* [nftables](https://wiki.debian.org/nftables)

[Translation(s)](https://wiki.debian.org/DebianWiki/EditorGuide#translation): English - [Русский](https://wiki.debian.org/ru/nftables)

nftables is a framework by the Netfilter Project that provides packet filtering, network address translation (NAT) and other packet mangling.

Two of the most common uses of nftables is to provide firewall support and NAT.

nftables is the default and recommended firewalling framework in Debian, and it replaces the old [iptables](https://wiki.debian.org/iptables) (and related) tools.

**Contents**

1. [Current status](https://wiki.debian.org/nftables#Current_status)
2. [Hints](https://wiki.debian.org/nftables#Hints)
   1. [Use firewalld](https://wiki.debian.org/nftables#Use_firewalld)
   2. [Reverting to legacy xtables](https://wiki.debian.org/nftables#Reverting_to_legacy_xtables)
   3. [nftables in Debian the easy way](https://wiki.debian.org/nftables#nftables_in_Debian_the_easy_way)
3. [FAQ](https://wiki.debian.org/nftables#FAQ)
   1. [What is nftables?](https://wiki.debian.org/nftables#What_is_nftables.3F)
   2. [Should I build a firewall using a nftables?](https://wiki.debian.org/nftables#Should_I_build_a_firewall_using_a_nftables.3F)
   3. [Should I replace an iptables firewall with a nftables one?](https://wiki.debian.org/nftables#Should_I_replace_an_iptables_firewall_with_a_nftables_one.3F)
   4. [Why a new framework?](https://wiki.debian.org/nftables#Why_a_new_framework.3F)
   5. [What are the major differences?](https://wiki.debian.org/nftables#What_are_the_major_differences.3F)
   6. [Should I mix nftables and iptables/ebtables/arptables rulesets?](https://wiki.debian.org/nftables#Should_I_mix_nftables_and_iptables.2Febtables.2Farptables_rulesets.3F)
   7. [I knew the iptables syntax. Is there a new syntax in nftables?](https://wiki.debian.org/nftables#I_knew_the_iptables_syntax.__Is_there_a_new_syntax_in_nftables.3F)
4. [new syntax](https://wiki.debian.org/nftables#new_syntax)
5. [external resources](https://wiki.debian.org/nftables#external_resources)

Current status

**NOTE: Debian Buster uses the nftables framework by default**.

Starting with Debian Buster, nf\_tables is the default backend when using iptables, by means of the iptables-nft layer (i.e, using iptables syntax with the nf\_tables kernel subsystem). This also affects ip6tables, arptables and ebtables.

Hints

Some hints folks might find interesting in some situations.

Use firewalld

You should consider using a wrapper instead of writing your own firewalling scripts. It is recommended to run [?](https://wiki.debian.org/firewalld)firewalld, which integrates pretty well into the system. See also <https://firewalld.org/>

The firewalld software takes control of all the firewalling setup in your system, so you don't have to know all the details of what is happening in the underground. There are many other system components that can integrate with firewalld, like [NetworkManager](https://wiki.debian.org/NetworkManager), libvirt, podman, fail2ban, docker, etc.

Reverting to legacy xtables

You can switch back and forth between iptables-nft and iptables-legacy by means of update-alternatives (same applies to arptables and ebtables).

The default starting with Debian Buster:

# update-alternatives --set iptables /usr/sbin/iptables-nft

# update-alternatives --set ip6tables /usr/sbin/ip6tables-nft

# update-alternatives --set arptables /usr/sbin/arptables-nft

# update-alternatives --set ebtables /usr/sbin/ebtables-nft

Switching to the legacy version:

# update-alternatives --set iptables /usr/sbin/iptables-legacy

# update-alternatives --set ip6tables /usr/sbin/ip6tables-legacy

# update-alternatives --set arptables /usr/sbin/arptables-legacy

# update-alternatives --set ebtables /usr/sbin/ebtables-legacy

nftables in Debian the easy way

If you want to enable a default firewall in Debian, follow these steps:

# aptitude install nftables

# systemctl enable nftables.service

This way, nftables is active at boot. By default, rules are located in **/etc/nftables.conf**.

To stop nftables from doing anything, just drop all the rules:

# nft flush ruleset

To prevent nftables from starting at boot:

# systemctl mask nftables.service

To uninstall it and purge any traces of nftables in your system:

# aptitude purge nftables

FAQ

What is nftables?

Is the new framework by the Netfilter Project, allowing you to perform packet filtering (firewalling), NAT, mangling and packet classification.

Should I build a firewall using a nftables?

Yes. Building new firewalls on top of iptables is discouraged.

Should I replace an iptables firewall with a nftables one?

Yes, nftables is the replacement for iptables. There are some tools in place to ease in this task.

Please read: <https://wiki.nftables.org/wiki-nftables/index.php/Moving_from_iptables_to_nftables>

Why a new framework?

The previous framework (iptables) has several problems hard to address, regarding scalability, performance, code maintenance, etc..

What are the major differences?

In iptables there are several tables (filter, nat) and chains (FORWARD, INPUT...) by default. In nftables, there are no default tables/chains.

Also, in iptables you only have one target per rule (-j ACCEPT, -j LOG ...). In nftables, you can perform several actions in one single rule.

nftables includes built-in data sets capabilities. In iptables this is not possible, and there is a separated tool: [?](https://wiki.debian.org/ipset)ipset.

In the iptables framework there are tools per family: iptables, ip6tables, arptables, ebtables. Now, nftables allows you to manage all families in one single CLI tool.

This new framework features a new linux kernel subsystem, known as nf\_tables. The new engine mechanism is inspired by BPF-like systems, with a set of basic expressions, which can be combined to build complex filtering rules.

Should I mix nftables and iptables/ebtables/arptables rulesets?

No, unless you know what you are doing.

I knew the iptables syntax. Is there a new syntax in nftables?

Yes, but the nftables one is better :-)

Help in migrating to nftables: <https://wiki.nftables.org/wiki-nftables/index.php/Moving_from_iptables_to_nftables>

new syntax

Create a basic IPv4/IPv6 dual-stack table:

# nft add table inet filter

Create a chain for input IPv4/IPv6 dual-stack traffic:

# nft add chain inet filter input { type filter hook input priority 0\; }

A rule to check that all is fine (IPv4/IPv6 dual-stack):

# nft add rule inet filter input counter accept

Show all the previous:

# nft list table inet filter

Flush rules in chain filter/input:

# nft flush chain inet filter input

Delete the chain filter/input:

# nft delete chain inet filter input

Delete the table filter:

# nft delete table inet filter

The family parameter is optional. The default is 'ip'. Other families are 'inet', 'ip6', 'arp', 'bridge' or 'netdev':

# nft add table ip6 filter

# nft add chain ip6 filter input

# nft add rule ip6 filter input counter accept

Debian ships example configurations in:

#/usr/share/doc/nftables/examples/

Count traffic on destination port tcp/22 (IPv4/IPv6 dual-stack):

# nft add rule inet filter input tcp dport 22 counter

Count and accept traffic in 80/tcp and 443/tcp in new and established state (IPv4/IPv6 dual-stack):

# nft add rule inet filter input tcp dport {80, 443} ct state new,established counter accept

external resources

Check out the official nftables wiki: <http://wiki.nftables.org/>