Netfilter updates since last NetDev conference

Pablo Neira Ayuso
<pablo@netfilter.org>
Netdev Conference (California, USA)
Jul 16 2024

Topics

- nftables releases since last NetDev

nftables releases

- 1 release for userspace, since last NetDev conference in 2023:
  - 1.1.0: 16 Jul 2024 (268 commits)
    - Mostly fixes
    - Many new tests
- Plans to release stable version
  - 1.0.6.1 (TBA)

nftables 1.1.0: fixes

- Restore compatibility set element dump with <= 0.9.8

  add **element** t s { 23 counter packets 10 bytes 20 timeout 10s }
  add **element** t s { 42 timeout 10s counter packets 10 bytes 20 }

- Disallow ifname less than zero

  meta iifname “”
  Error: Empty string is not allowed

- Do not omit tproxy port for non-value expressions

  tproxy ip to 127.0.0.1:8000
  meta l4proto 6 **tproxy** ip to 127.0.0.1:**symhash** mod 2 map { 0 : 8000, 1 : 8010 }

nftables 1.1.0: fixes (2)

- Listing meta hour with negative time offset

  TZ=UTC-4 nft add rule x y meta hour “22:00”

- Byteorder conversion with {ct,meta} statements

```
map mapv6 {
        typeof ip6 dscp : meta mark;
}
meta mark set ip6 dscp map @map1

 [ payload load 2b @ network header + 0 => reg 1 ]
 [ bitwise reg 1 = ( reg 1 & 0x0000c00f ) ^ 0x00000000 ]
 [ byteorder reg 1 = ntoh(reg 1, 2, 2) ]
 [ bitwise reg 1 = ( reg 1 >> 0x00000006 ) ]
 [ lookup reg 1 set mapv6 dreg 1 ]
 [ meta set mark with reg 1 ]
```

nftables 1.1.0: fixes (3)

- Unbreak create set command

```
define ip-block-4 = { 1.1.1.1 }
create set netdev filter ip-block-4-test {
        type ipv4_addr
        flags interval
        auto-merge
        elements = $ip-block-4
}
```

- Restore rule replace command

replace rule ip t1 c1 handle 3 'jhash ip protocol . ip saddr mod 170 vmap { 0-94 : goto wan1, 95-169 : goto wan2, 170-269 }"

- Restore addition of netdevice to flowtable

create flowtable inet filter f1 { hook ingress priority 0; counter }
add flowtable inet filter f1 { devices = { dummy1 } ; }

nftables 1.1.0: fixes (4)

- Byteorder conversion in set with concatenation and ranges

```
map ipsec_in {
        typeof ipsec in reqid . iif : verdict
        flags interval
}

ipsec in reqid . iif vmap @ipsec_in
  [ xfrm load in 0 reqid => reg 1 ]
  [ byteorder reg 1 = hton(reg 1, 4, 4) ]
  [ meta load iif => reg 9 ]
  [ byteorder reg 9 = hton(reg 9, 4, 4) ]
  [ lookup reg 1 set ipsec_in dreg 0 ]
```

- Support for chain multidevice in JSON

nftables 1.1.0: fixes (5)

- Lots of fixes to address input sanitization:
  - turn valuation assert() into errors
  - turn evaluation error instead of crash
  - parser crash
  - expression with no datatype & incompatible key with datatype in set,
  - OOB
  - memleaks

- Fix monitor mode with set intervals & concatenation
- Unbreak tcp option with numbers
  tcp option 254
- Unbreak {meta,ct} mark statement with maps
  meta mark set vlan id map { 1 : 0x00000001, 4095 : 0x00004095 }

- Reject large raw payload and concat expression

  Error: Concatenation of size 544 exceeds maximum size of 512
  udp length . @th,0,512 . @th,512,512 { 47-63 . 0xe373135363130 . 0x331313037353353203 }
  ^^^^^^^^

nftables 1.1.0: fixes (6)

- Search for group, rt_mark, rt_realms at:
  - /etc/iproute2/
  - /use/share/iproute2/
- … and display values via nft describe

  # nft **describe** meta rtclassid
  meta expression, datatype realm (routing realm) (basetype integer), 32 bits

  pre-defined symbolic constants from /etc/iproute2/rt_realms (in decimal):
  cosmos 0
- Reject statement with range
  meta mark set 0-100

- Support for auto-merge flag in sets in JSON
- Print 0s in time datatype
- Speed up *list tables* by fetching tables only

nftables 1.1.0: fixes (7)

- Skip byteorder conversion with 8-byte fields

```
set test {
        type ipv4_addr . ether_addr . inet_proto
        flags interval
}

ip saddr . ether saddr . meta l4proto @test counter
```

- Honor -t/--terse with *list table* and *list set* to speed up listing
- Allow for host-endian in set lookups

```
map ipsec_in {
    typeof ipsec in reqid . iif : verdict
    flags interval
 }

 ipsec in reqid . 100 @ipsec_in
```

- Better error report when *destroy* command is not supported (requires >= 6.3)

nftables 1.1.0: fixes (8)

- Allow to define maps with:
  - ct timeout
  - ct expectation
  - ct helper
- Translate meter into dynamic set

add **rule** t c **tcp** dport 80 **meter** m size 128 { **ip** saddr timeout 2s **limit** rate 10/second }

becomes

```
set m {
        type ipv4_addr
        size 128
        flags dynamic,timeout
 }

 tcp dport 80 update @m { ip saddr timeout 2s limit rate 10/second burst 5 packets }
```

nftables 1.1.0: fixes (9)

- No payload merge on negation

  tcp sport != 22 tcp dport != 23

- JSON updates:
  - List empty chain early before set/maps
  - Support for maps with concatenated data
  - Support for synproxy objects
- Restore binop syntax for flags

  tcp flags & (fin | syn | rst | ack) == syn

- Cross-day meta hour issues

  TZ=EADT $NFT add rule t c meta hour "03:00"-"14:00"

- Remove prefix notation from mark

  meta mark & 0xffffff00 == 0xffffff00

  instead of

  meta mark 0xffffff00/24

nftables 1.1.0: fixes (10)

- Use numeric icmp codes in listing
  - Codes are dependent of type
- Add table persist flag to JSON
- Support for variables in map expressions

```
define dst_map = { ::1234 : 5678 }


table ip6 nat {
  map dst_map {
    typeof ip6 daddr : tcp dport;
    elements = $dst_map
  }
  chain prerouting {
    ip6 nexthdr tcp redirect to ip6 daddr map @dst_map
  }
}
```

nftables 1.1.0: fixes (11)

- VLAN support:
  # payload statement
  ip saddr 10.1.1.1 icmp type echo-request vlan id set 321

  # payload expression (QinQ matching)
  ether type 8021ad **vlan** id 10 **vlan** type 8021q **vlan** id 100 **vlan** type ip **accept**

- Recycle existing cache if generation ID did not change
- Better error reporting when redefining chain

  ruleset.nft:7:9-52: Error: Chain "input" already exists in table ip 'filter' with different declaration
  type filter hook postrouting priority filter;
  ^^

- Issues with variables

  define m = { 3, 4 }
  add **element** ip a x \$m
  add **element** ip a x { 5 }

nftables 1.1.0: fixes (12)

- Broader IPv4-Mapped IPv6 (similar to iptables)
  aaaa::1.2.3.4
- -f/--filename includes path relative to the current (the including) file's directory
- -I/--include: default include path now searched at the end.
- New string preprocessor (only for log statement)
  define message=”test”
  log prefix “my $message”
- Fix set element deletion is maps:

```
map m {
        typeof ct bytes : meta priority
        flags interval
        elements = { 2048001-4000000 : 1:2 }
}

meta priority set ct bytes map @m
```

nftables 1.1.0: fixes (13)

- Unbreak -o/--optimize with counter statement

  # nft -c -o -f ruleset.nft

  Merging:
  ruleset.nft:5:17-45: ct state invalid counter drop
  ruleset.nft :6:17-59: ct state established,related counter accept
  into:
  ct state vmap { invalid counter : drop, established counter : accept, related counter : accept }

  Merging:
  ruleset.nft:7:17-43: tcp dport 80 counter accept
  ruleset.nft:8:17-44: tcp dport 123 counter accept
  into:
  tcp dport { 80, 123 } counter accept
  Merging:
  ruleset.nft:9:17-64: ip saddr 1.1.1.1 ip daddr 2.2.2.2 counter accept
  ruleset.nft:10:17-62: ip saddr 1.1.1.2 ip daddr 3.3.3.3 counter drop
  into:
  ip saddr . ip daddr vmap { 1.1.1.1 . 2.2.2.2 counter : accept, 1.1.1.2 . 3.3.3.3 counter : drop }

nftables 1.1.0: fixes (12)

- Broader IPv4-Mapped IPv6 (similar to iptables)
  aaaa::1.2.3.4
- -f/--filename includes path relative to the current (the including) file's directory
- -I/--include: default include path now searched at the end.
- New string preprocessor (only for log statement)
  define message=”test”
  log prefix “my $message”
- Fix set element deletion is maps:

```
map m {
        typeof ct bytes : meta priority
        flags interval
        elements = { 2048001-4000000 : 1:2 }
}

meta priority set ct bytes map @m
```

Kernel updates

- Many fixes and stable backports up to 4.19
- 6.9: Pipapo improvements with transactions (Florian)
- 6.10-rc: Reduce set element transaction object to 96 bytes (Florian)