9.3. Commands

In this section we will cover all the different commands and what can be done with them. The command tells **iptables** what to do with the rest of the rule that we send to the parser. Normally we would want either to add or delete something in some table or another. The following commands are available to iptables:

Table 9-2. Commands

Command	**-A**, **--append**
Example	**iptables -A INPUT ...**
Explanation	This command appends the rule to the end of the chain. The rule will in other words always be put last in the rule-set and hence be checked last, unless you append more rules later on.
Command	**-D**, **--delete**
Example	**iptables -D INPUT --dport 80 -j DROP**, **iptables -D INPUT 1**
Explanation	This command deletes a rule in a chain. This could be done in two ways; either by entering the whole rule to match (as in the first example), or by specifying the rule number that you want to match. If you use the first method, your entry must match the entry in the chain exactly. If you use the second method, you must match the number of the rule you want to delete. The rules are numbered from the top of each chain, starting with number 1.
Command	**-R**, **--replace**
Example	**iptables -R INPUT 1 -s 192.168.0.1 -j DROP**
Explanation	This command replaces the old entry at the specified line. It works in the same way as the **--delete** command, but instead of totally deleting the entry, it will replace it with a new entry. The main use for this might be while you're experimenting with iptables.
Command	**-I**, **--insert**
Example	**iptables -I INPUT 1 --dport 80 -j ACCEPT**
Explanation	Insert a rule somewhere in a chain. The rule is inserted as the actual number that we specify. In other words, the above example would be inserted as rule 1 in the INPUT chain, and hence from now on it would be the very first rule in the chain.
Command	**-L**, **--list**
Example	**iptables -L INPUT**
Explanation	This command lists all the entries in the specified chain. In the above case, we would list all the entries in the INPUT chain. It's also legal to not specify any chain at all. In the last case, the command would list all the chains in the specified table (To specify a table, see the *Tables* section). The exact output is affected by other options sent to the parser, for example the **-n** and **-v** options, etc.

Command	**-F, --flush**
Example	**iptables -F INPUT**
Explanation	This command flushes all rules from the specified chain and is equivalent to deleting each rule one by one, but is quite a bit faster. The command can be used without options, and will then delete all rules in all chains within the specified table.
Command	**-Z, --zero**
Example	**iptables -Z INPUT**
Explanation	This command tells the program to zero all counters in a specific chain, or in all chains. If you have used the **-v** option with the **-L** command, you have probably seen the packet counter at the beginning of each field. To zero this packet counter, use the **-Z** option. This option works the same as **-L**, except that **-Z** won't list the rules. If **-L** and **-Z** is used together (which is legal), the chains will first be listed, and then the packet counters are zeroed.
Command	**-N, --new-chain**
Example	**iptables -N allowed**
Explanation	This command tells the kernel to create a new chain of the specified name in the specified table. In the above example we create a chain called **allowed**. Note that there must not already be a chain or target of the same name.
Command	**-X, --delete-chain**
Example	**iptables -X allowed**
Explanation	This command deletes the specified chain from the table. For this command to work, there must be no rules that refer to the chain that is to be deleted. In other words, you would have to replace or delete all rules referring to the chain before actually deleting the chain. If this command is used without any options, all chains but those built in to the specified table will be deleted.
Command	**-P, --policy**
Example	**iptables -P INPUT DROP**
Explanation	This command tells the kernel to set a specified default target, or policy, on a chain. All packets that don't match any rule will then be forced to use the policy of the chain. Legal targets are **DROP** and **ACCEPT** (There might be more, mail me if so).
Command	**-E, --rename-chain**
Example	**iptables -E allowed disallowed**
Explanation	The **-E** command tells **iptables** to change the first name of a chain, to the second name. In the example above we would, in other words, change the name of the chain from `allowed` to `disallowed`. Note that this will not affect the actual way the table will work. It is, in other words, just a cosmetic change to the table.

You should always enter a complete command line, unless you just want to list the built-in help for **iptables** or get the version of the command. To get the version, use the **-v** option and to get the help message, use the **-h** option. As usual, in other words. Next comes a few options that can be used with various different commands. Note that we tell you with which commands the options can be used and what effect they will have. Also note that we do not include any options here that affect rules or matches. Instead, we'll take a look at matches and targets in a later section of this chapter.

Table 9-3. Options

Option	**-v**, **--verbose**
Commands used with	**--list**, **--append**, **--insert**, **--delete**, **--replace**
Explanation	This command gives verbose output and is mainly used together with the **--list** command. If used together with the **--list** command, it outputs the interface address, rule options and TOS masks. The **--list** command will also include a bytes and packet counter for each rule, if the **--verbose** option is set. These counters uses the K (x1000), M (x1,000,000) and G (x1,000,000,000) multipliers. To overrule this and get exact output, you can use the **-x** option, described later. If this option is used with the **--append**, **--insert**, **--delete** or **--replace** commands, the program will output detailed information on how the rule was interpreted and whether it was inserted correctly, etc.
Option	**-x**, **--exact**
Commands used with	**--list**
Explanation	This option expands the numerics. The output from **--list** will in other words not contain the K, M or G multipliers. Instead we will get an exact output from the packet and byte counters of how many packets and bytes that have matched the rule in question. Note that this option is only usable in the **--list** command and isn't really relevant for any of the other commands.
Option	**-n**, **--numeric**
Commands used with	**--list**
Explanation	This option tells iptables to output numerical values. IP addresses and port numbers will be printed by using their numerical values and not host-names, network names or application names. This option is only applicable to the **--list** command. This option overrides the default of resolving all numerics to hosts and names, where this is possible.
Option	**--line-numbers**
Commands used with	**--list**
Explanation	The **--line-numbers** command, together with the **--list** command, is used to output line numbers. Using this option, each rule is output with its number. It could be convenient to know which rule has which number when inserting rules. This option only works with the **--list** command.
Option	**-c**, **--set-counters**
Commands used with	**--insert**, **--append**, **--replace**
Explanation	This option is used when creating a rule or modifying it in some way. We can then use the option to initialize the packet and byte counters for the rule. The syntax would be something like **--set-counters 20 4000**, which would tell the kernel to set the packet counter to 20 and byte counter to 4000.
Option	**--modprobe**
Commands	All

used with	
Explanation	The **--modprobe** option is used to tell **iptables** which module to use when probing for modules or adding them to the kernel. It could be used if your **modprobe** command is not somewhere in the search path etc. In such cases, it might be necessary to specify this option so the program knows what to do in case a needed module is not loaded. This option can be used with all commands.

Prev
Tables
Home
Up
Next
Iptables matches