Firewall

17.1. What is `netfilter` and what are its benefits?

17.2. What are the five `netfilter` hooks for IPv4? What are their purposes?

17.3. Why do we need to build a kernel module in order to use the `netfilter` hooks?

17.4. The following code tries to block the computer from accessing the web server (HTTP) running on host `10.0.2.5`. Please complete the code by replacing `@@@@@@@` with actual code.

```
static struct nf_hook_ops filterHook;
int setUpFilter(void){
  filterHook.hook = @@@@@@;                    ①
  filterHook.hooknum = NF_INET_POST_ROUTING;
  filterHook.pf = PF_INET;
  filterHook.priority = NF_IP_PRI_FIRST;
  nf_register_hook(&@@@@@@@);                  ②
  return 0;
}
void removeFilter(void){
  nf_unregister_hook(&@@@@@@@);                ③
}
module_init(@@@@@@@);                          ④
module_exit(@@@@@@@);                          ⑤


unsigned int block(void *priv, struct sk_buff *skb,
                 const struct nf_hook_state *state)
{
  if(!skb){
    printk(KERN_INFO,"packet receive not correct\n");
    return NF_DROP;
  }

  struct iphdr *iph;
  struct tcphdr *tcph;
  iph = ip_hdr(@@@@@@@);                       ⑥
  tcph = (void *)iph+iph->ihl*4;

  __u32 sou_ip = iph->saddr;
  __u32 des_ip = iph->daddr;
  __u16 sou_port = tcph->source;
  __u16 des_port = tcph->dest;

  if(des_ip==in_aton("@@@@@@") &&              ⑦
     ntohs(des_port)== @@@@@@) {               ⑧
    return @@@@@@;                             ⑨
```

```
  }
  return NF_ACCEPT;
}
```

17.5. Based on the `netfilter` diagram (can be found in the book), please describe which filter is best for enforcing the following rules:

- Restricting what comes into a computer
- Restricting what goes out of a computer

17.6. Other than being used to implement firewalls to block packets, can `netfilter` be used to modify packets? What are the other applications of `netfilter`?

17.7. The `SYNPROXY` is a firewall to filter out SYN flooding attack packets. Please find articles from the Internet about `SYNPROXY`, and explain at high-level how it works.

17.8. What are the benefits of stateful firewalls that support connection-based firewall rules? Please use examples to illustrate the benefit.

17.9. The UDP and ICMP protocols are not connection-based protocols, how do firewalls know whether a UDP or ICMP packet is part of an existing "connection"?

17.10. In Ubuntu, a program is called `ufw`, which stands for Uncomplicated Firewall. Is this a real firewall?

17.11. Add a rule in iptables to accept packets from a trusted network `192.168.10.0/24`

17.12. A machine has an IP address `10.0.20.5`. On this machine, you need to block incoming connections to its ports 22, 23, 80, and 443. What will you do?

17.13. A TCP server is running on a remote machine called `sirius` using "`nc -lv 9090`". This machine is on a planet outside the Solar system. An alien named Alice living on the Earth wants to communicate with the TCP server on `sirius`, but unfortunately, the Earth has a firewall that prevents all computers on the Earth from accessing any machine outside the Solar system. Alice does have a computer on Mars, which does not have such a restrict firewall rule. Alice's computer on Mars is called `mars`, and her account name is called `alien`. (1) Please describe how Alice can use an SSH tunnel to bypass Earth's firewall, so she can talk to `sirius`. (2) Without the firewall, if Alice wants to communicate with the TCP server on `sirius`, she can use the "`nc sirius 9090`" command. Now, with the SSH tunnel and the firewall, what command should Alice run to access the server?

17.14. This problem is based on Problem 17.13. After Alice has established an SSH tunnel between her local computer `earth` and `mars`, she can use the `nc` command to communicate with the `netcat` server on `sirius`. Please describe how the TCP packets flow, from the `netcat` client program to the destination `netcat` server.

17.15. This problem is based on Problem 17.13. The alien Alice has many friends outside the Solar system, and she wants to connect with them by visiting various social network sites hosted on the planets where her friends live. Establishing one SSH tunnel for each social network site is tedious. Can you help Alice set up one single SSH tunnel, which she can use to stay connected with her friends?

17.16. This problem is based on Problem 17.13. Alice also runs a web server on her machine `earth` on the Earth, and she would like her friends from her home planet to visit this web server. Unfortunately, the Earth has a firewall the prevents computers outside the Solar system from accessing any computer on the Earth. Alice would like to use her computer on Mars to set up a port forwarding using SSH, so instead of visiting Alice's machine on Earth, her friends can point their browsers to `mars`, which automatically forward the traffic to Alice's machine on the Earth. This kind of port forwarding is called *remote port forwarding*, which is not covered in the book. Please read about this kind of port forwarding from the Internet, and then describe how to use it to help Alice.