NFTables

forwarding: https://www.eukhost.com/kb/how-to-enable-ip-forwarding-on-linux-ipv4-ipv6/

```
apt install nftables
echo 1 > /proc/sys/net/ipv4/ip_forward

nft list ruleset
nft add table nat
nft add chain ip nat prerouting { type nat hook prerouting priority 100 \; }
nft add chain ip nat postrouting { type nat hook postrouting priority 100 \;
}
nft add rule nat postrouting masquerade

#portforwarding
nft add rule nat prerouting iif lan1 tcp dport 443 dnat 192.168.0.10:443 #
ip needs to be routed to other interface then in-interface (here lan1)
```

named priorities (>=0.9.1):
https://thermalcircle.de/doku.php?id=blog:linux:nftables_packet_flow_netfilter_hooks_detail

links

- https://developers.redhat.com/blog/2017/01/10/migrating-my-iptables-setup-to-nftables/
- https://wiki.gentoo.org/wiki/Nftables/Examples#Basic_routing_firewall
- https://wiki.nftables.org/wiki-nftables/index.php
- https://wiki.nftables.org/wiki-nftables/index.php/Quick_reference-nftables_in_10_minutes

hwnat

https://github.com/frank-w/BPI-R2-4.14/commits/5.12-hnat

ipv6 mangle does not support hnat (connection reset!)

to get hwnat working, a newer version of nftables is needed than available in debian buster

https://github.com/frank-w/nftables-bpi

compiled: https://drive.google.com/drive/folders/1hajKvqQa96WRrAy52fQX90i59I1s0h-i?usp=sharing

basic IPv4 Ruleset:

```
flush ruleset
table ip filter {
        flowtable f {
                hook ingress priority filter + 1
```

```
                devices = { lan3, lan0, wan }
                flags offload;
        }
        chain input {
                type filter hook input priority filter; policy accept;
        }

        chain output {
                type filter hook output priority filter; policy accept;
        }

        chain forward {
                type filter hook forward priority filter; policy accept;
                ip protocol { tcp, udp } flow add @f
        }
}
table ip nat {
        chain post {
                type nat hook postrouting priority filter; policy accept;
                oifname "wan" masquerade
        }

        chain pre {
                type nat hook prerouting priority filter; policy accept;
        }
}
```

basic v6 Ruleset (hw-nat for IPv6 not supported):

```
flush ruleset
table ip6 filter {
        flowtable f {
                hook ingress priority 1
                devices = { lan3, lan0, wan }
                flags offload;
        }
        chain input {
                type filter hook input priority 0; policy accept;
        }

        chain output {
                type filter hook output priority 0; policy accept;
        }

        chain forward {
                type filter hook forward priority 0; policy accept;
#                ip6 nexthdr { tcp, udp } flow add @f
        }
}
table ip6 nat {
```

```
        chain post {
                type nat hook postrouting priority 0; policy accept;
                #oifname "wan" masquerade
        }

        chain pre {
                type nat hook prerouting priority 0; policy accept;
        }
}
```

test it:

```
nft -f nft-nat-flowoffload.nft
#generate traffic from client e.g. iperf3
cat /sys/kernel/debug/mtk_ppe/entries
```

IPV6-Setup

```
#!/bin/bash
#on main-router:
#ip -6 route add fd00:a2::/64 via fd00:a::12
#ip -6 route add 2001:470:xxxx:a2::/64 via 2001:470:xxxx::12

ip -6 addr add fd00:a::12/64 dev wan
ip -6 addr add fd00:a2::12/64 dev lan3

ip -6 addr add 2001:470:xxxx::12/64 dev wan
ip -6 addr add 2001:470:xxxx:a2::12/64 dev lan3

sysctl -w net.ipv6.conf.all.forwarding=1
```