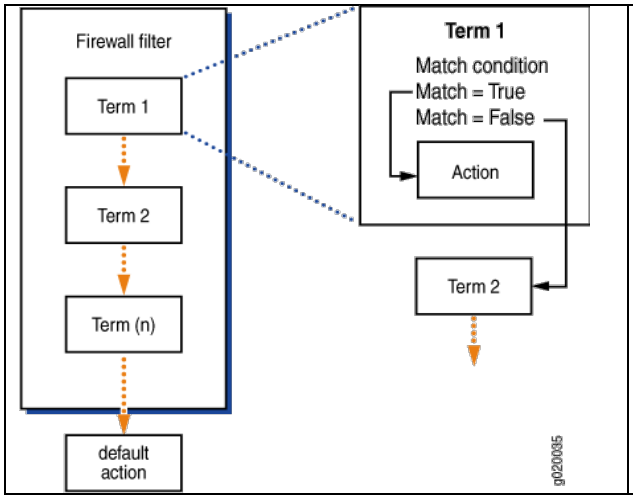
FILTERS (JUNIPER) cloudkod.com



1. The packet is evaluated against the conditions in the `from` statement in the first term.
2. If the packet matches all the conditions in the term, the action in the `then` statement is taken and the evaluation ends. Subsequent terms in the filter are not evaluated.
3. If the packet does not match all the conditions in the term, the packet is evaluated against the conditions in the `from` statement in the second term.

   This process continues until either the packet matches the conditions in the `from` statement in one of the subsequent terms or there are no more terms in the filter.

4. If a packet passes through all the terms in the filter without a match, the packet is discarded.

Every firewall filter contains an implicit deny statement at the end of the filter, which equivalent to following explicit filter term:

```
term implicit-rule {
then discard;
}
```

Where You Can Apply Filters

After you configure the firewall filter, you can apply it to the following:

- Port—Filters Layer 2 traffic transiting system ports.
- VLAN—Filters & provides access control for Layer 2 packets that enter a VLAN, are bridged within a VLAN, or leave VLAN.
- Layer 3 (routed) interface—Filters traffic on IPv4 and IPv6 interfaces, routed VLAN interfaces (RVI), and the loopback interface. The loopback interface filters traffic sent to the switch itself or generated by the switch.
- Layer 2 CCC interface—Filters Layer 2 circuit cross-connect (CCC) interfaces.
- MPLS—Filters MPLS interfaces.

Applying the appropriate firewall filters to the Routing Engine protects against these types of attacks.

- A TCP flood attack of SYN packets initiating connection requests can overwhelm the device until it can no longer process legitimate connection requests, resulting in denial of service.
- An ICMP flood can overload the device with so many echo requests (ping requests) that it expends all its resources responding and can no longer process valid network traffic, also resulting in denial of service.

Firewall Filters

1) Configure an IPv4 firewall filter allowing protocol messages from AH, GRE, BFD, VRRP, OSPF, BGP, PIM, IGMP, MSDP protocols.

Please see the example below.

```
[edit firewall family inet filter protect-RE-inet]
lab@Mercury# show
term ah {
    from {
        protocol ah;
    }
    then accept;
}
term gre {
    from {
        protocol gre;
    }
    then accept;
}
term bfd {
    from {
        protocol udp;
        destination-port 3784;
    }
    then accept;
}
term vrrp {
    from {
        protocol vrrp;
    }
    then accept;
}
term ospf {
    from {
        protocol ospf;
    }
    then accept;
}
term bgp-1 {
    from {
        protocol tcp;
        destination-port bgp;
    }
    then accept;
}
term bgp-2 {
    from {
        protocol tcp;
        source-port bgp;
    }
    then accept;
}
term pim {
    from {
        protocol pim;
    }
    then accept;
}
term igmp {
    from {
        protocol igmp;
    }
    then accept;
}
term msdp-1 {
    from {
        protocol tcp;
        source-port msdp;
    }
    then accept;
}
term msdp-2 {
    from {
        protocol tcp;
        destination-port msdp;
    }
    then accept;
}
```

2) Configure the firewall filter to accept NTP, RADIUS, DNS, SNMP management protocols only from the 10.10.10/24 network.

Please see the example below.

```
[edit firewall family inet filter protect-RE-inet]
lab@Mercury# show
term ntp {
    from {
        source-address {
            10.10.10.0/24;
        }
        protocol udp;
        source-port ntp;
    }
    then accept;
}
term radius {
    from {
        source-address {
            10.10.10.0/24;
        }
        protocol [ udp tcp ];
        source-port radius;
    }
    then accept;
}
term dns {
    from {
        source-address {
            10.10.10.0/24;
        }
        protocol [ udp tcp ];
        source-port domain;
    }
    then accept;
}
term snmp {
    from {
        source-address {
            10.10.10.0/24;
        }
        protocol udp;
        destination-port snmp;
    }
    then accept;
}
```

3) Configure the firewall filter to accept SSH, Telnet, HTTP, and HTTPS protocols only from the 10.10.1/24 management network.

Please see the example below.

```
[edit firewall family inet filter protect-RE-inet]
lab@Mercury# show
term ssh {
    from {
        source-address {
            10.10.1/24;
        }
        protocol tcp;
        destination-port ssh;
    }
    then accept;
}
term telnet {
    from {
        source-address {
            10.10.1/24;
        }
        destination-port telnet;
    }
    then accept;
}
term http {
    from {
        source-address {
            10.10.1/24;
        }
        protocol tcp;
        destination-port http;
    }
    then accept;
}
term https {
    from {
        source-address {
            10.10.1/24;
        }
        protocol tcp;
        destination-port https;
    }
    then accept;
}
```

4) Configure the firewall filter to accept ICMP and traceroute messages. Ensure that on the D1 D4 devices the flow of the messages is limited with a policer. The policer should be configured with the following parameters: bandwidth limit 100 kbps, burst size 25 kB. Excess traffic must be dropped.

Note that an ingress policer is not supported on EX4200 loopback interfaces in Junos 11.2
Configure the policer with the required parameters as indicated in the example below.

```
[edit firewall]
lab@Mercury# show
policer re-policer {
    if-exceeding {
        bandwidth-limit 100k;
        burst-size-limit 25k;
    }
    then discard;
}
```

Configure the firewall filter terms as shown in the example below.

```
[edit firewall family inet filter protect-RE-inet]
lab@Mercury# show
term icmp {
    from {
        protocol icmp;
    }
    then {
        policer re-policer;
        accept;
    }
}
term traceroute {
    from {
        protocol udp;
    }
    then {
        policer re-policer;
        accept;
    }
}
```

5) Configure the firewall filter to discard any other traffic and increment a named drop counter.

Please see the example below.

```
[edit firewall family inet filter protect-RE-inet]
lab@Mercury# show
term explicit_discard {
    then {
        count dropped;
        discard;
    }
```

6) Apply the firewall filter such as to ensure that it is used for the RE protection.

Apply the firewall filter to the loopback interface as shown in the example below.

```
lab@Mercury# show
lo0 {
    unit 0 {
        family inet {
            filter {
                input protect-RE-inet;
            }
        }
```