FSI LEI 2025/2026

Practical Exercises #2 Resolution examples

Goals

Configure a system firewall using IPTables in Linux

Use IPTables to configure a Linux system firewall

1. **Clear** all the rules on the system's firewall configuration

```
# The following command flushes (clears) all firewall rules
# Please note that this does not change the policies
iptables -F
```

2. Create a firewall rule to ignore incoming **ping** requests from another host (can be tested using the localhost address – 127.0.0.1), while authorizing all the remaining IP packets. Note: ping uses ICMP packets of types 8 (**echo request**) and 0 (**echo reply**)

```
# The following rule drops all ping requests entering the system
iptables -A INPUT -s 127.0.0.1 -p icmp --icmp-type echo-request -j DROP
# Alternatively, we can use the input interface in the rule (loopback
# interface or "lo")
iptables -A INPUT -i lo -p icmp --icmp-type echo-request -j DROP
```

3. Create firewall rules to authorize the following incoming TCP connections (filter table, INPUT chain), while rejecting (only) other TCP communications:
    a. **SSH** connections originated at the server student.dei.uc.pt
    b. **POP3** and **IMAP4** connections originated at any other hosts.

```
iptables -A INPUT -s student.dei.uc.pt -p tcp --dport ssh -j ACCEPT
iptables -A INPUT -p tcp --dport pop3 -j ACCEPT
iptables -A INPUT -p tcp --dport imap -j ACCEPT
# Now we drop any other request for new TCP connections to our server
# while without dropping packets that belong to already established
# communications
iptables -A INPUT -p tcp --syn -j DROP
```

4. Add to the previous configuration firewall rules to authorize the following outgoing TCP connections (filter table, OUTPUT chain), while rejecting (only) other TCP communications:
    a. **HTTP** and **HTTPS** connections destined to the server student.dei.uc.pt
    b. **SSH** connections destined to any other hosts.

```
iptables -A OUTPUT -d student.dei.uc.pt -p tcp --dport http -j ACCEPT
iptables -A OUTPUT -d student.dei.uc.pt -p tcp --dport https -j ACCEPT
iptables -A OUTPUT -p tcp --dport ssh -j ACCEPT
# Now we deny any other request for the establishment of new outgoing
# TCP connections, while without dropping packets of already
# connections
iptables -A OUTPUT -p tcp --syn -j DROP
```

5. **Clear** all the firewall rules defined in the previous exercises

```
# The following command flushes (clears) all rules
# Please note that it does not change chain's policies
iptables -F
```

6. Use IPTables to authorize the following communications, while denying

Materials

- Segurança Prática em Sistemas e Redes com Linux, Jorge Granjal, FCA 2017, "Capítulo 17. Proteção de Servidores"
- Red Hat Enterprise Linux Security Guide: 2.8 Firewalls
- The netfilter.org Project
- Linux 2.4 Packet Filtering HOWTO

the remaining IP traffic (policy DROP on both the INPUT and OUTPUT chains):

a. Incoming **SSH** and **HTTP** connections
b. Outgoing **SSH**, **HTTP** and **HTTPS** connections
c. **DNS** queries sent to the server dns.dei.uc.pt and dns2.dei.uc.pt
d. Incoming **ping** requests from the server student.dei.uc.pt
e. All IP communications to or from the **localhost** (127.0.0.1, or interface **lo**)

```
# In this configuration the goal is to apply the DROP policy to both
# the INPUT and OUTPUT chains
# In this context, we need to authorize all applications in the two chains

# Incoming SSH and HTTP connections
iptables -A INPUT -p tcp --dport ssh -j ACCEPT
iptables -A INPUT -p tcp --dport http -j ACCEPT
iptables -A OUTPUT -p tcp --sport ssh -j ACCEPT
iptables -A OUTPUT -p tcp --sport http -j ACCEPT

# Outgoing SSH, HTTP and HTTPS connections
iptables -A OUTPUT -p tcp --dport ssh -j ACCEPT
iptables -A OUTPUT -p tcp --dport http -j ACCEPT
iptables -A OUTPUT -p tcp --dport https -j ACCEPT
iptables -A INPUT -p tcp --sport ssh -j ACCEPT
iptables -A INPUT -p tcp --sport http -j ACCEPT
iptables -A INPUT -p tcp --sport https -j ACCEPT

# DNS queries sent to the server dns.dei.uc.pt and dns2.dei.uc.pt
iptables -A OUTPUT -p udp --dport domain -d dns.dei.uc.pt -j ACCEPT
iptables -A OUTPUT -p udp --dport domain -d dns2.dei.uc.pt -j ACCEPT
iptables -A INPUT -p udp --sport domain -s dns.dei.uc.pt -j ACCEPT
iptables -A INPUT -p udp --sport domain -s dns2.dei.uc.pt -j ACCEPT

# Incoming ping requests received from the server student.dei.uc.pt
iptables -A INPUT -s student.dei.uc.pt -p icmp --icmp-type echo-request -j
ACCEPT
iptables -A OUTPUT -d student.dei.uc.pt -p icmp --icmp-type echo-reply -j
ACCEPT

# All IP communications to or from the localhost (127.0.0.1, or interface
"lo")
iptables -A INPUT -i lo -s 127.0.0.1 -j ACCEPT
iptables -A OUTPUT -o lo -d 127.0.0.1 -j ACCEPT

# DROP all remaining IP communications
iptables -P INPUT DROP
iptables -P OUTPUT DROP
```

7. Activate the previous firewall configuration **permanently** on the system

```
# The iptables-save copies the (runtime) firewall configuration to a file
iptables-save > my_iptables.backup

# The configuration save in a file can restored (enabled in runtime)
iptables-restore < my_iptables.backup
```