

Using iptables as a Firewall

LPIC-2: Linux Engineer (202-450)

Objectives:

At the end of this episode, I will be able to:

1. List and define the chains and actions used in *iptables*
2. Configure firewall rules in *iptables*

Additional resources used during the episode can be obtained using the download link on the overview episode.

-
- Using iptables as a Firewall
 - Chains and Actions
 - Creating firewall rules
 - Managing bidirectional communications
 - Three chains
 - INPUT
 - Incoming traffic
 - Where most of our filtering is performed
 - FORWARD
 - Routed traffic
 - Not used on most machines
 - Typically used for routers/firewalls
 - OUTPUT
 - Outbound traffic
 - Not typically filtered for normal hosts
 - Firewall actions
 - ALLOW - Permits the connection
 - DROP - Discards any connection traffic without notifying the sender
 - REJECT - Discards any connection traffic and notifies the sender
 - LOG - Creates a record of the traffic
 - Using the iptables command
 - Block traffic to SSH
 - `iptables -A INPUT -p tcp --dport ssh -j DROP`
 - Allow a single host access to SSH
 - `iptables -A INPUT -p tcp --dport ssh -s 10.10.10.10 -m state --state NEW,ESTABLISHED -j ACCEPT`
 - Save configuration
 - `sudo iptables-save | sudo tee /etc/iptables/rules.v4`
 - Modifying firewall rules
 - Rules are applied in order.
 - Sometimes easier to directly edit the config
 - Modifying the iptables configuration file
 - `/etc/iptables/rules.v4`
 - `systemctl restart iptables`
 - Bidirectional Control Example
 - `iptables -A OUTPUT -p tcp -d 172.16.0.1 --dport 3306 -m state --state NEW,ESTABLISHED`
 - `iptables -A INPUT -p tcp --sport 3306 -m state --state ESTABLISHED -j ACCEPT`

- Testing *iptables*
 - You can test connections manually
 - Tools like *nmap* can help as well
 - Log option
 - You can create a duplicate rule with the *log* action
 - `iptables -A INPUT -p tcp --dport ssh -j DROP`
 - `iptables -A INPUT -p tcp --dport ssh -j LOG`
 - Statistics can be monitored with the *watch* command
 - `iptables -vnL --line`
 - `watch -n 0.5 iptables -vnL`