Mangle

Document revision .NaN (February 11, 2008, 4:14 GMT)
This document applies to V3.0

Table of Contents

General Information

Summary

The mangle facility allows to mark IP packets with special marks. These marks are used by various other router facilities to identify the packets. Additionaly, the mangle facility is used to modify some fields in the IP header, like TOS (DSCP) and TTL fields.

Specifications

Packages required: ***system***
License required: ***level1***
Home menu level: ***/ip firewall mangle***
Standards and Technologies: ***IP***
Hardware usage: *Increases with count of mangle rules*

Mangle

Home menu level: ***/ip firewall mangle***

Description

Mangle is a kind of 'marker' that marks packets for future processing with special marks. Many other facilities in RouterOS make use of these marks, e.g. queue trees and NAT. They identify a packet based on its mark and process it accordingly. The mangle marks exist only within the router, they are not transmitted across the network.

Property Description

action (*accept* | *add-dst-to-address-list* | *add-src-to-address-list* | *change-dscp* | *change-mss* | *change-ttl* | *jump* | *log* | *mark-connection* | *mark-packet* | *mark-routing* | *passthrough* | *return* | *set-priority* | *strip-ipv4-options*; default: **accept**) - action to undertake if the packet matches the rule

- **accept** - accept the packet. No action, i.e., the packet is passed through and no more rules are applied to it
- **add-dst-to-address-list** - add destination address of an IP packet to the address list specified by address-list parameter
- **add-src-to-address-list** - add source address of an IP packet to the address list specified by address-list parameter
- **change-dscp** - change Differentiated Services Code Point (DSCP) field value specified by the new-dscp parameter
- **change-mss** - change Maximum Segment Size field value of the packet to a value specified by the new-mss parameter
- **change-ttl** - change Time to Live field value of the packet to a value specified by the new-ttl parameter
- **jump** - jump to the chain specified by the value of the jump-target parameter
- **log** - each match with this action will add a message to the system log
- **mark-connection** - place a mark specified by the new-connection-mark parameter on the entire connection that matches the rule
- **mark-packet** - place a mark specified by the new-packet-mark parameter on a packet that matches the rule
- **mark-routing** - place a mark specified by the new-routing-mark parameter on a packet. This kind of marks is used for policy routing purposes only
- **passthrough** - ignore this rule go on to the next one
- **return** - pass control back to the chain from where the jump took place
- **set-priority** - set priority speciefied by the new-priority parameter on the packets sent out through a link that is capable of transporting priority (VLAN or WMM-enabled wireless interface)
- **strip-ipv4-options** - strip IPv4 option fields from the IP packet

address-list (*name*) - specify the name of the address list to collect IP addresses from rules having action=add-dst-to-address-list or action=add-src-to-address-list actions. These address lists could be later used for packet matching

address-list-timeout (*time*; default: **00:00:00**) - time interval after which the address will be removed from the address list specified by address-list parameter. Used in conjunction with add-dst-to-address-list or add-src-to-address-list actions

- **00:00:00** - leave the address in the address list forever

chain (*forward* | *input* | *output* | *postrouting* | *prerouting*) - specify the chain to put a particular rule into. As the different traffic is passed through different chains, always be careful in choosing the right chain for a new rule. If the input does not match the name of an already defined chain, a new chain will be created

comment (*text*) - free form textual comment for the rule. A comment can be used to refer the particular rule from scripts

connection-bytes (*integerinteger*) - match packets only if a given amount of bytes has been

transfered through the particular connection

- **0** - means infinity, exempli gratia: connection-bytes=2000000-0 means that the rule matches if more than 2MB has been transfered through the relevant connection

connection-limit (*integernetmask*) - restrict connection limit per address or address block

connection-mark (*name*) - match packets marked via mangle facility with particular connection mark

connection-state (*estabilished* | *invalid* | *new* | *related*) - interprets the connection tracking analysis data for a particular packet

- **estabilished** - a packet which belongs to an existing connection, exempli gratia a reply packet or a packet which belongs to already replied connection
- **invalid** - a packet which could not be identified for some reason. This includes out of memory condition and ICMP errors which do not correspond to any known connection. It is generally advised to drop these packets
- **new** - a packet which begins a new TCP connection
- **related** - a packet which is related to, but not part of an existing connection, such as ICMP errors or a packet which begins FTP data connection (the later requires enabled FTP connection tracking helper under /ip firewall service-port)

connection-type (*ftp* | *gre* | *h323* | *irc* | *mms* | *pptp* | *quake3* | *tftp*) - match packets from related connections based on information from their connection tracking helpers. A relevant connection helper must be enabled under /ip firewall service-port

content (*text*) - the text packets should contain in order to match the rule

dscp (*integer*: 0..63) - DSCP (ex-ToS) IP header field value

dst-address (*IP addressnetmaskIP addressIP address*) - specify the address range an IP packet is destined to. Note that console converts entered address/netmask value to a valid network address, i.e.:1.1.1.1/24 is converted to 1.1.1.0/24

dst-address-list (*name*) - match destination address of a packet against user-defined address list

dst-address-type (*unicast* | *local* | *broadcast* | *multicast*) - match destination address type of the IP packet, one of the:

- **unicast** - IP addresses used for one point to another point transmission. There is only one sender and one receiver in this case
- **local** - match addresses assigned to router's interfaces
- **broadcast** - the IP packet is sent from one point to all other points in the IP subnetwork
- **multicast** - this type of IP addressing is responsible for transmission from one or more points to a set of other points

dst-limit (*integertimeintegerdst-address* | *dst-port* | *src-addresstime*) - limit the packet per second (pps) rate on a per destination IP or per destination port base. As opposed to the limit match, every destination IP address / destination port has it's own limit. The options are as follows (in order of appearance):

- **count** - maximum average packet rate, measured in packets per second (pps), unless followed by time option
- **time** - specifies the time interval over which the packet rate is measured
- **burst** - number of packets to match in a burst
- **mode** - the classifier(-s) for packet rate limiting

- **expire** - specifies interval after which recorded IP addresses / ports will be deleted

dst-port (*integer*: 0..65535*integer*: 0..65535) - destination port number or range

fragment (yes | no) - whether the packet is a fragment of an IP packet. Starting packet (i.e., first fragment) does not count. Note that is the connection tracking is enabled, there will be no fragments as the system automatically assembles every packet

hotspot (*multiple choice: auth* | *from-client* | *http* | *local-dst* | *to-client*) - matches packets received from clients against various HotSpot conditions. All values can be negated

- **auth** - true, if a packet comes from an authenticted HotSpotclient
- **from-client** - true, if a packet comes from any HotSpot client
- **http** - true, if a HotSpot client sends a packet to the address and port previously detected as his proxy server (Universal Proxy technique) or if the destination port is 80 and transparent proxying is enabled for that particular client
- **local-dst** - true, if a packet has local destination IP address
- **to-client** - true, if a packet is sent to a client

icmp-options (*integerinteger*) - match ICMP Type:Code fields

in-bridge-port (*name*) - actual interface the packet has entered the router through (if bridged, this property matches the actual bridge port, while in-interface - the bridge itself)

in-interface (*name*) - interface the packet has entered the router through (if the interface is bridged, then the packet will appear to come from the bridge interface itself)

ingress-priority (*integer*: 0..63) - INGRESS (received) priority of the packet, if set (0 otherwise). The priority may be derived from either VLAN or WMM priority

ipv4-options (*any* | *loose-source-routing* | *no-record-route* | *no-router-alert* | *no-source-routing* | *no-timestamp* | *none* | *record-route* | *router-alert* | *strict-source-routing* | *timestamp*) - match ipv4 header options

- **any** - match packet with at least one of the ipv4 options
- **loose-source-routing** - match packets with loose source routing option. This option is used to route the internet datagram based on information supplied by the source
- **no-record-route** - match packets with no record route option. This option is used to route the internet datagram based on information supplied by the source
- **no-router-alert** - match packets with no router alter option
- **no-source-routing** - match packets with no source routing option
- **no-timestamp** - match packets with no timestamp option
- **record-route** - match packets with record route option
- **router-alert** - match packets with router alter option
- **strict-source-routing** - match packets with strict source routing option
- **timestamp** - match packets with timestamp

jump-target (*forward* | *input* | *output* | *postrouting* | *preroutingname*) - name of the target chain to jump to, if the action=jump is used

layer7-protocol (*name*) - Layer 7 filter name as set in the /ip firewall layer7-protocol menu. Caution: this matcher needs high computational power

limit (*integertimeinteger*) - restrict packet match rate to a given limit. Usefull to reduce the amount of log messages

- **count** - maximum average packet rate, measured in packets per second (pps), unless followed by time option
- **time** - specify the time interval over which the packet rate is measured
- **burst** - number of packets to match in a burst

log-prefix (*text*) - all messages written to logs will contain the prefix specified herein. Used in conjunction with action=log

new-connection-mark (*name*) - specify the new value of the connection mark to be used in conjunction with action=mark-connection

new-dscp (*integer*: 0..63) - specify the new value of the DSCP field to be used in conjunction with action=change-dscp

new-mss (*integer*) - specify MSS value to be used in conjunction with action=change-mss

new-packet-mark (*name*) - specify the new value of the packet mark to be used in conjunction with action=mark-packet

new-priority (*integer*) - specify the new value of packet priority for the priority-enabled interfaces, used in conjunction with action=set-priority

- **from-dscp** - set packet priority form its DSCP field value
- **from-ingress** - set packet priority from the INGRESS priority of the packet (in case packet has been received from an interface that supports priorities - VLAN or WMM-enabled wireless interface; 0 if not set)

new-routing-mark (*name*) - specify the new value of the routing mark used in conjunction with action=mark-routing

new-ttl (*decrement* | *increment* | *setinteger*) - specify the new TTL field value used in conjunction with action=change-ttl

- **decrement** - the value of the TTL field will be decremented for value
- **increment** - the value of the TTL field will be incremented for value
- **set:** - the value of the TTL field will be set to value

nth (*integerinteger*: 0..15*integer*) - match a particular Nth packet received by the rule. One of 16 available counters can be used to count packets

- **every** - match every every+1th packet. For example, if every=1 then the rule matches every 2nd packet
- **counter** - specifies which counter to use. A counter increments each time the rule containing nth match matches
- **packet** - match on the given packet number. The value by obvious reasons must be between 0 and every. If this option is used for a given counter, then there must be at least every+1 rules with this option, covering all values between 0 and every inclusively.

out-bridge-port (*name*) - actual interface the packet is leaving the router through (if bridged, this property matches the actual bridge port, while out-interface - the bridge itself)

out-interface (*name*) - interface the packet is leaving the router through (if the interface is bridged, then the packet will appear to leave through the bridge interface itself)

p2p (*all-p2p* | *bit-torrent* | *direct-connect* | *edonkey* | *fasttrack* | *gnutella* | *soulseek* | *warez* | *winmx*) - match packets belonging to connections of the above P2P protocols

packet-mark (*name*) - match the packets marked in mangle with specific packet mark

packet-size (*integer*: 0..65535*integer*: 0..65535) - matches packet of the specified size or size range in bytes

- **min** - specifies lower boundary of the size range or a standalone value
- **max** - specifies upper boundary of the size range

passthrough (yes | no; default: **yes**) - whether to let the packet to pass further (like action passthrough) after marking it with a given mark (property only valid if action is mark packet, connection or routing mark)

port (*port*) - matches if any (source or destination) port matches the specified list of ports or port ranges (note that the protocol must still be selected, just like for the regular src-port and dst-port matchers)

protocol (*ddp* | *egp* | *encap* | *ggp* | *gre* | *hmp* | *icmp* | *idrp-cmtp* | *igmp* | *ipencap* | *ipip* | *ipsec-ah* | *ipsec-esp* | *iso-tp4* | *ospf* | *pup* | *rdp* | *rspf* | *st* | *tcp* | *udp* | *vmtp* | *xns-idp* | *xtpinteger*) - matches particular IP protocol specified by protocol name or number. You should specify this setting if you want to specify ports

psd (*integertimeintegerinteger*) - attempts to detect TCP and UDP scans. It is advised to assign lower weight to ports with high numbers to reduce the frequency of false positives, such as from passive mode FTP transfers

- **WeightThreshold** - total weight of the latest TCP/UDP packets with different destination ports coming from the same host to be treated as port scan sequence
- **DelayThreshold** - delay for the packets with different destination ports coming from the same host to be treated as possible port scan subsequence
- **LowPortWeight** - weight of the packets with privileged (<=1024) destination port
- **HighPortWeight** - weight of the packet with non-priviliged destination port

random (*integer*: 1..99) - matches packets randomly with given propability

routing-mark (*name*) - matches packets marked with the specified routing mark

src-address (*IP addressnetmaskIP addressIP address*) - specifies the address range an IP packet is originated from. Note that console converts entered address/netmask value to a valid network address, i.e.:1.1.1.1/24 is converted to 1.1.1.0/24

src-address-list (*name*) - matches source address of a packet against user-defined address list

src-address-type (*unicast* | *local* | *broadcast* | *multicast*) - matches source address type of the IP packet, one of the:

- **unicast** - IP addresses used for one point to another point transmission. There is only one sender and one receiver in this case
- **local** - matches addresses assigned to router's interfaces
- **broadcast** - the IP packet is sent from one point to all other points in the IP subnetwork
- **multicast** - this type of IP addressing is responsible for transmission from one or more points to a set of other points

src-mac-address (*MAC address*) - source MAC address

src-port (*integer*: 0..65535*integer*: 0..65535) - source port number or range

tcp-flags (*multiple choice: ack* | *cwr* | *ece* | *fin* | *psh* | *rst* | *syn* | *urg*) - tcp flags to match

- **ack** - acknowledging data
- **cwr** - congestion window reduced

- **ece** - ECN-echo flag (explicit congestion notification)
- **fin** - close connection
- **psh** - push function
- **rst** - drop connection
- **syn** - new connection
- **urg** - urgent data

tcp-mss (*integer*: 0..65535) - matches TCP MSS value of an IP packet

time (*timetimesat* | *fri* | *thu* | *wed* | *tue* | *mon* | *sun*) - allows to create filter based on the packets' arrival time and date or, for locally generated packets, departure time and date

Notes

Instead of making two rules if you want to mark a packet, connection or routing-mark and finish mangle table processing on that event (in other words, mark and simultaneously accept the packet), you may disable the set by default **passthrough** property of the marking rule.

Usually routing-mark is not used for P2P, since P2P traffic always is routed over a default getaway.

Application Examples

Description

The following section discusses some examples of using the mangle facility.

Peer-to-Peer Traffic Marking

To ensure the quality of service for network connection, interactive traffic types such as VoIP and HTTP should be prioritized over non-interactive, such as peer-to-peer network traffic. RouterOS QOS implementation uses mangle to mark different types of traffic first, and then place them into queues with different limits.

The following example enforces the P2P traffic will get no more than 1Mbps of the total link capacity when the link is heavily used by other traffic otherwice expanding to the full link capacity:

```
[admin@MikroTik] > /ip firewall mangle add chain=forward \
\... p2p=all-p2p action=mark-connection new-connection-mark=p2p_conn
[admin@MikroTik] > /ip firewall mangle add chain=forward \
\... connection-mark=p2p_conn action=mark-packet new-packet-mark=p2p
[admin@MikroTik] > /ip firewall mangle add chain=forward \
\... connection-mark=!p2p_conn action=mark-packet new-packet-mark=other
[admin@MikroTik] > /ip firewall mangle print
Flags: X - disabled, I - invalid, D - dynamic
 0   chain=forward p2p=all-p2p action=mark-connection new-connection-mark=p2p_conn

 1   chain=forward connection-mark=p2p_conn action=mark-packet new-packet-mark=p2p

 2   chain=forward packet-mark=!p2p_conn action=mark-packet new-packet-mark=other
[admin@MikroTik] >
[admin@MikroTik] > /queue tree add parent=Public packet-mark=p2p limit-at=1000000 \
\... max-limit=100000000 priority=8
[admin@MikroTik] > /queue tree add parent=Local packet-mark=p2p limit-at=1000000 \
\... max-limit=100000000 priority=8
[admin@MikroTik] > /queue tree add parent=Public packet-mark=other  limit-at=1000000 \
```

```
\... max-limit=100000000 priority=1
[admin@MikroTik] > /queue tree add parent=Local packet-mark=other  limit-at=1000000 \
\... max-limit=100000000 priority=1
```

Mark by MAC address

To mark traffic from a known MAC address which goes to the router or through it, do the following:

```
[admin@MikroTik] > / ip firewall mangle add chain=prerouting \
\... src-mac-address=00:01:29:60:36:E7 action=mark-connection
new-connection-mark=known_mac_conn
[admin@MikroTik] > / ip firewall mangle add chain=prerouting \
\... connection-mark=known_mac_conn action=mark-packet new-packet-mark=known_mac
```

Change MSS

It is a well known fact that VPN links have smaller packet size due to incapsulation overhead. A large packet with MSS that exceeds the MSS of the VPN link should be fragmented prior to sending it via that kind of connection. However, if the packet has DF flag set, it cannot be fragmented and should be discarded. On links that have broken path MTU discovery (PMTUD) it may lead to a number of problems, including problems with FTP and HTTP data transfer and e-mail services.

In case of link with broken PMTUD, a decrease of the MSS of the packets coming through the VPN link solves the problem. The following example demonstrates how to decrease the MSS value via mangle:

```
[admin@MikroTik] > /ip firewall mangle add out-interface=pppoe-out \
\... protocol=tcp tcp-flags=syn action=change-mss new-mss=1300 chain=forward
[admin@MikroTik] > /ip firewall mangle print
Flags: X - disabled, I - invalid, D - dynamic
 0   chain=forward out-interface=pppoe-out protocol=tcp tcp-flags=syn
     action=change-mss new-mss=1300

[admin@MikroTik] >
```