# nftables



## Ogólne

Pakiet zastępuje stare *iptables* i nie jest z nim kompatybilny. Korzysta z jądra i składnika *nft* w *userspace*.

## Kernel

[*] Networking support -->
    Networking options -->
        [*] Network packet filtering framework (Netfilter) -->
            Core Netfilter Configuration  --->
                <*> FTP protocol support
                <*> NetBIOS name service protocol support
                <*> SIP protocol support
                <*> Connection tracking netlink interface
                <*> Network Address Translation support
                <*> Netfilter nf_tables support
                    [*]   Netfilter nf_tables netdev tables support
                    [*]   Netfilter nf_tables mixed IPv4/IPv6 tables support
                    // Poniższe odpowiada za **ct state** (*Connection Tracking*):
                    <*>   Netfilter nf_tables conntrack module
                    <*>   Netfilter nf_tables log module
                    <*>   Netfilter nf_tables masquerade support
                    <*>   Netfilter nf_tables redirect support
                    <*>   Netfilter nf_tables nat module
                    <*>   Netfilter nf_tables reject support
                    <*>   Netfilter nf_tables netdev packet forwarding support
                <*> Netfilter Xtables support (required for ip_tables) // Dla iptables
                    <*>   nfmark target and match support
                    <*>   LOG target support
                    <*>   "SNAT and DNAT" targets support
                    <*>   "NFLOG" target support
                    <*>   REDIRECT target support
                    <*>   MASQUERADE target support
                    <*>   "conntrack" connection tracking match support
                    <*>   IPsec "policy" match support
                    <*>   "state" match support
            IP: Netfilter Configuration  --->
                <*> IPv4 socket lookup support
                [*] IPv4 nf_tables support
                [*] ARP nf_tables support
                <*> ARP packet logging
                <*> IPv4 packet logging
                <*> IPv4 packet rejection

<*> IP tables support (required for filtering/masq/NAT)
<*> Packet filtering
    <*> REJECT target support
<*>  iptables NAT support
    <*> MASQUERADE target support
<*> Packet mangling

# Jak to działa?

W *nftables* nie ma predefiniowanych tabel i łacuchów. Najpierw musimy stworzyć tabelę, nadać jej nazwę i określić jej „*address family*”, którym może być:

- **ip** (łańcuchy IPv4)
- **ip6** (łańcuchy IPv6)
- **inet** (łańcuchy IPv4 i IPv6)
- **arp** (łańcuchy MAC)
- **bridge**
- **netdev** (przedwstępne filtrowanie)
- itp.

Następnie tworzymy łańcuchy, które służą do grupowania reguł. Reguły mają „haki” (hook):

- **prerouting** - wstępna filtracja zanim pakiety wejdą do systemu
- **input** - pakiety, które weszły do systemu
- **forward** - pakiety, które weszły do systemu, ale przeznaczone są dla innego
- **output** - pakiety, które zostały wysłane przez lokalny system
- **postrouting** - wszystkie pakiety opuszczające system niezależnie od źródła
- **ingress** - wszystkie pakiety wchodzące do systemu jeszcze przed preroutingiem (inet)

# Obsługa

Tworzenie tabeli:
# **nft add table ip** *nazwaTabeli*
# **nft add table arp** *nazwaTabeli*
# **nft list tables** // Jakie utworzono tabele
# **nft list table ip** *nazwaTabeli* // Zawartość konkretnej tabeli
# **nft delete table ip** *nazwaTabeli* // Usunięcie tabeli

Dodawanie łańcucha:
# **nft add chain ip** *nazwaTabeli nazwaŁańcucha* "{**type filter hook input priority 0**;}"

Wyświetlanie łańcucha:
# **nft list chain ip** *nazwaTabeli nazwaŁańcucha*

Usuwanie (pustego) łańcucha:
# **nft delete chain ip** *nazwaTabeli nazwaŁańcucha*

Dodawanie reguły, która odrzuca wszystkie zewnętrzne połączenia na port 80:
# **nft add rule ip** *nazwaTabeli nazwaŁańcucha* **tcp dport 80 drop**

Usunięcie reguły:

# **nft delete rule ip** *nazwaTabeli nazwaŁańcucha* **handle 3**

## Migracja iptables → nftables

# **USE=”nftables” emerge iptables** // Nie instaluje *nftables*, lecz jedynie jego obsługę
# **iptables-save >** *iptables-rules.txt*
# **iptables-restore-translate -f** *iptables-rules.txt* **>** *nftables-rules.txt*
# **cat** *nftables-rules.txt*
# **emerge nftables**
Reguły można wczytywać z pliku (tzw. *ruleset*):
# **nft -c -f** *nftables-rules.txt*
Uwaga: Zamiast wczytywać cały plik z regułami, można wczytać każdą regułę z osobna:
# **iptables-translate -A INPUT -p tcp --dport 22 -m conntrack --ctstate NEW -j ACCEPT**
*nft add rule ip filter INPUT tcp dport 22 ct state new counter accept* // Wynik transformacji

Jeśli wszystko w porządku, to:
# **/etc/init.d/iptables stop**
# **/etc/init.d/nftables start**
# **rc-update add nftables default**
# **emerge --unmerge iptables**
# **nano** */etc/portage/make.conf*
*USE=”... -iptables -xtables ...”*
# **emerge -uDN @world**

Dalsza obsługa:
# **/etc/init.d/nftables list** // Jakie są obecnie reguły

Poniższe, zapisuje obecne reguły do domyślnej lokalizacji: */var/lib/nftables/rules-save*
# **/etc/init.d/nftables save**
lub
# **rc-service nftables save**

Zapisujemy sobie osobistą kopię reguł:
# **cp** */var/lib/nftables/rules-save /home/użytkownik/.kopie/nftables-reguly.gotowe*
# **chmod u=rw,g=r,o=r** */home/użytkownik/.kopie/nftables-reguly.gotowe*
# **chown użytkownik:users** */home/użytkownik/.kopie/nftables-reguly.gotowe*

Poniższe, wczytuje reguły z pliku */var/lib/nftables/rules-save*:
# **/etc/init.d/nftables reload**

Ostatnia aktualizacja: 30 maja 2024.