Layer7

Summary

Layer7-protocol is a method of searching for patterns in ICMP/TCP/UDP streams.

> ⚠ The L7 matcher is very resource-intensive. Use this feature only for very specific traffic. It is not recommended to use the L7 matcher for generic traffic, such as for blocking web pages. This will almost never work correctly and your device will exhaust its resources, trying to catch all the traffic. Use other features to block webpages by URL.

L7 matcher collects the first **10 packets** of a connection or the first **2KB** of a connection and searches for the pattern in the collected data. If the pattern is not found in the collected data, the matcher stops inspecting further. Allocated memory is freed and the protocol is considered **unknown**. You should take into account that a lot of connections will significantly increase memory and CPU usage. To avoid this, add regular firewall matchers to reduce the amount of data passed to layer-7 filters repeatedly.

An additional requirement is that the layer7 matcher must see both directions of traffic (incoming and outgoing). To satisfy this requirement l7 rules should be set in the **forward** chain. If the rule is set in the **input/prerouting** chain then the same rule **must** be also set in the **output/postrouting** chain, otherwise, the collected data may not be complete resulting in an incorrectly matched pattern.

> ⚠ Layer 7 matcher is case insensitive!

Example L7 patterns compatible with RouterOS can be found on the l7-filter project page.

> ⚠ In some cases when layer 7 regular expression cannot be performed, RouterOS will log *topic=firewall, warning* with an error message stating the problem in the message!

Properties

```
/ip firewall layer7-protocol
```

Property	Description
name (*string*; Default:)	Descriptive name of l7 pattern used by configuration in firewall rules.
regexp (*string*; Default:)	POSIX compliant regular expression is used to match a pattern.

Examples

Simple L7 usage example

First, add Regexp strings to the protocols menu, to define the strings you will be looking for. In this example, we will use a pattern to match RDP packets.

```
/ip firewall layer7-protocol
add name=rdp regexp="rdpdr.*cliprdr.*rdpsnd"
```

Then, use the defined protocols in the firewall.

```
/ip firewall filter

# add few known protocols to reduce mem usage
add action=accept chain=forward comment="" disabled=no port=80 protocol=tcp
add action=accept chain=forward comment="" disabled=no port=443 protocol=tcp

# add l7 matcher
add action=accept chain=forward comment="" disabled=no layer7-protocol=\
    rdp protocol=tcp
```

As you can see before the l7 rule we added several regular rules that will match known traffic thus reducing memory usage.

L7 in the input chain

In this example, we will try to match the telnet protocol connecting to our router.

```
/ip firewall layer7-protocol add comment="" name=telnet regexp="^\\xff[\\xfb-\\xfe].\\xff[\\xfb-\\xfe].\\xff
[\\xfb-\\xfe]"
```

Note that we need both directions which is why we need also the l7 rule in the output chain that sees outgoing packets.

```
/ip firewall filter

add action=accept chain=input comment="" disabled=no layer7-protocol=telnet \
    protocol=tcp

add action=passthrough chain=output comment="" disabled=no layer7-protocol=telnet \
    protocol=tcp
```

Youtube Matcher

⚠ When a user is logged in YouTube will use HTTPS, meaning that L7 will not be able to match this traffic. Only unencrypted HTTP can be matched.

```
/ip firewall layer7-protocol
add name=youtube regexp="(GET \\/videoplayback\\\?|GET \\/crossdomain\\.xml)"
```