TARGETS

A TARGET is the action that is triggered when a packet meets the matching criteria of a rule.

A TARGET can be:

- Terminating (Example: ACCEPT, DROP)
- Non-terminating (Example: LOG, TEE)

The table and the chain will dictate the TARGET availability.

ACCEPT

- ACCEPT is a terminating target.

Example: iptables -A INPUT -p tcp --dport 25 -j ACCEPT

DROP

- DROP is a terminating target
- DROP denies the packet and doesn’t send any packet back to the source

Example: iptables -I OUTPUT -p udp --dport 53 ! -d 8.8.8.8 -j DROP

REJECT

- REJECT is a terminating target
- Like DROP it denies the packet but also sends back a reply packet to the source
- By default it sends back an ICMP port unreachable packet
- It is possible to modify the response packet using --reject-with option
- Sometimes it’s more efficient to REJECT than DROPPING the packet

Example: iptables -I FORWARD -p udp --dport 69 \
-j REJECT --reject-with icmp-port-unreachable

TCP/UDP Ports

Port types:

1. **OPEN**
   - There is an application that listens on an OPEN port. We can communicate with that application
   - An OPEN port responds
   - We can list open ports using netstat command
2. **CLOSED**
   - There is no application that listens on a CLOSED port
   - A CLOSED port responds too, with tcp reset for tcp traffic and with icmp for udp traffic
3. **FILTERED/STEALTH**
   - A firewall drops the packet. The port can be OPEN or CLOSED on host, but we can’t communicate with it
   - A FILTERED port doesn’t respond

LOG

- LOG is a non-terminating target
- It logs detailed information about packets headers
- Logs can be read with dmesg or from syslogd daemon
- LOG is used instead of DROP in the debugging phase
- ULOG has MySql support (extensive logging)

Example: iptables -A INPUT -p tcp --dport 22 --syn -j LOG --log-prefix="incoming ssh:" --log-level info

TEE

The TEE target will clone a packet and redirect this clone to another machine on the local subnet.

It is used for traffic mirroring.

Example:
iptables -A FORWARD -i eth0 -o eth1 -p tcp -d 80.0.0.1 -j TEE --gateway 10.0.0.10

Raw table & NOTRACK Target

- The raw table has 2 built-in CHAINS: **PREROUTING** & **OUTPUT**
- Every packet is tracked by the netfilter connection tracking system
- The NOTRACK target is used for packets that must be **skipped** by the connection tracking system

REDIRECT

- Used to redirect packets from one port to another on the same machine.
- The **REDIRECT** target is extremely good to use for transparent proxying, where the *LAN* hosts do not know about the proxy at all.
- REDIRECT target is only valid within the **PREROUTING** and **OUTPUT** chains of the **nat table**.

Example:

iptables -t nat -A PREROUTING -p tcp --dport 80 -j REDIRECT --to-ports 8080