## NAME

xtables-monitor — show changes to rule set and trace-events

## SYNOPSIS

**xtables−monitor** [**−t**] [**−e**] [**−4||−6]**

## DESCRIPTION

**xtables-monitor** is used to monitor changes to the ruleset or to show rule evaluation events for packets tagged using the TRACE target. **xtables-monitor** will run until the user aborts execution, typically by using CTRL-C.

## OPTIONS

**−e**, **−−event**

Watch for updates to the rule set.

Updates include creation of new tables, chains and rules and the name of the program that caused the rule update.

**−t**, **−−trace**

Watch for trace events generated by packets that have been tagged using the TRACE target.

**−4** Restrict output to IPv4.

**−6** Restrict output to IPv6.

## EXAMPLE OUTPUT

**xtables-monitor −−trace**

1 TRACE: 2 fc475095 raw:PREROUTING:rule:0x3:CONTINUE −4 −t raw −A PREROUTING −p icmp −j TRACE
2 PACKET: 0 fc475095 IN=lo LL=0x304 00000000000000000000000800 SRC=127.0.0.1 DST=127.0.0.1 LEN=84 TOS=0x0 TTL=64 ID=38349DF
3 TRACE: 2 fc475095 raw:PREROUTING:return:
4 TRACE: 2 fc475095 raw:PREROUTING:policy:ACCEPT
5 TRACE: 2 fc475095 filter:INPUT:return:
6 TRACE: 2 fc475095 filter:INPUT:policy:DROP
7 TRACE: 2 0df9d3d8 raw:PREROUTING:rule:0x3:CONTINUE −4 −t raw −A PREROUTING −p icmp −j TRACE

The first line shows a packet entering rule set evaluation. The protocol number is shown (AF_INET in this case), then a packet identifier number that allows to correlate messages coming from rule set evaluation of this packet. After this, the rule that was matched by the packet is shown. This is the TRACE rule that turns on tracing events for this packet.

The second line dumps information about the packet. Incoming interface and packet headers such as source and destination addresses are shown.

The third line shows that the packet completed traversal of the raw table PREROUTING chain, and is returning, followed by use the chain policy to make accept/drop decision (the example shows accept being applied). The fifth line shows that the packet leaves the filter INPUT chain, i.e., no rules in the filter tables INPUT chain matched the packet. It then got DROPPED by the policy of the INPUT table, as shown by line six. The last line shows another packet arriving −− the packet id is different.

When using the TRACE target, it is usually a good idea to only select packets that are relevant, for example via
iptables −t raw −A PREROUTING −p tcp −−dport 80 −−syn −m limit −−limit 1/s −j TRACE

**xtables-monitor −−event**

1 EVENT: nft: NEW table: table filter ip flags 0 use 4 handle 444
2 EVENT: # nft: ip filter INPUT use 2 type filter hook input prio 0 policy drop packets 0 bytes 0

3 EVENT: # nft: ip filter FORWARD use 0 type filter hook forward prio 0 policy accept packets 0 bytes 0
4 EVENT: # nft: ip filter OUTPUT use 0 type filter hook output prio 0 policy accept packets 0 bytes 0
5 EVENT: −4 −t filter −N TCP
6 EVENT: −4 −t filter −A TCP −s 192.168.0.0/16 −p tcp −m tcp −−dport 22 −j ACCEPT
7 EVENT: −4 −t filter −A TCP −p tcp −m multiport −−dports 80,443 −j ACCEPT
8 EVENT: −4 −t filter −A INPUT −p tcp −j TCP
9 EVENT: −4 −t filter −A INPUT −m conntrack −−ctstate RELATED,ESTABLISHED −j ACCEPT
10 NEWGEN: GENID=13904 PID=25167 NAME=iptables-nftables-restore

This example shows event monitoring. Line one shows creation of a table (filter in this case), followed by three base hooks INPUT, FORWARD and OUTPUT. The iptables-nftables tools all create tables and base chains automatically when needed, so this is expected when a table was not yet initialized or when it is re-created from scratch by iptables-nftables-restore. Line five shows a new user-defined chain (TCP) being added, followed by addition a few rules. the last line shows that a new ruleset generation has become active, i.e., the rule set changes are now active. This also lists the process id and the programs name.

## LIMITATIONS

**xtables-monitor** only works with rules added using iptables-nftables, rules added using iptables-legacy cannot be monitored.

## BUGS

Should be reported or by sending email to netfilter-devel@vger.kernel.org or by filing a report on https://bugzilla.netfilter.org/.

## SEE ALSO

**iptables**(8), **xtables**(8), **nft**(8)