IN3210/4210 – solution proposal: iptables firewall configuration

This exercise should be performed using iptables (e.g., in SEED Ubuntu). Refer *man iptables* and https://help.ubuntu.com/community/IptablesHowTo for information.

1. Configure the INPUT chain to only accept incoming traffic for already established connections, dropping all other traffic.

   Solution proposal:

   Add a rule to the INPUT chain to accept established connections:

   - *sudo iptables -A INPUT –m conntrack –-ctstate ESTABLISHED -j ACCEPT*

   You may also include *RELATED,* in addition to *ESTABLISHED* (i.e., *ESTABLISHED,RELATED*), to allow new connections that are related to an existing connection (e.g., in FTP).

   Add rule to the end of the INPUT chain to drop all traffic:

   - *sudo iptables -A INPUT -j DROP*

   You may want to specify the interface(s) this applies to (by including e.g., *-i eth0,eth1* above) or to add a separate rule to allow the loopback interface at the beginning of the chain, e.g.,: *sudo iptables -I INPUT 1 -i lo -j ACCEPT*)

   You may show the resulting iptables configuration using:

   - *sudo iptables -L* -v

```
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
   84  8990 ACCEPT     all  --  lo     any     anywhere             anywhere
  288  223K ACCEPT     all  --  any    any     anywhere             anywhere             ctstate ESTABLISHED
    0     0 DROP       all  --  any    any     anywhere             anywhere

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination

Chain OUTPUT (policy ACCEPT 362 packets, 43974 bytes)
 pkts bytes target     prot opt in     out     source               destination
```

2. How will you configure the OUTPUT and FORWARD chains? How would this differ for different scenarios/deployments (e.g., home PC, mail server,…)?

   Solution proposal:
   The above configuration could provide a "minimal" configuration for e.g., a home network that would require little configuration maintenance (possibly including the RELATED state). Alternatively the OUTPUT chain could be restricted to only allow selected protocols (i.e., destination port numbers), e.g., HTTP and HTTPS:
   - *sudo iptables -I OUTPUT -p tcp --dport https -j ACCEPT*
   - *sudo iptables -I OUTPUT -p tcp --dport http -j ACCEPT*
   - *sudo iptables -I OUTPUT -j DROP*

   Using the above configuration we would at least also need to add DNS (port 53), which could then also be restricted to the IP-addresses(s) of the DNS server(s) to be used. It may be

argued that we would want a more restrictive configuration for outgoing traffic from a server, which protocol usage is more predictable/static, than for a home network in general.