iptables

From Wikipedia, the free encyclopedia

[Jump to navigation](https://en.wikipedia.org/wiki/Iptables#mw-head)[Jump to search](https://en.wikipedia.org/wiki/Iptables#searchInput)

|  |  |
| --- | --- |
|  | This article includes a [list of references](https://en.wikipedia.org/wiki/Wikipedia:Citing_sources), but **its sources remain unclear because it has insufficient**[**inline citations**](https://en.wikipedia.org/wiki/Wikipedia:Citing_sources#Inline_citations). Please help to [improve](https://en.wikipedia.org/wiki/Wikipedia:WikiProject_Fact_and_Reference_Check) this article by [introducing](https://en.wikipedia.org/wiki/Wikipedia:When_to_cite) more precise citations. *(April 2015) (*[*Learn how and when to remove this template message*](https://en.wikipedia.org/wiki/Help:Maintenance_template_removal)*)* |

|  |  |
| --- | --- |
| **iptables** | |
| [**Original author(s)**](https://en.wikipedia.org/wiki/Software_developer) | [Rusty Russell](https://en.wikipedia.org/wiki/Rusty_Russell) |
| [**Developer(s)**](https://en.wikipedia.org/wiki/Software_developer) | Netfilter Core Team |
| **Initial release** | 1998 |
| [**Stable release**](https://en.wikipedia.org/wiki/Software_release_life_cycle) | 1.8.5 / June 3, 2020; 3 months ago[[1]](https://en.wikipedia.org/wiki/Iptables#cite_note-1) |
| [**Repository**](https://en.wikipedia.org/wiki/Repository_(version_control)) | * [git.netfilter.org/iptables](git://git.netfilter.org/iptables)   [Edit this at Wikidata](https://www.wikidata.org/wiki/Q284612#P1324) |
| **Written in** | [C](https://en.wikipedia.org/wiki/C_(programming_language)) |
| [**Operating system**](https://en.wikipedia.org/wiki/Operating_system) | [Linux](https://en.wikipedia.org/wiki/Linux) |
| [**Platform**](https://en.wikipedia.org/wiki/Computing_platform) | [Netfilter](https://en.wikipedia.org/wiki/Netfilter) |
| [**Type**](https://en.wikipedia.org/wiki/Software_categories#Categorization_approaches) | [Packet filtering](https://en.wikipedia.org/wiki/Network_packet) |
| [**License**](https://en.wikipedia.org/wiki/Software_license) | [GPL](https://en.wikipedia.org/wiki/GPL) |
| **Website** | [www.netfilter.org](https://www.netfilter.org/) |

**iptables** is a [user-space](https://en.wikipedia.org/wiki/User_space) utility program that allows a [system administrator](https://en.wikipedia.org/wiki/System_administrator) to configure the [IP packet filter rules](https://en.wikipedia.org/wiki/Packet_filter) of the [Linux kernel](https://en.wikipedia.org/wiki/Linux_kernel) [firewall](https://en.wikipedia.org/wiki/Firewall_(computing)), implemented as different [Netfilter](https://en.wikipedia.org/wiki/Netfilter) modules. The filters are organized in different tables, which contain chains of rules for how to treat network traffic packets. Different kernel modules and programs are currently used for different protocols; *iptables* applies to IPv4, *ip6tables* to IPv6, *arptables* to [ARP](https://en.wikipedia.org/wiki/Address_Resolution_Protocol), and *ebtables* to [Ethernet frames](https://en.wikipedia.org/wiki/Ethernet_frame).

iptables requires elevated privileges to operate and must be executed by user [root](https://en.wikipedia.org/wiki/Superuser), otherwise it fails to function. On most Linux systems, iptables is installed as /usr/sbin/iptables and documented in its [man pages](https://en.wikipedia.org/wiki/Man_page), which can be opened using man iptables when installed. It may also be found in /sbin/iptables, but since iptables is more like a service rather than an "essential binary", the preferred location remains /usr/sbin.

The term *iptables* is also commonly used to inclusively refer to the kernel-level components. *x\_tables* is the name of the kernel module carrying the shared code portion used by all four modules that also provides the API used for extensions; subsequently, *Xtables* is more or less used to refer to the entire firewall (v4, v6, arp, and eb) architecture.

iptables superseded [ipchains](https://en.wikipedia.org/wiki/Ipchains); and the successor of iptables is [nftables](https://en.wikipedia.org/wiki/Nftables), which was released on 19 January 2014[[2]](https://en.wikipedia.org/wiki/Iptables#cite_note-2) and was merged into the [Linux kernel mainline](https://en.wikipedia.org/wiki/Linux_kernel_mainline) in kernel version 3.13.

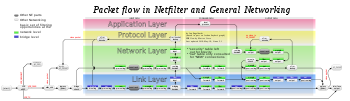


**Contents**

* [1Overview](https://en.wikipedia.org/wiki/Iptables#Overview)
* [2Userspace utilities](https://en.wikipedia.org/wiki/Iptables#Userspace_utilities)
  + [2.1Front-ends](https://en.wikipedia.org/wiki/Iptables#Front-ends)
  + [2.2Other notable tools](https://en.wikipedia.org/wiki/Iptables#Other_notable_tools)
* [3See also](https://en.wikipedia.org/wiki/Iptables#See_also)
* [4References](https://en.wikipedia.org/wiki/Iptables#References)
* [5Literature](https://en.wikipedia.org/wiki/Iptables#Literature)
* [6External links](https://en.wikipedia.org/wiki/Iptables#External_links)

Overview[[edit](https://en.wikipedia.org/w/index.php?title=Iptables&action=edit&section=1)]

Xtables allows the [system administrator](https://en.wikipedia.org/wiki/System_administrator) to define *tables* containing *chains* of *rules* for the treatment of packets. Each table is associated with a [different kind of packet processing](https://en.wikipedia.org/wiki/Netfilter#iptables). Packets are processed by sequentially traversing the rules in chains. A rule in a chain can cause a goto or jump to another chain, and this can be repeated to whatever level of nesting is desired. (A jump is like a “call”, i.e. the point that was jumped from is remembered.) Every network packet arriving at or leaving from the computer traverses at least one chain.

[](https://en.wikipedia.org/wiki/File:Netfilter-packet-flow.svg)

Packet flow paths. Packets start at a given box and will flow along a certain path, depending on the circumstances.

The origin of the packet determines which chain it traverses initially. There are five *predefined chains* (mapping to the five available Netfilter hooks), though a table may not have all chains. Predefined chains have a *policy*, for example DROP, which is applied to the packet if it reaches the end of the chain. The system administrator can create as many other chains as desired. These chains have no policy; if a packet reaches the end of the chain it is returned to the chain which called it. A chain may be empty.

* PREROUTING: Packets will enter this chain before a routing decision is made.
* INPUT: Packet is going to be locally delivered. It does not have anything to do with processes having an opened socket; local delivery is controlled by the "local-delivery" routing table: ip route show table local.
* FORWARD: All packets that have been routed and were not for local delivery will traverse this chain.
* OUTPUT: Packets sent from the machine itself will be visiting this chain.
* POSTROUTING: Routing decision has been made. Packets enter this chain just before handing them off to the hardware.

A chain does not exist by itself; it belongs to a *table*. There are three tables: *nat*, *filter*, and *mangle*. Unless preceded by the option *-t*, an iptables command concerns the *filter* table by default. For example, the command iptables -L -v -n, which shows some chains and their rules, is equivalent to iptables -t filter -L -v -n. To show chains of table *nat*, use the command iptables -t nat -L -v -n

Each rule in a chain contains the specification of which packets it matches. It may also contain a *target* (used for extensions) or *verdict* (one of the built-in decisions). As a packet traverses a chain, each rule in turn is examined. If a rule does not match the packet, the packet is passed to the next rule. If a rule does match the packet, the rule takes the action indicated by the target/verdict, which may result in the packet being allowed to continue along the chain or may not. Matches make up the large part of rulesets, as they contain the conditions packets are tested for. These can happen for about any layer in the [OSI](https://en.wikipedia.org/wiki/OSI_model) model, as with e.g. the --mac-source and -p tcp --dport parameters, and there are also protocol-independent matches, such as -m time.

The packet continues to traverse the chain until either

1. a rule matches the packet and decides the ultimate fate of the packet, for example by calling one of the ACCEPT or DROP, or a module returning such an ultimate fate; or
2. a rule calls the RETURN verdict, in which case processing returns to the calling chain; or
3. the end of the chain is reached; traversal either continues in the parent chain (as if RETURN was used), or the base chain policy, which is an ultimate fate, is used.

Targets also return a verdict like ACCEPT (NAT modules will do this) or DROP (e.g. the REJECT module), but may also imply CONTINUE (e.g. the LOG module; CONTINUE is an internal name) to continue with the next rule as if no target/verdict was specified at all.

Userspace utilities[[edit](https://en.wikipedia.org/w/index.php?title=Iptables&action=edit&section=2)]

**Front-ends**[[edit](https://en.wikipedia.org/w/index.php?title=Iptables&action=edit&section=3)]

There are numerous third-party software applications for iptables that try to facilitate setting up rules. Front-ends in [textual](https://en.wikipedia.org/wiki/Ncurses) or graphical fashion allow users to click-generate simple rulesets; scripts usually refer to [shell scripts](https://en.wikipedia.org/wiki/Unix_shell) (but other scripting languages are possible too) that call iptables or (the faster) iptables-restore with a set of predefined rules, or rules expanded from a template with the help of a simple configuration file. Linux distributions commonly employ the latter scheme of using templates. Such a template-based approach is practically a limited form of a rule generator, and such generators also exist in standalone fashion, for example, as PHP web pages.

Such front-ends, generators and scripts are often limited by their built-in template systems and where the templates offer substitution spots for user-defined rules. Also, the generated rules are generally not optimized for the particular firewalling effect the user wishes, as doing so will likely increase the maintenance cost for the developer. Users who reasonably understand iptables and want their ruleset optimized are advised to construct their own ruleset.

**Other notable tools**[[edit](https://en.wikipedia.org/w/index.php?title=Iptables&action=edit&section=4)]

* [FireHOL](https://en.wikipedia.org/wiki/FireHOL) – a shell script wrapping iptables with an easy-to-understand plain-text configuration file
* [NuFW](https://en.wikipedia.org/wiki/NuFW) – an authenticating firewall extension to Netfilter
* [Shorewall](https://en.wikipedia.org/wiki/Shorewall) – a gateway/firewall configuration tool, making it possible to use easier rules and have them mapped to iptables

See also[[edit](https://en.wikipedia.org/w/index.php?title=Iptables&action=edit&section=5)]

* ******[***Free and open-source software portal***](https://en.wikipedia.org/wiki/Portal:Free_and_open-source_software)
* ***[icon](https://en.wikipedia.org/wiki/File:NewTux.svg)***[***Linux portal***](https://en.wikipedia.org/wiki/Portal:Linux)
* [NPF (firewall)](https://en.wikipedia.org/wiki/NPF_(firewall))
* [PF (firewall)](https://en.wikipedia.org/wiki/PF_(firewall))
* [ipfirewall](https://en.wikipedia.org/wiki/Ipfirewall) (ipfw)
* [ipfilter](https://en.wikipedia.org/wiki/Ipfilter)
* [XDP](https://en.wikipedia.org/wiki/Express_Data_Path)
* [ipchains](https://en.wikipedia.org/wiki/Ipchains)

References[[edit](https://en.wikipedia.org/w/index.php?title=Iptables&action=edit&section=6)]

* 1. [**^**](https://en.wikipedia.org/wiki/Iptables#cite_ref-1) [*"News of the netfilter/iptables project"*](https://netfilter.org/news.html)*. netfilter.org. 2020-06-12. Retrieved 2020-06-14.*
  2. [**^**](https://en.wikipedia.org/wiki/Iptables#cite_ref-2) [*"Linux 3.13, Section 1.2. nftables, the successor of iptables"*](http://kernelnewbies.org/Linux_3.13#head-f628a9c41d7ec091f7a62db6a49b8da50659ec88)*. kernelnewbies.org. 2014-01-19. Retrieved 2014-01-20.*

Literature[[edit](https://en.wikipedia.org/w/index.php?title=Iptables&action=edit&section=7)]

* *Gregor N. Purdy (25 August 2004).*[*Linux iptables Pocket Reference: Firewalls, NAT & Accounting*](https://books.google.com/books?id=CCk6xpGeYjwC)*. O'Reilly Media, Inc.*[*ISBN*](https://en.wikipedia.org/wiki/ISBN_(identifier))[*978-1-4493-7898-1*](https://en.wikipedia.org/wiki/Special:BookSources/978-1-4493-7898-1)*.*

External links[[edit](https://en.wikipedia.org/w/index.php?title=Iptables&action=edit&section=8)]

**Iptables**at Wikipedia's [sister projects](https://en.wikipedia.org/wiki/Wikipedia:Wikimedia_sister_projects)

* [News](https://en.wikinews.org/wiki/Special:Search/Iptables) from Wikinews
* [Textbooks](https://en.wikibooks.org/wiki/Communication_Networks/IP_Tables) from Wikibooks
* [The netfilter/iptables project Web page](http://www.netfilter.org/)
* [*"iptables"*](http://freecode.com/projects/iptables/)*.*[*Freecode*](https://en.wikipedia.org/wiki/Freecode)*.*
* [The netfilter/iptables documentation page](http://www.netfilter.org/documentation/index.html) (outdated)[[*clarification needed*](https://en.wikipedia.org/wiki/Wikipedia:Please_clarify)]
* [Detecting and deceiving network scans](http://inai.de/documents/Chaostables.pdf) – countermeasures against nmap
* [The IPTables ManPage for syntax help](http://ipset.netfilter.org/iptables.man.html)
* [Iptables Tutorial 1.2.2 by Oskar Andreasson](https://www.frozentux.net/iptables-tutorial/iptables-tutorial.html)
* [IPTABLES: The Default Linux Firewall](http://www.linux-india.org/linux-firewall/)
* [Acceleration of iptables Linux Packet Filtering using GPGPU](https://www.researchgate.net/profile/Mahmood_Ahmadi/publication/261550748_Acceleration_of_IPTABLES_Linux_Packet_Filtering_using_GPGPU/links/0c96053494b348cf52000000)