iptables

to install:

```
apt install iptables
```

```
┌──(root💀kali)-[~]
└─# apt install iptables
Reading package lists ... Done
Building dependency tree
Reading state information ... Done
iptables is already the newest version (1.8.6-1).
The following packages were automatically installed and are no longer required:
  libgdal27 libgeos-3.8.1 libllvm10 libmicrohttpd12 libpython3.8 libpython3.8-dev libwireshark13 libwiretap10 libwsutil11 libxcb-util0 linux-image-5.9.0-kali1-amd64
  python3-h2 python3-hpack python3-hyperframe python3.8-dev
Use 'apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 99 not upgraded.
```

to list all rules:

```
iptables -L
```

```
┌──(root💀kali)-[~]
└─#  iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

chain : a group of rules:

INPUT => is used for any packet coming into the system.
FORWARD => is for packets that are forwarded (routed) through the system.
OUTPUT=> is for any packet leaving the system.

to specify a chain:

'The default for the default rule is indeed to ACCEPT everything'
ACCEPT.
DROP.
REJECT.

Append to chain:

Iptables append firewall rules to the end of the selected chain.

```
iptables -A OUTPUT -p tcp --sport 2201 -j REJECT
iptables -A INPUT -p tcp --dport 2201 -j ACCEPT
```

```
┌──(root💀kali)-[~]
└─# iptables -A OUTPUT -p tcp --sport 22 -j REJECT
```

```
┌──(root💀kali)-[~]
└─# iptables -A INPUT -p tcp --dport 22 -j ACCEPT
```

We can see this now:

```
┌──(root💀kali)-[~]
└─# iptables -L --line-number
Chain INPUT (policy ACCEPT)
num  target     prot opt source               destination
1    ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:ssh

Chain FORWARD (policy ACCEPT)
num  target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
num  target     prot opt source               destination
1    REJECT     tcp  --  anywhere             anywhere             tcp spt:ssh reject-with icmp-port-unreachable
```

inserts:

insert will make the rule the first one or you can specify a number.

```
┌──(root💀kali)-[~]
└─# iptables -I INPUT 2 -p tcp --dport 443 -j ACCEPT
```

Let's review what we did:
-I INPUT 2 - Insert a rule to the "input" chain in the 2rd slot
-p tcp - Apply the rule to the tcp protocol
--dport 443 - Apply the rule to the port used by https (443)
-j ACCEPT - Set it to accept traffic to the input chain when using tcp on port 443

each rule has a number:

```
iptables -L --line-number
```

```
┌──(root💀kali)-[~]
└─# iptables -L --line-number
Chain INPUT (policy ACCEPT)
num  target     prot opt source               destination
1    ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:ssh

Chain FORWARD (policy ACCEPT)
num  target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
num  target     prot opt source               destination
1    REJECT     tcp  --  anywhere             anywhere             tcp spt:ssh reject-with icmp-port-unreachable
```

to show the range:

`iptables -L -n`

```
┌──(root💀kali)-[~]
└─# iptables -L --line-number -n
Chain INPUT (policy ACCEPT)
num  target     prot opt source               destination
1    ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0            tcp dpt:22

Chain FORWARD (policy ACCEPT)
num  target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
num  target     prot opt source               destination
1    REJECT     tcp  --  0.0.0.0/0            0.0.0.0/0            tcp spt:22 reject-with icmp-port-unreachable
```

Drop vs reject:

Drop : makes it holding. NO RESULT

Reject : reject it. THERE'S A RESULT.

internal LAN : Use reject.

external : Drop.

`iptables -A OUTPUT -p tcp --sport 22 -j REJECT`

```
┌──(root💀kali)-[~]
└─# iptables -A OUTPUT -p tcp --sport 22 -j REJECT
```

Remove rules:

Deleting Rules by Chain and Number:

`iptables -D INPUT 2`

```
┌──(root💀kali)-[~]
└─# iptables -L --line-number
Chain INPUT (policy ACCEPT)
num  target     prot opt source               destination
1    REJECT     tcp  --  anywhere             anywhere             tcp dpt:ssh reject-with icmp-port-unreachable

Chain FORWARD (policy ACCEPT)
num  target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
num  target     prot opt source               destination
1    REJECT     tcp  --  anywhere             anywhere             tcp spt:2201 reject-with icmp-port-unreachable
2    REJECT     tcp  --  anywhere             anywhere             tcp spt:ssh reject-with icmp-port-unreachable
3    REJECT     tcp  --  anywhere             anywhere             tcp spt:2201 reject-with icmp-port-unreachable
4    REJECT     tcp  --  anywhere             anywhere             tcp spt:ssh reject-with icmp-port-unreachable
5    REJECT     tcp  --  anywhere             anywhere             tcp spt:ssh reject-with icmp-port-unreachable
6    REJECT     tcp  --  anywhere             anywhere             tcp spt:ssh reject-with icmp-port-unreachable

┌──(root💀kali)-[~]
└─# iptables -D INPUT 1
```

We can see this now:

```
┌──(root💀kali)-[~]
└─# iptables -L --line-number
Chain INPUT (policy ACCEPT)
num  target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
num  target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
num  target     prot opt source               destination
1    REJECT     tcp  --  anywhere             anywhere             tcp spt:2201 reject-with icmp-port-unreachable
2    REJECT     tcp  --  anywhere             anywhere             tcp spt:ssh reject-with icmp-port-unreachable
3    REJECT     tcp  --  anywhere             anywhere             tcp spt:2201 reject-with icmp-port-unreachable
4    REJECT     tcp  --  anywhere             anywhere             tcp spt:ssh reject-with icmp-port-unreachable
5    REJECT     tcp  --  anywhere             anywhere             tcp spt:ssh reject-with icmp-port-unreachable
6    REJECT     tcp  --  anywhere             anywhere             tcp spt:ssh reject-with icmp-port-unreachable
```

Flush tables:

To flush a specific chain, which will delete all of the rules in the chain.
For example, to delete all of the rules in the INPUT chain
Now that you know how to delete individual firewall rules, let’s go over how you can flush chains of rules.

```
iptables -F INPUT
```

```
┌──(root💀kali)-[~]
└─# iptables -F INPUT
```

Flushing All Chains:
To flush all chains, which will delete all of the firewall rules.

```
┌──(root💀kali)-[~]
└─# iptables -F
```

Saving iptables firewall rules permanently on Linux:

we need to use the following commands to save iptables firewall rules
firstly install iptables-persistent.

```
apt install iptables-persistent
```

```
┌──(root💀kali)-[~]
└─# apt install iptables-persistent                                                    100 ✕
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libgdal27 libgeos-3.8.1 libllvm10 libmicrohttpd12 libpython3.8 libpython3.8-dev libwireshark13 libwiretap10 libwsutil11 libxcb-util0 linux-image-5.9.0-kali1-amd64
  python3-h2 python3-hpack python3-hyperframe python3.8-dev
Use 'apt autoremove' to remove them.
The following additional packages will be installed:
  netfilter-persistent
The following NEW packages will be installed:
  iptables-persistent netfilter-persistent
0 upgraded, 2 newly installed, 0 to remove and 99 not upgraded.
Need to get 22.9 kB of archives.
After this operation, 87.0 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://kali.download/kali kali-rolling/main amd64 netfilter-persistent all 1.0.14 [10.6 kB]
Get:2 http://kali.download/kali kali-rolling/main amd64 iptables-persistent all 1.0.14 [12.3 kB]
Fetched 22.9 kB in 5s (4,628 B/s)
Preconfiguring packages ...
Selecting previously unselected package netfilter-persistent.
(Reading database ... 272103 files and directories currently installed.)
Preparing to unpack .../netfilter-persistent_1.0.14_all.deb ...
Unpacking netfilter-persistent (1.0.14) ...
Selecting previously unselected package iptables-persistent.
Preparing to unpack .../iptables-persistent_1.0.14_all.deb ...
Unpacking iptables-persistent (1.0.14) ...
Setting up netfilter-persistent (1.0.14) ...
update-rc.d: We have no instructions for the netfilter-persistent init script.
update-rc.d: It looks like a non-network service, we enable it.
netfilter-persistent.service is a disabled or a static unit, not starting it.
Setting up iptables-persistent (1.0.14) ...
update-alternatives: using /lib/systemd/system/netfilter-persistent.service to provide /lib/systemd/system/iptables.service (iptables.service) in auto mode
Processing triggers for man-db (2.9.3-2) ...
Processing triggers for kali-menu (2021.1.1) ...
```

Choose Yes

```
┤ Configuring iptables-persistent ├

Current iptables rules can be saved to the configuration file /etc/iptables/rules.v4. These rules will then be loaded automatically during system startup.

Rules are only saved automatically during package installation. See the manual page of iptables-save(8) for instructions on keeping the rules file up-to-date.

Save current IPv4 rules?

                    <Yes>                                        <No>
```

to include new rules into your system. To make changes permanent after reboot use command:

```
iptables-save > /etc/iptables/rules.v4
```

```
┌──(root💀kali)-[~]
└─# iptables-save > /etc/iptables/rules.v4
```

To read the rules:

```
cat /etc/iptables/rules.v4
```

```
┌──(root💀kali)-[~]
└─# cat /etc/iptables/rules.v4
# Generated by iptables-save v1.8.6 on Tue May 11 22:17:31 2021
*filter
:INPUT ACCEPT [98:35400]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [65:5112]
-A OUTPUT -p tcp -m tcp --sport 2201 -j REJECT --reject-with icmp-port-unreachable
-A OUTPUT -p tcp -m tcp --sport 22 -j REJECT --reject-with icmp-port-unreachable
-A OUTPUT -p tcp -m tcp --sport 2201 -j REJECT --reject-with icmp-port-unreachable
-A OUTPUT -p tcp -m tcp --sport 22 -j REJECT --reject-with icmp-port-unreachable
-A OUTPUT -p tcp -m tcp --sport 22 -j REJECT --reject-with icmp-port-unreachable
-A OUTPUT -p tcp -m tcp --sport 22 -j REJECT --reject-with icmp-port-unreachable
COMMIT
# Completed on Tue May 11 22:17:31 2021
```

Asalah Mohammed.
All The best!