Common Firewall Matchers and Actions

Common Actions and Associated properties

Property	Description
action (*action name*; Default: **accept**)	Action to take if a packet is matched by the rule: • accept - accept the packet. A packet is not passed to the next firewall rule. • add-dst-to-address-list - add destination address to address list specified by `address-list` parameter • add-src-to-address-list - add source address to address list specified by `address-list` parameter • jump - jump to the user-defined chain specified by the value of `jump-target` parameter • log - add a message to the system log containing the following data: in-interface, out-interface, src-mac, protocol, src-ip:port->dst-ip:port and length of the packet. After a packet is matched it is passed to the next rule in the list, similar as `passthrough` • passthrough - if a packet is matched by the rule, increase counter and go to next rule (useful for statistics) • return - passes control back to the chain from where the jump took place
address-list (*name*; Default:)	Name of the address list to be used. Applicable if action is add-dst- to-address-list or add-src-to-address-list
address-list-timeout (*none-dynamic | none-static | time*; Default: **none-dynamic**)	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 • Value of `none-dynamic` (`00:00:00`) will leave the address in the address list till reboot • Value of `none-static` will leave the address in the address list forever and will be included in the configuration export/backup
jump-target (*name*; Default:)	Name of the target chain to jump to. Applicable only if `action=jump`
log (*yes | no; Default:* **no**)	Add a message to the system log containing the following data: in-interface, out-interface, src-mac, protocol, src-ip:port->dst-ip:port, and length of the packet. Allows to log packets even if action is not "**log**", useful for debugging firewall.
log-prefix (*string*; Default:)	Adds specified text at the beginning of every log message. Applicable if *`action=log`* or *`log=yes`* configured.

Stats

To view matching statistics by firewall rules, run `/ip firewall filter print stats` command or `/ipv6 firewall filter print stats` for IPv6 firewall.

Property	Description
bytes (*integer*)	The total amount of bytes matched by the rule
packets (*integer*)	The total amount of packets matched by the rule

```
[admin@MikroTik] > ip firewall filter print stats
Flags: X - disabled, I - invalid, D - dynamic
 #    CHAIN
           ACTION                              BYTES         PACKETS
 0  D ;;; special dummy rule to show fasttrack counters
      forward
           passthrough              50 507 925 242      50 048 246
 1    ;;; defconf: drop invalid
      forward
           drop                            432 270           9 719
 2    ;;; defconf: drop invalid
      input
           drop                            125 943           2 434
 3    input
           accept                   20 090 211 549      20 009 864
 4    ;;; defconf: accept ICMP
      input
           accept                          634 926           7 648
 5    ;;; defconf: drop all not coming from LAN
      input
           drop                          4 288 079          83 428
 6    ;;; defconf: accept in ipsec policy
      forward
           accept                                0               0
7    ;;; defconf: accept out ipsec policy
      forward
           accept                                0               0
8    ;;; defconf: fasttrack
      forward
           fasttrack-connection     28 505 528 775      31 504 682
9    ;;; defconf: accept established,related, untracked
      forward
           accept                   28 505 528 775      31 504 682
10    ;;; defconf: drop all from WAN not DSTNATed
      forward
           drop                                  0               0
```

Statistics parameters can be reset by following commands:

Command	Description
reset-counters (*id*)	Reset statistics counters for specific firewall rule or list of rules.
reset-counters-all	Reset statistics counters for all firewall rules in the table.

Other Useful Commands

By default print is equivalent to `print static` and shows only static rules.

To print also dynamic rules use `print all`.

Or to print only dynamic rules use `print dynamic`.

Matchers

Tables below shows all the properties that can be used as a matchers in the firewall rules.

Matchers are executed in a specific order.

For IPv4:

- Source MAC Address
- In/Out interfaces

- In/Out interface lists
- IP Range
- Address type
- Address list
- TTL
- DSCP
- Length
- TLS
- IPv4 Options
- Dst Port
- Src Port
- Any Port
- TCP Options
- TCP MSS
- ICMP Codes
- Ingress Priority
- Priority
- Packet Mark
- Realm (routing table)
- Hotsopot
- Connection Mark
- Connection State
- Connection NAT State
- Connection Bytes
- Connection Limit
- Connection Rate
- Ipsec Policy
- Helper
- String (content)
- PSD
- Layer7
- Random
- Nth
- PCC
- Limit
- Dst Limit
- Log

For IPv6:

- Address type
- Address list
- Source MAC Address
- In/Out interfaces
- In/Out interface lists
- Hop Limit
- DSCP
- Length
- TLS
- IPv6 Header
- Dst Port
- Src Port
- Any Port
- TCP Options
- TCP MSS
- ICMPv6 Codes
- Ingress Priority
- Priority
- Packet Mark
- Connection Mark
- Connection State
- Connection NAT State
- Connection Bytes
- Connection Limit
- Connection Rate
- Ipsec Policy

- Helper
- Match String (content)
- Random
- Nth
- PCC
- Limit
- Dst Limit
- Log

Properties are split in two parts:

- **stateless** - properties do not require connection tracking to function and can be used in stateless RAW firewall matching.
- **stateful** - properties either require connection tracking to function or is available only in stateful firewall config.

Stateless Properties

Property	Description
chain (*name*; Default:)	Specifies to which chain rule will be added. If the input does not match the name of an already defined chain, a new chain will be created
comment (*string*; Default:)	Descriptive comment for the rule
content (*string*; Default:)	Match packets that contain specified text
dscp (*integer: 0..63*; Default:)	Matches DSCP IP header field.
dst-address (*IP/netmask* | *IP range*; Default:)	Matches packets whose destination is equal to the specified IP or falls into the specified IP range.
dst-address-list (*name*; Default:)	Matches the destination address of a packet against a user-defined address-list. Supports only one list!
dst-address-type (*unicast* | *local* | *broadcast* | *multicast*)	Matches destination address type: - `unicast` - IP address used for point to point transmission - `local` - if dst-address is assigned to one of the router's interfaces - `broadcast` - packet is sent to all devices in a subnet - `multicast` - packet is forwarded to a defined group of devices
dst-limit (*integer[/time],integer,dst-address* | *dst-port* | *src-address[/time]*; Default:)	Matches packets until a given rate is exceeded. Rate is defined as packets per time interval. As opposed to the limit matcher, every flow has its own limit. Flow is defined by a mode parameter. Parameters are written in the following format: `rate[/time],burst,mode[/expire]`. - **rate** - packet count per time interval per-flow to match - **time** - specifies the time interval in which the packet count rate per flow cannot be exceeded (optional, 1s will be used if not specified) - **burst** - initial number of packets per flow to match: this number gets recharged by one every `time`/`rate`, up to this number - **mode** - this parameter specifies what unique fields define flow (src-address, dst-address, src-and-dst-address, dst-address-and-port, addresses-and-dst-port) - **expire** - specifies interval after which flow with no packets will be allowed to be deleted (optional)
dst-port (*integer[-integer]: 0..65535*; Default:)	List of destination port numbers or port number ranges

fragment (*yes|no*; Default:)	Matches fragmented packets. The first (starting) fragment does not count. If connection tracking is enabled there will be no fragments as the system automatically assembles every packet. **IPv4** only.
header (*Type[:Mode]; Mode=contains|exact; Type=hop|dst|route|frag|ah|esp|none|proto*)	Matches IPv6 next-header. Two types of header matching are possible controlled by "mode" parameter: • **contains** - soft matching, matches at least selected headers • **exact** - matches exact set of selected headers **IPv6** only**.**
hop-limit (Mode:Value; Mode=equal | greater-than | less-than | not-equal; Value=0..255)	Matches hop limit field in the IPv6 header. **IPv6** only.
hotspot (*auth | from-client | http | local-dst | to-client*; Default:)	Matches packets received from HotSpot clients against various HotSpot matchers. • `auth` - matches authenticated HotSpot client packets • `from-client` - matches packets that are coming from the HotSpot client • `http` - matches HTTP requests sent to the HotSpot server • `local-dst` - matches packets that are destined to the HotSpot server • `to-client` - matches packets that are sent to the HotSpot client **IPv4** Only.
icmp-options (*integer:integer*; Default:)	Matches ICMP type: code fields
in-bridge-port (*name*; Default:)	Actual interface the packet has entered the router if the incoming interface is a bridge. Works only if `use-ip-firewall` is enabled in bridge settings.
in-bridge-port-list (*name*; Default:)	Set of interfaces defined in interface list. Works the same as in-bridge-port
in-interface (*name*; Default:)	Interface the packet has entered the router
in-interface-list (*name*; Default:)	Set of interfaces defined in interface list. Works the same as in-interface
ingress-priority (*integer: 0..63*; Default:)	Matches the priority of an ingress packet. Priority may be derived from VLAN, WMM, DSCP, or MPLS EXP bit. read more
ipsec-policy (*in | out, ipsec | none*; Default:)	Matches the policy used by IPsec. Value is written in the following format: **`direction, policy`**. The direction is Used to select whether to match the policy used for decapsulation or the policy that will be used for encapsulation. • `in` - valid in the PREROUTING, INPUT, and FORWARD chains • `out` - valid in the POSTROUTING, OUTPUT, and FORWARD chains • `ipsec` - matches if the packet is subject to IPsec processing; • `none` - matches packets that are not subject to IPsec processing (for example, IPSec transport packet). For example, if a router receives an IPsec encapsulated Gre packet, then rule `ipsec-policy=in,ipsec` will match Gre packet, but a rule `ipsec-policy=in,none` will match the ESP packet.

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* ; Default:)	Matches IPv4 header options. • `any` - match packet with at least one of the ipv4 options • `loose-source-routing` - match packets with a 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 a strict source routing option • `timestamp` - match packets with a timestamp **IPv4** only.
limit (*integer,time,integer*; Default:)	Matches packets up to a limited rate (packet rate or bit rate). A rule using this matcher will match until this limit is reached. Parameters are written in the following format: `rate[/time],burst:mode`. • **rate** - packet or bit count per time interval to match • **time** - specifies the time interval in which the packet or bit rate cannot be exceeded (optional, 1s will be used if not specified) • **burst** - initial number of packets or bits to match: this number gets recharged every 10ms so burst should be at least 1/100 of a rate per second • **mode** - packet or bit mode
nth (*integer,integer*; Default:)	Matches every nth packet: `nth=2,1` rule will match every first packet of 2, hence, 50% of all the traffic that is matched by the rule
out-bridge-port (*name*; Default:)	Actual interface the packet leaves the router if the outgoing interface is a bridge. Works only if **`use-ip-firewall`** is enabled in bridge settings.
out-bridge-port-list (*name*; Default:)	Set of interfaces defined in interface list. Works the same as out-bridge-port
out-interface (; Default:)	Interface the packet is leaving the router
out-interface-list (*name*; Default:)	Set of interfaces defined in interface list. Works the same as out-interface
packet-mark (*no-mark* / *string*; Default:)	Matches packets marked via mangle facility with particular packet mark. If **`no-mark`** is set, the rule will match any unmarked packet.
packet-size (*integer[-integer]:0..65535*; Default:)	Matches packets of specified size or size range in bytes.
per-connection-classifier (*ValuesToHash:Denominator/Remainder*; Default:)	PCC matcher (or Per Stream Classifier) allows dividing traffic into equal streams with the ability to keep packets with a specific set of options in one particular stream. Streams are hashed based on selected values to hash: • both-addresses • both-addresses-and-ports • both-ports • dst-address • dst-address-and-port • dst-port • src-address • src-address-and-port • src-port Read more >>
port (*integer[-integer]: 0..65535*; Default:)	Matches if any (source or destination) port matches the specified list of ports or port ranges. Applicable only if `protocol` is TCP or UDP

priority (*integer: 0..63*; Default:)	Matches the packet's priority after a new priority has been set. Priority may be derived from VLAN, WMM, DSCP, MPLS EXP bit, or from the priority that has been set using the set-priority action. Read more
protocol (*name or protocol ID*; Default: **tcp**)	Matches particular IP protocol specified by protocol name or number
psd (*integer,time,integer,integer*; Default:)	Attempts to detect TCP and UDP scans. Parameters are in the following format `WeightThreshold, DelayThreshold, LowPortWeight, HighPortWeight` • **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** - the weight of the packets with privileged (<1024) destination port • **HighPortWeight** - the weight of the packet with a non-privileged destination port **IPv4** only.
random (*integer: 1..99*; Default:)	Matches packets randomly with a given probability
src-address (*Ip/Netmask, Ip range*; Default:)	Matches packets whose source is equal to a specified IP or falls into a specified IP range
src-address-list (*name*; Default:)	Matches the source address of a packet against a user-defined address list. Supports only one list!
src-address-type (*unicast | local | broadcast | multicast | blackhole | prohibit | unreachable ;* Default:)	mote{ta{tableMatches source address type: • unicast - IP address used for point to point transmission • local - if an address is assigned to one of the router's interfaces • broadcast - packet is sent to all devices in the subnet • multicast - packet is forwarded to a defined group of devices
src-port (*integer[-integer]: 0..65535*; Default:)	List of source ports and ranges of source ports. Applicable only if a protocol is TCP or UDP
src-mac-address (*MAC address*; Default:)	Matches the source MAC address of the packet
tcp-flags (*ack | cwr | ece | fin | psh | rst | syn | urg*; Default:)	Matches specified TCP flags • `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[-integer]: 0..65535*; Default:)	Matches TCP MSS value of an IP packet
time (*time-time,sat | fri | thu | wed | tue | mon | sun*; Default:)	Allows to create a filter based on the packets' arrival time and date or, for locally generated packets, departure time and date
tls-host (*string*; Default:)	Allows matching HTTPS traffic based on TLS SNI hostname. Accepts GLOB syntax for wildcard matching. Note that the matcher will not be able to match the hostname if the TLS handshake frame is fragmented into multiple TCP segments (packets). Watch our video about this value.
ttl (*integer: 0..255*; Default:)	Matches packets TTL value. **IPv4** Only.

Stateful Properties

Property	Description
connection-bytes (*integer-integer*; Default:)	Matches packets only if a given amount of bytes has been transferred through the particular connection. 0 - means infinity, for example `connection-bytes=2000000-0` means that the rule matches if more than 2MB has been transferred through the relevant connection
connection-limit (*integer,netmask*; Default:)	Matches connections per address or address block after a given value is reached. Should be used together with `connection-state=new` and/or with `tcp-flags=syn` because matcher is very resource-intensive
connection-mark (*no-mark | string*; Default:)	Matches packets marked via mangle facility with particular connection mark. If **no-mark** is set, the rule will match any unmarked connection
connection-nat-state (*srcnat | dstnat*; Default:)	Can match connections that are srcnatted, distracted, or both. Note that `connection-state=related` connections connection-nat-state is determined by the direction of the first packet. and if connection tracking needs to use dst-nat to deliver this connection to the same hosts as the main connection it will be in connection-nat-state=dstnat even if there are no dst-nat rules at all
connection-rate (*integer 0..4294967295*; Default:)	Connection Rate is a firewall matcher that allows capturing traffic based on the present speed of the connection
connection-state (*established | invalid | new | related | untracked*; Default:)	Interprets the connection tracking analytics data for a particular packet: • `established` - a packet that belongs to an existing connection • `invalid` - a packet that does not have a determined state in connection tracking (usually - severe out-of-order packets, packets with wrong sequence/ack number, or in case of a resource over usage on the router), for this reason, an invalid packet will not participate in NAT (as only connection-state=new packets do), and will still contain original source IP address when routed. We strongly suggest dropping all *connection-state=invalid* packets in the firewall filter forward and input chains • `new` - the packet has started a new connection or is otherwise associated with a connection that has not seen packets in both directions. • `related` - a packet that is related to, but not parts of an existing connection, such as ICMP errors or a packet that begins an FTP data connection • `untracked` - packet that was set to bypass connection tracking in firewall RAW tables.
connection-type (*ftp | h323 | irc | pptp | quake3 | sip | tftp*; Default:)	Matches packets from related connections based on information from their connection tracking helpers. A relevant connection helper must be enabled under the: `/ip firewall service-port`
layer7-protocol (*name*; Default:)	Layer7 filter name defined in layer7 protocol menu. Read more>>.
p2p ()	Matches some unencrypted P2P protocols. Deprecated in modern days since mostly everything is encrypted and requires deep packet inspection to identify. **IPv4** only.
realm (*integer: 0..4294967295*; Default:)	**IPv4** only.
routing-mark (*string*; Default:)	Matches packets marked by mangle facility with particular routing mark