Workshop 10 – Firewall

Part1: Service status checking

Step 1: Checking service status:
 # service iptables status

Step 2: Run if it is not running
 # service iptables start

Step 3: Permanent the servcie
 # chkconfig iptables on

Part2: Delete Existing Rules

Before you start building new set of rules, you might want to clean-up all the default rules, and existing rules.

 # iptables -F
(or)
 # iptables –flush

Part 3: Set Default Chain Policies

The default chain policy is ACCEPT. Change this to DROP for all INPUT, FORWARD, and OUTPUT chains as shown below.

 # iptables -P INPUT DROP
 # iptables -P FORWARD DROP
 # iptables -P OUTPUT DROP

Note: When you make both INPUT, and OUTPUT chain's default policy as DROP, for every firewall rule requirement you have, you should define two rules. i.e one for incoming and one for outgoing.

Part 4: Allow ALL Incoming SSH

The following rules allow ALL incoming ssh connections on eth0 interface.

 # iptables -A INPUT -i eth0 -p tcp --dport 22 -m state --state NEW,ESTABLISHED -j ACCEPT
 # iptables -A OUTPUT -o eth0 -p tcp --sport 22 -m state --state ESTABLISHED -j ACCEPT

Part 5: Allow Incoming SSH only from a Sepcific Network

The following rules allow incoming ssh connections only from 192.168.100.X network.
 # iptables -A INPUT -i eth0 -p tcp -s 192.168.100.0/24 --dport 22 -m state --state NEW,ESTABLISHED -j ACCEPT
 # iptables -A OUTPUT -o eth0 -p tcp --sport 22 -m state --state ESTABLISHED -j ACCEPT

Note: In the above example, instead of /24, you can also use the full subnet mask. i.e "192.168.100.0/255.255.255.0".

Part 6: Allow Incoming HTTP and HTTPS

Step1: The following rules allow all incoming web traffic. i.e HTTP traffic to port 80.

 # iptables -A INPUT -i eth0 -p tcp --dport 80 -m state --state NEW,ESTABLISHED -j ACCEPT
 # iptables -A OUTPUT -o eth0 -p tcp --sport 80 -m state --state ESTABLISHED -j ACCEPT

Step2: The following rules allow all incoming secure web traffic. i.e HTTPS traffic to port 443.

 # iptables -A INPUT -i eth0 -p tcp --dport 443 -m state --state NEW,ESTABLISHED -j ACCEPT
 # iptables -A OUTPUT -o eth0 -p tcp --sport 443 -m state --state ESTABLISHED -j ACCEPT

Part 7: Combine Multiple Rules Together using MultiPorts

The following example allows all incoming SSH, HTTP and HTTPS traffic.

 # iptables -A INPUT -i eth0 -p tcp -m multiport --dports 22,80,443 -m state --state NEW,ESTABLISHED -j ACCEPT

 # iptables -A OUTPUT -o eth0 -p tcp -m multiport --sports 22,80,443 -m state --state ESTABLISHED -j ACCEPT

Part 8: Allow Outgoing SSH only to a Specific Network

 # iptables -A OUTPUT -o eth0 -p tcp -d 192.168.100.0/24 --dport 22 -m state --state NEW,ESTABLISHED -j ACCEPT
 # iptables -A INPUT -i eth0 -p tcp --sport 22 -m state --state ESTABLISHED -j ACCEPT

Part 9: Allow Ping from Outside to Inside
The following rules allow outside users to be able to ping your servers.

 # iptables -A INPUT -p icmp --icmp-type echo-request -j ACCEPT
 # iptables -A OUTPUT -p icmp --icmp-type echo-reply -j ACCEPT

Part 10: Allow Ping from Inside to Outside
The following rules allow you to ping from inside to any of the outside servers.

 # iptables -A OUTPUT -p icmp --icmp-type echo-request -j ACCEPT
 # iptables -A INPUT -p icmp --icmp-type echo-reply -j ACCEPT

Part 11: Enabling NAT Features

 # iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE

Part 12: New Chain, log and limit example:

First flush all rules and policies

```
# iptables -N SSH
# iptables -A INPUT -p tcp --dport 22 ! --syn  -j ACCEPT
# iptables -A INPUT -p tcp --dport 22 --syn  -j SSH
# iptables -A SSH -s 192.168.0.0/27 -j ACCEPT
# iptables -A SSH -m limit --limit 5/s -j LOG
# iptables -A SSH -j DROP
```