	forward			6 (tcp)		25
;;; allow TCP								
5		acc...	forward			6 (tcp)		
;;; allow ping								
6		acc...	forward			1 (icmp)		
;;; allow udp								
7		acc...	forward			17 (udp)		
;;; Chain VIRUS								
8		drop	Virus			6 (tcp)		135-139
9		drop	virus			17 (udp)		135-139
10		drop	virus			6 (tcp)		445
11		drop	virus			17 (udp)		445
12		drop	virus			6 (tcp)		593

Jika kita melihat list tersebut, letak rule jump berada pada urutan ***nomer 2*** sedangkan chain virus berada pada urutan paling bawah. Nah, ketika ada trafik paket data akan yang melewati router (forward) maka akan diperiksa dengan rule-rule firewall filter. Dan saat proses pemeriksaan sampai di rule nomer 2, maka akan di-jump (lompat) ke chain virus diurutan nomor 8. Apabila paket data tersebut mengandung virus dengan protokol dan port yang telah didefinisikan pada rule chain virus maka paket data tersebut akan di drop. Namun, apabila tidak mengandung virus, maka pemeriksaan akan dikembalikan ke atas dengan melanjutkan pemeriksaan di rule berikutnya.