iptables

(Linux firewall tool)

netfilter and **iptables:**

iptables is a Linux firewall tool that controls network traffic by defining ***rules*** in ***tables*** and ***chains***. It's part of the **netfilter** framework in the Linux kernel.

netfilter: is the Linux kernel's network packet processing subsystem.

iptables: is the command used to configure it.

Concepts:

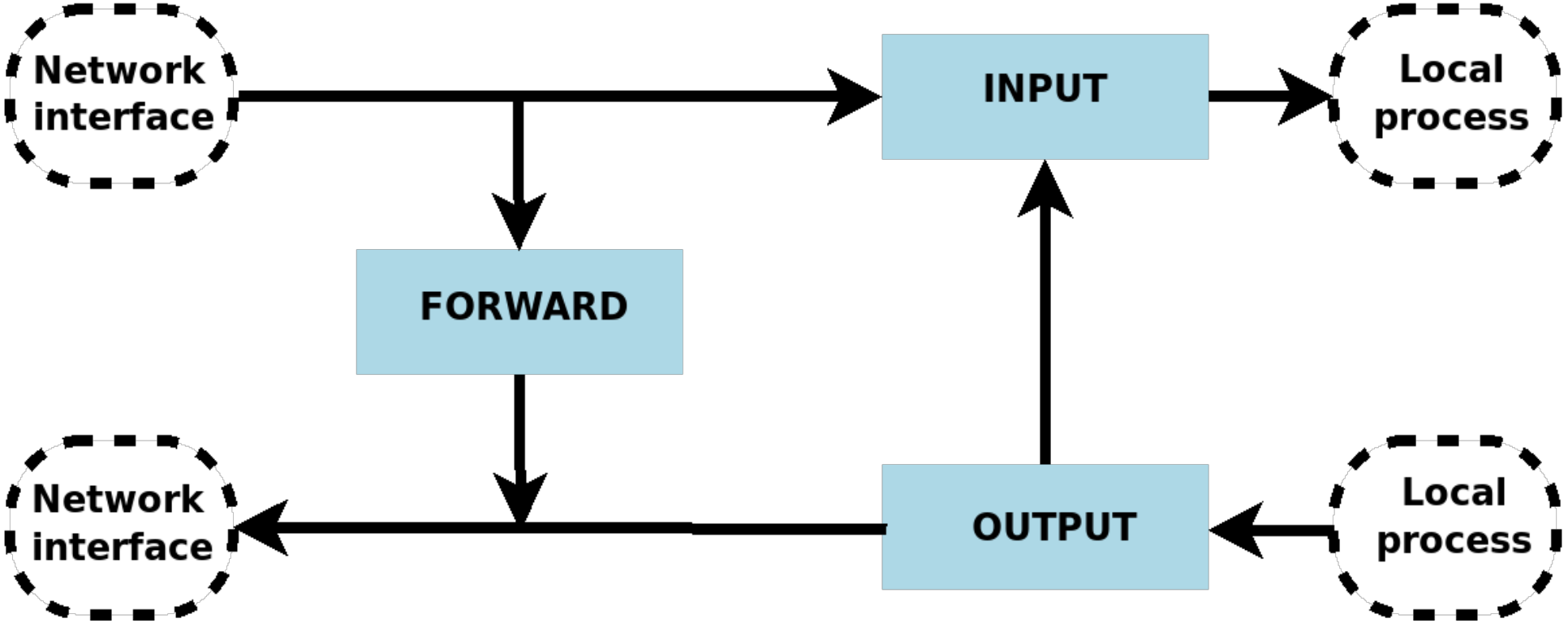
iptables defines five "hook points" in the kernel's packet processing pathways: ***PREROUTING, INPUT, FORWARD, POSTROUTING*** and ***OUTPUT***. Built-in chains are attached to these hook points; we can add a sequence of rules for each hook point. Each rule represents an opportunity to affect or monitor packet flow.

‘iptables’ -- tables and chains:

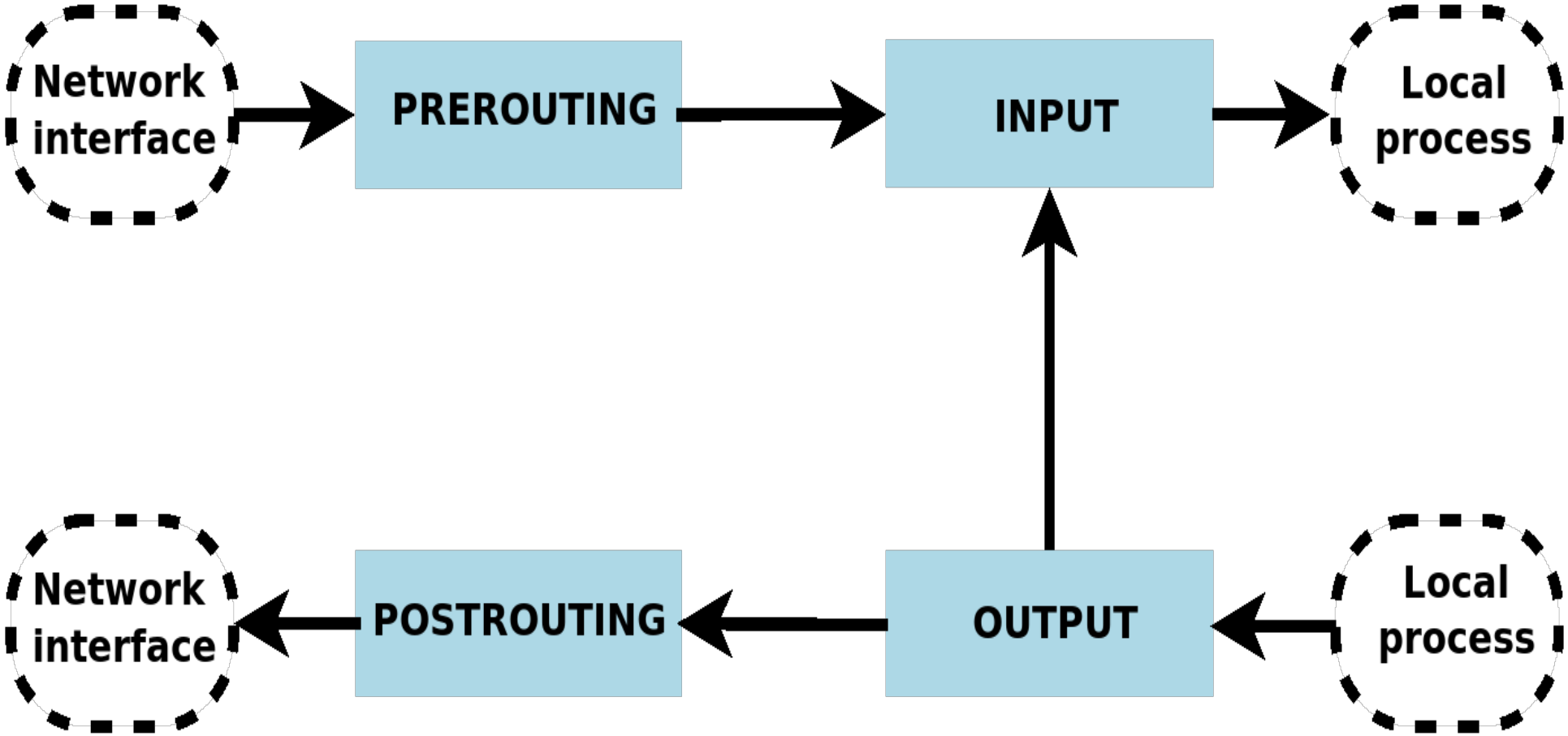
Tables	Description
mangle	Used for specialized packet alteration, such as stripping off IP options. Its built-in chains are: **PREROUTING, INPUT, FORWARD, OUTPUT and POSTROUTING**
nat	Used with connection tracking to redirect connections for network address translation; typically based on source or destination addresses. Its built-in chains are: **PREROUTING, INPUT, OUTPUT** and **POSTROUTING**
filter	Used to set policies for the type of traffic allowed into, through, and out of the computer. Its built-in chains are: **INPUT, FORWARD** and **OUTPUT**

Other tables: **raw** and **security.**

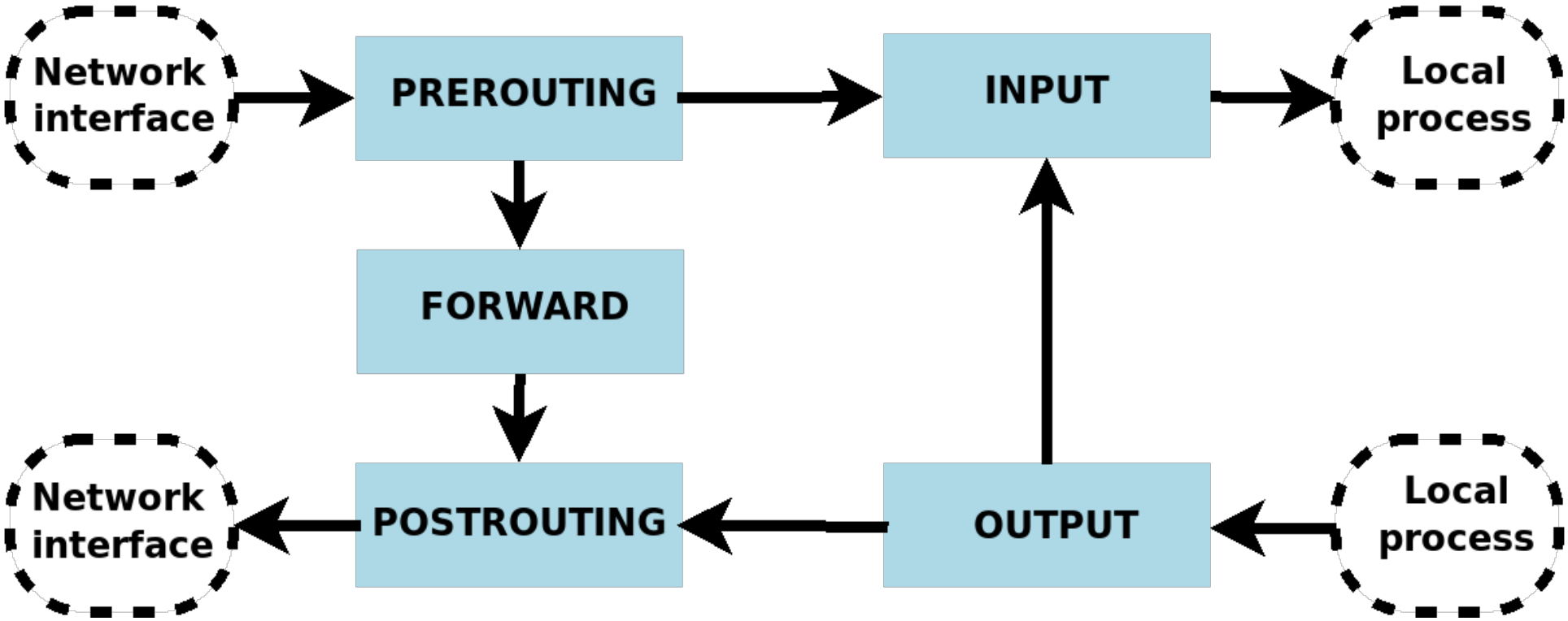
Network packet flow and hook points for filtering



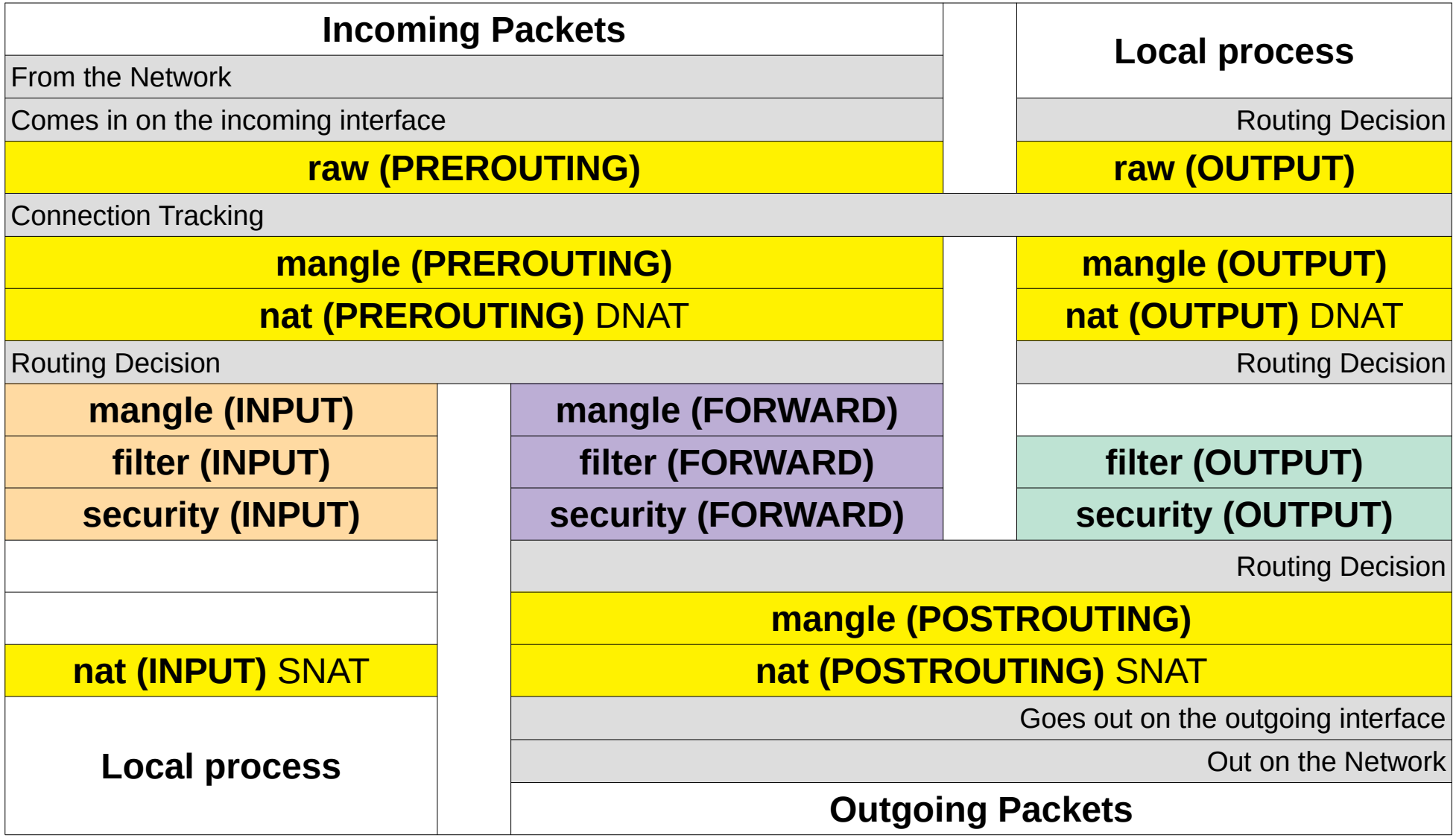
Network packet flow and hook points for NAT



Network packet flow and hook points for mangling



How packets traverse?:



Configuring ruleset using ‘iptables’:

‘**iptables**’ is a generic firewalling software that allows you to define rulesets. Each *rule* within **iptable** consists of a number of classifiers (i**ptables matches**) and one connected action (**iptables target**).

1. iptables command syntax:

iptables <commands> [<rule>]

where, *rule* = <*matches...*> *-j* <*target*>

2. iptables commands:

Commands	
-t, --table <table>	This option specifies the packet matching table which the command should operate on.
-A, --append <chain> <*rule*>	Append one or more rules to the end of the selected chain.
-D, --delete <chain> <*rule*> -D, --delete <chain> <rule-num>	Delete one or more rules from the selected chain.
-L, --list [<chain>]	List all rules in the selected chain. If no chain is selected, all chains in the table are listed.
-F, --flush [<chain>]	Flush the selected chain (all the chains in the table if none is given). This is equivalent to deleting all the rules one by one.
-Z, --zero [<chain> [<rule-num>]]	Zero the packet and byte counters in all chains, or only the given chain, or only the given rule in a chain.
-N, --new-chain <chain>	Create a new user-defined chain by the given name.
-X, --delete-chain [<chain>]	If there are, you must delete or replace the referring rules before the chain can be deleted. The chain must be empty, i.e. not contain any rules. If no argument is given, it will attempt to delete every non-builtin chain in the table.

2. iptables commands (cont.):

Commands	
-P, --policy <chain> <target>	Set the policy for the built-in (non-user-defined) chain to the given target. The policy target must be either **ACCEPT** or **DROP**.
-m, --match <match>	Specifies a match (for explicit matches) to use, that is, an extension module that tests for a specific property.
-j, --jump <target>	This specifies the target of the rule; i.e., what to do if the packet matches it. The target can be a user-defined chain (other than the one this rule is in).
-v, --verbose	Verbose output. -v may be specified multiple times to possibly emit more detailed debug statements.
-n, --numeric	Numeric output. IP addresses and port numbers will be printed in numeric format.
-x, --exact	Expand numbers. Display the exact value of the packet and byte counters
--line-numbers	When listing rules, add line numbers to the beginning of each rule, corresponding to that rule's position in the chain.

3. To view/list rules:

Command	To view all rules ...
iptables -n -v -x -L	all chains (INPUT, FORWARD, and OUTPUT) of filter table
iptables -nvxL	
iptables -nvxL --line-numbers	
iptables -t filter -nvxL	
iptables -nvx -L INPUT	INPUT chain of filter table.
iptables -t filter -nvxL INPUT	
iptables -t nat -nvxL	all chains of nat table.
iptables -t nat -nvxL POSTROUTING	POSTROUTING chain of nat table.
iptables -t mangle -nvxL	all chains of mangle table.
iptables -t mangle -nvxL INPUT	INPUT chain of mangle table.

4. Target (Terminating targets):

Target	
ACCEPT	This causes iptables to accept the packet
DROP	This causes iptables to drops the packet.
REJECT	iptables “rejects” the packet. It sends a “connection reset” packet in case of TCP, or a “destination host unreachable” packet in case of UDP or ICMP.

5. Matches (generic matches):

Match	
[!] -p, --protocol <protocol>	The protocol of the rule or of the packet to check. The specified protocol can be one of *tcp, udp, icmp, sctp* or it can be a numeric value, representing one of these protocols or a different one. A protocol name from /etc/protocols is also allowed. A "!" argument before the protocol inverts the test.
[!] -s, --src, --source <address[/mask][,...]>	Source specification. Address can be either a network name, a hostname, a network IP address (with /mask)
[!] -d, --dst, --destination <address[/mask][,...]>	Destination specification. Address can be either a network name, a hostname, a network IP address (with /mask)
[!] -i, --in-interface <ifname>	Name of an interface via which a packet was received (only for packets entering the INPUT, FORWARD and PREROUTING chains). If the interface name ends in a "+", then any interface which begins with this name will match.
[!] -o, --out-interface <ifname>	Name of an interface via which a packet is going to be sent (for packets entering the FORWARD, OUTPUT and POSTROUTING chains).

6. Matches (implicit matches):

ICMP Match	use the *-p icmp* or *--protocol icmp* option
[!] --icmp-type {<type[/code]>| <typename>}	This allows specification of the ICMP type, which can be a numeric ICMP type, type/code pair, or one of the ICMP type names shown by the command iptables -p icmp -h

TCP Match	use the *-p tcp* or *--protocol tcp* option
[!] --source-port,--sport <port[:port]>	Source port or port range specification. This can either be a service name or a port number. An inclusive range can also be specified, using the format first:last. If the first port is omitted, "0" is assumed; if the last is omitted, "65535" is assumed. The flag --sport is a convenient alias for this option.
[!] --destination-port,--dport <port[:port]>	Destination port or port range specification. The flag --dport is a convenient alias for this option.
[!] --tcp-flags <mask comp>	Match when the TCP flags are as specified. The first argument mask is the flags which we should examine, written as a comma-separated list, and the second argument comp is a comma-separated list of flags which must be set. Flags are: SYN ACK FIN RST URG PSH ALL NONE. Hence the command iptables -A FORWARD -p tcp --tcp-flags SYN,ACK,FIN,RST SYN will only match packets with the SYN flag set, and the ACK, FIN and RST flags unset.

UDP Match	use the *-p udp* or *--protocol udp* option
[!] --source-port,--sport <port[:port]>	Source port or port range specification. See the description of the --source-port option of the TCP extension for details.
[!] --destination-port,--dport <port[:port]>	Destination port or port range specification. See the description of the --destination-port option of the TCP extension for details.

7. Save and Restore Iptables Firewall Rules:

To save the chain rules that you have defined, use the iptables-save command as shown.

```
$ sudo iptables-save > ~/iptables.rules
```

```
admin@mypc:~
admin@mypc:~$ sudo iptables-save > ~/iptables.rules
```

If you wish to restore firewall rules from your previously created file, use the iptables-restore command as shown.

```
$ sudo iptables-restore < ~/iptables.rules
```

```
admin@mypc:~
admin@mypc:~$ sudo iptables-restore < ~/iptables.rules
```

Related Resources:

1. iptables Manpages $ man iptables
2. iptables-extensions Manpages $ man iptables-extensions