Options Used in iptables Commands

Rules that allow packets to be filtered by the kernel are put in place by running the iptables command. When using the iptables command, you must specify the following options:

- *Packet Type* — This dictates what type of packets the command filters.
- *Packet Source or Destination* — This dictates what packets the command filters based on the source or destination of the packet.
- *Target* — This dictates what action is taken on packets matching the above criteria.

The options used with given iptables rule must be grouped logically, based on the purpose and conditions of the overall rule, in order for the rule to be valid.

Tables

A powerful aspect of iptables is that multiple tables can be used to decide the fate of a particular packet, depending upon the type of packet being monitored and what is to be done with the packet. Thanks to the extensible nature of iptables, specialized tables can be created and stored in the /etc/modules/*<kernel-version>*/kernel/net/ipv4/netfilter/ directory to meet specific goals. Think of iptables as being able to run multiple sets of ipchains rules in defined chains, with each set fulfilling a particular role.

The default table, named filter, contains the standard built-in INPUT, OUTPUT, and FORWARD chains. This is somewhat similar to the standard chains in use with ipchains. However, by default, iptables also includes two additional tables that perform specific packet filtering jobs. The nat table can be used to modify the source and destination addresses recorded in packets, and the mangle table allows you to alter packets in specialized ways.

Each table contains default chains that perform necessary tasks based on the purpose of the table, but you can easily set up new chains in each of the tables.

Structure

Many iptables commands have the following structure:

```
iptables [-t <table-name>] <command> <chain-name> <parameter-1> \
         <option-1> <parameter-n> <option-n>
```

In this example, the *<table-name>* option allows the user to select a table other than the default `filter` table to use with the command. The *<command>* option is the center of the command, dictating a specific action to perform, such as appending or deleting a rule from a particular chain, which is specified by the *<chain-name>* option. Following the *<chain-name>* are pairs of parameters and options that actually define the way the rule will work and what will happen when a packet matches the rule.

When looking at the structure of an `iptables` command, it is important to remember that, unlike most other commands, the length and complexity of an `iptables` command can change based on its purpose. A simple command to remove a rule from a chain can be very short, while a command designed to filter packets from a particular subnet using a variety of specific parameters and options can be rather lengthy. When creating `iptables` commands it is helpful to recognize that some parameters and options may create the need for other parameters and options to further specify the previous option's request. In order to construct a valid rule, this must continue until every parameter and option that requires another set of options is satisfied.

Type `iptables -h` to see a comprehensive list of `iptables` command structures.

Commands

Commands tell `iptables` to perform a specific action. Only one command is allowed per `iptables` command string. With the exception of the help command, all commands are written in upper-case characters.

The `iptables` commands are as follows:

- `-A` — Appends the `iptables` rule to the end of the specified chain. This is the command used to simply add a rule when rule order in the chain does not matter.
- `-C` — Checks a particular rule before adding it to the user-specified chain. This command can help you construct complicated `iptables` rules by prompting you for additional parameters and options.
- `-D` — Deletes a rule in a particular chain by number (such as 5 for the fifth rule in a chain). You can also type the entire rule, and `iptables` will delete the rule in the chain that matches it.
- `-E` — Renames a user-defined chain. This does not affect the structure of the table. Rather, it just saves you the trouble of deleting the chain, creating it under the new name, and reconfiguring all of your rules for that chain.
- `-F` — Flushes the selected chain, which effectively deletes every rule in the the chain. If no chain is specified, this command flushes every rule from every chain.
- `-h` — Provides a list of helpful command structures, as well as a quick summary of command parameters and options.
- `-I` — Inserts a rule in a chain at a particular point. Assign a number to the rule to be inserted and `iptables` will put it there. If no number is specified, `iptables` will place your command at the top of the rule list.

Caution

Be aware of which option (`-A` or `-I`) you are using when adding a rule. The order of the rules can be very important when determining if a particular packet applies to one rule or another. Make sure when adding a rule to the beginning or end of the chain that it does not affect other rules in that chain.

- `-L` — Lists all of the rules in the chain specified after the command. To list all rules in all chains in the default `filter` table, do not specify a chain or table. Otherwise, the following syntax should be used to list the rules in a specific chain in a particular table:

  ```
  iptables -L <chain-name> -t <table-name>
  ```

  Powerful options for the `-L` command that provide rule numbers and allow more verbose rule descriptions, among others, are described in the Section called *Listing Options*.

- `-N` — Creates a new chain with a user-specified name.
- `-P` — Sets the default policy for a particular chain, so that when packets traverse an entire chain without matching a rule, they will be sent on to a particular target, such as ACCEPT or DROP.
- `-R` — Replaces a rule in a particular chain. You must use a rule's number after the chain's name to replace that rule. The first rule in a chain relates to rule number `1`.
- `-X` — Deletes a user-specified chain. Deleting a built-in chain for any table is not allowed.
- `-Z` — Zeros the byte and packet counters in all chains for a particular table.

Parameters

Once certain `iptables` commands are specified, including those used to add, append, delete, insert, or replace rules within a particular chain, parameters are required to begin the construction of the packet filtering rule.

- `-c` — Resets the counters for a particular rule. This parameter accepts the PKTS and BYTES options to specify what counter to reset.
- `-d` — Sets the destination hostname, IP address, or network of a packet that will match the rule. When matching a network, you can use two different methods for signifying the netmask, such as `192.168.0.0/255.255.255.0` or `192.168.0.0/24`.
- `-f` — Applies this rule only to fragmented packets.

  By using the `!` option after this parameter, only unfragmented packets will be matched.

- `-i` — Sets the incoming network interface, such as `eth0` or `ppp0`, to use with a particular rule. With `iptables`, this optional parameter may only be used with the INPUT and FORWARD chains when used with the `filter` table and the PREROUTING chain with the `nat` and `mangle` tables.

  This parameter features several useful options that may be used before specifying the name of an interface:

  - `!` — Tells this parameter not to match, meaning that any specified interfaces are specifically excluded from this rule.
  - `+` — A wildcard character used to match all interfaces which match a particular string. For example, the parameter `-i eth+` would apply this rule to any Ethernet interfaces on your system but exclude any other interfaces, such as `ppp0`.

  If the `-i` parameter is used but no interface is specified, then every interface is affected by the rule.

- `-j` — Tells `iptables` to jump to a particular target when a packet matches a particular rule. Valid targets to be used after the `-j` option include the standard options, `ACCEPT`, `DROP`, `QUEUE`, and `RETURN`, as well as extended options that are available through modules loaded by default with the Red Hat Linux `iptables` RPM package, such as `LOG`, `MARK`, and `REJECT`, among others. See the `iptables` man page for more information on these and other targets, including rules regarding their use.

  You may also direct a packet matching this rule to a user-defined chain outside of the current chain. This allows you to apply other rules against this packet, further filtering it with more specific criteria.

  If no target is specified, the packet moves past the rule with no action taken. However, the counter for this rule is still increased by one, as the packet matched the specified rule.

- `-o` — Sets the outgoing network interface for a particular rule, and may only be used with OUTPUT and FORWARD chains in the `filter` table and the POSTROUTING chain in the `nat` and `mangle` tables. This parameter's options are the same as those of the incoming network interface parameter (`-i`).

- `-p` — Sets the IP protocol for the rule, which can be either `icmp`, `tcp`, `udp`, or `all`, to match every supported protocol. In addition, lesser used protocols listed in `/etc/protocols` may also be used. If this option is omitted when creating a rule, the `all` option is the default.

- `-s` — Sets the source for a particular packet, using the same syntax as the destination (`-d`) parameter.

Match Options

Different network protocols provide specialized matching options which may be set in specific ways to match a particular packet using that protocol. Of course, the protocol must first be specified in the `iptables` command, such as using `-p tcp <protocol-name>`, to make the options for that protocol available.

TCP Protocol

These match options are available for the TCP protocol (`-p tcp`):

- `--dport` — Sets the destination port for the packet. You can use either a network service name (such as `www` or `smtp`), port number, or range of port numbers to configure this option. To browse the names and aliases of network services and the port numbers they use, view the `/etc/services` file. You can also use `--destination-port` to specify this match option.

  To specify a specific range of port numbers, separate the two numbers with a colon (`:`), such as `-p tcp --dport 3000:3200`. The largest valid range is `0:65535`.

  You may also use an exclamation point character (`!`) as a flag after the `--dport` option to tell `iptables` to match all packets which *do not* use that network service or port.

- `--sport` — Sets the source port of the packet, using the same options as `--dport`. You can also use `--source-port` to specify this match option.

- `--syn` — Applies to all TCP packets designed to initiate communication, commonly called *SYN packets*. Any packets that carry a data payload are not touched. Placing an exclamation point character (`!`) as a flag after the `--syn` option causes all non-SYN packets to be matched.

- `--tcp-flags` — Allows TCP packets with specific bits, or flags, set to be matched with a rule. The `--tcp-flags` match option accepts two parameters after it, which are flags for the various bits arranged in a comma-separated list. The first parameter is the mask, which sets the flags to be examined on the packet. The second parameter refers to the flags that must be set in the packet to make a match. The possible flags are `ACK`, `FIN`, `PSH`, `RST`, `SYN`, and `URG`. In addition, `ALL` and `NONE` can also be used to match every flag or none of them.

  For example, an `iptables` rule which contains `-p tcp --tcp-flags ACK,FIN,SYN SYN` will only match TCP packets that have the SYN flag set and the ACK and FIN flags unset.

  Like many other options, using the exclamation point character (`!`) after `--tcp-flags` reverses the effect of the match option, so that the second parameter's flags must not be set in order to match.

- `--tcp-option` — Attempts to match with TCP-specific options that can be set within a particular packet. This match option can also be reversed with the exclamation point character (`!`).

UDP Protocol

These match options are available for the UDP protocol (`-p udp`):

- `--dport` — Specifies the destination port of the UDP packet, using the service name, port number, or range of port numbers. The `--destination-port` match option may be used instead of `--dport`. See the `--dport` match option in the Section called *TCP*

Protocol for various ways to use this option.

- `--sport` — Specifies the source port of the UDP packet, using the service name, port number, or range of port numbers. The `--source-port` match option may be used instead of `--sport`. See the `--dport` match option in the Section called *TCP Protocol* for various ways to use this option.

ICMP Protocol

Packets using the Internet Control Message Protocol (ICMP) can be matched using the following option when `-p icmp` is specified:

- `--icmp-type` — Sets the name or number of the ICMP type to match with the rule. A list of valid ICMP names can be seen by typing the `iptables -p icmp -h` command.

Modules with Additional Match Options

Additional match options are also available through modules loaded when the `iptables` command calls them. To use a match option module, you must load the module by name by including `-m <module-name>` in the `iptables` command.

A large number of modules are available by default. It is even possible to create your own modules to provide additional match option functionality. Many modules exist, but only the most popular ones are discussed here.

The `limit` module allows you to place a limit on how many packets will be matched to a particular rule. This is especially beneficial when logging rule matches so that a flood of matching packets will not fill up your logs with repetitive messages or use too many system resources.

- `--limit` — Sets the number of matches for a particular range of time, specified with a number and time modifier arranged in a `<number>/<time>` format. For example, using `--limit 5/hour` only lets a rule match five times in a single hour.

  If a number and time modifier are not used, the default value of `3/hour` is assumed.

- `--limit-burst` — Sets a limit on the number of packets able to match a rule at one time. This option should be used in conjunction with the `--limit` option, and it accepts a number to set the burst threshold.

  If no number is specified, only five packets are initially able to match the rule.

The `state` module, which uses the `--state` match option, can match a packet with these particular connection states:

- `ESTABLISHED` — The matching packet is associated with other packets in an established connection.
- `INVALID` — The matching packet cannot be tied to a known connection.

- `NEW` — The matching packet is either creating a new connection or is part of a two-way connection not previously seen.
- `RELATED` — The matching packet is starting a new connection related in some way to an existing connection.

These connection states can be used in combination with one another by separating them with commas, such as `-m state --state INVALID,NEW`.

To specifically match a hardware MAC address of an Ethernet device, use the `mac` module, which accepts `--mac-source` plus a MAC address as an option. To exclude a MAC address from a rule, place an exclamation point (`!`) after the `--mac-source` match option.

To view other match options available through modules, see the `iptables` man page.

Target Options

Once a packet has matched a particular rule, the rule can direct the packet to a number of different targets that decide its fate and, possibly, take additional actions, such as logging the action. Additionally, each chain has a default target, which is used if none of the rules on that chain match a packet or if none of the rules which match the packet specify a target.

There are only a few standard targets available to decide what happens with the packet:

- *<user-defined-chain>* — The name of a previously created and defined chain within this table with rules that will be checked against this packet, in addition to any other rules in any other chains that must be checked against this packet.
- `ACCEPT` — Allows the packet to successfully move on to its destination or another chain.
- `DROP` — Drops the packet without responding to the requester. The system that sent the packet is not notified of the failure. The packet is simply removed from the rule checking the chain and discarded.
- `QUEUE` — The packet is queued for handling by a user-space application.
- `RETURN` — Stops checking the packet against rules in the current chain. If the packet with a `RETURN` target matches a rule in a chain called from another chain, the packet is returned to the first chain to resume rule checking where it left off. If the `RETURN` rule is used on a built-in chain and the packet cannot move up to its previous chain, the default target for the current chain decides what action to take.

In addition to these standard targets, various other targets may be used with extensions called *target modules*. For more information about match option modules, see the Section called *Modules with Additional Match Options*.

There are many extended target modules, most of which only apply to specific tables or situations. A couple of the most popular target modules included by default in Red Hat Linux are:

- `LOG` Logs all packets that match this rule. Since the packets are logged by the kernel, the `/etc/syslog.conf` file determines where these log entries are written. By default, they are placed in the `/var/log/messages` file.

  Various options can be used after the `LOG` target to specify the way in which logging occurs:

  - `--log-level` — Sets the priority level a of logging event. A list of priority levels can be found in the `syslog.conf` man page.
  - `--log-ip-options` — Any options set in the header of a IP packet is logged.
  - `--log-prefix` — Places a string before the log line when it is written. Accepts up to 29 characters after the `--log-prefix` option. This is useful for writing syslog filters for use in conjunction with packet logging.
  - `--log-tcp-options` — Any options set in the header of a TCP packet is logged
  - `--log-tcp-sequence` — Writes the TCP sequence number for the packet in the log.
- `REJECT` — Sends an error packet back to the system which sent the packet, and then drops the packet. This target is useful if you would like to notify the system sending the matching packet of the problem.

  The `REJECT` target accepts a `--reject-with` *<type>* option which allows more detailed information to be sent with the error packet. The message `port-unreachable` is the default *<type>* error given if no other option is used. For a full list of *<type>* options that can be used, see the `iptables` man page.

Other target extensions, including several that are useful with masquerading using the `nat` table or with packet alteration using the `mangle` table, can be found in the `iptables` man page.

Listing Options

The default list command, `iptables -L`, provides a very basic overview of the default filter table's current chains. Additional options provide more information and arrange that information in specific ways:

- `-v` — Display verbose output, such as the number of packets and bytes each chain has seen, the number of packets and bytes each rule has matched, and which interfaces apply to a particular rule.
- `-x` — Expands numbers into their exact values. On a busy system, the number of packets and bytes seen by a particular chain or rule may be abbreviated using `K` (thousands), `M` (millions), and `G` (billions) at the end of the number. This option forces the full number to be displayed.
- `-n` — Displays IP addresses and port numbers in numeric format, rather than the default hostname and network service format.

- `--line-numbers` — Lists rules in each chain next to their numeric order in the chain. This option is useful when attempting to delete a specific rule in a chain, or to locate where to insert a rule within a chain.