LAB 6

57118116 陈煜

Task 1.A

将 kernel_module 复制到桌面，并编译：

```
[07/26/21]seed@VM:~/.../kernel_module$ make
make -C /lib/modules/5.4.0-54-generic/build M=/home/seed/Desktop/kernel_module m
odules
make[1]: Entering directory '/usr/src/linux-headers-5.4.0-54-generic'
  CC [M]  /home/seed/Desktop/kernel_module/hello.o
  Building modules, stage 2.
  MODPOST 1 modules
WARNING: modpost: missing MODULE_LICENSE() in /home/seed/Desktop/kernel_module/h
ello.o
see include/linux/module.h for more information
  CC [M]  /home/seed/Desktop/kernel_module/hello.mod.o
  LD [M]  /home/seed/Desktop/kernel_module/hello.ko
make[1]: Leaving directory '/usr/src/linux-headers-5.4.0-54-generic'
```

测试如下命令：

```
[07/26/21]seed@VM:~/.../kernel_module$  sudo insmod hello.ko
[07/26/21]seed@VM:~/.../kernel_module$  lsmod | grep hello
hello                  16384  0
[07/26/21]seed@VM:~/.../kernel_module$  sudo rmmod hello

[07/26/21]seed@VM:~/.../kernel_module$ dmesg | grep World
[67291.136456] Hello World!
[67369.790729] Bye-bye World!.
```

出现了 Hello World!和 Bye-bye World!，与预期结果相同。

Task 1.B

1. 使用如下命令：

```
[07/26/21]seed@VM:~/.../kernel_module$ dig @8.8.8.8 www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> @8.8.8.8 www.example.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 28364
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;www.example.com.               IN      A

;; ANSWER SECTION:
www.example.com.        21111   IN      A       93.184.216.34

;; Query time: 271 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Mon Jul 26 03:51:55 EDT 2021
;; MSG SIZE  rcvd: 60
```

可以得到响应。

将 packet_filter 拷贝到桌面，编译，加载内核：

```
[07/26/21]seed@VM:~/.../packet_filter$ make
make -C /lib/modules/5.4.0-54-generic/build M=/home/seed/Desktop/packet_filter m
odules
make[1]: Entering directory '/usr/src/linux-headers-5.4.0-54-generic'
  CC [M]  /home/seed/Desktop/packet_filter/seedFilter.o
  Building modules, stage 2.
  MODPOST 1 modules
  CC [M]  /home/seed/Desktop/packet_filter/seedFilter.mod.o
  LD [M]  /home/seed/Desktop/packet_filter/seedFilter.ko
make[1]: Leaving directory '/usr/src/linux-headers-5.4.0-54-generic'
[07/26/21]seed@VM:~/.../packet_filter$ sudo insmod seedFilter.ko
[07/26/21]seed@VM:~/.../packet_filter$ lsmod | grep seedFilter
seedFilter             16384  0
```

再次运行上述命令：

```
[07/26/21]seed@VM:~/.../packet_filter$ dig @8.8.8.8 www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> @8.8.8.8 www.example.com
; (1 server found)
;; global options: +cmd
;; connection timed out; no servers could be reached
```

连接不成功，防火墙生效。

最后移除模块。

```
[07/26/21]seed@VM:~/.../packet_filter$ sudo rmmod seedFilter
[07/26/21]seed@VM:~/.../packet_filter$ lsmod | grep seedFilter
```

2. 在 seedFilter.c 中修改代码如下，增加 hook：

```
12 static struct nf_hook_ops hook1, hook2, hook3, hook4, hook5, hook6;
```

```
74 int registerFilter(void) {
75    printk(KERN_INFO "Registering filters.\n");
76
77    hook1.hook = printInfo;
78    hook1.hooknum = NF_INET_LOCAL_OUT;
79    hook1.pf = PF_INET;
80    hook1.priority = NF_IP_PRI_FIRST;
81    nf_register_net_hook(&init_net, &hook1);
82
83    hook2.hook = blockUDP;
84    hook2.hooknum = NF_INET_POST_ROUTING;
85    hook2.pf = PF_INET;
86    hook2.priority = NF_IP_PRI_FIRST;
87    nf_register_net_hook(&init_net, &hook2);
88
89    hook3.hook = printInfo;
90    hook3.hooknum = NF_INET_LOCAL_IN;
91    hook3.pf = PF_INET;
92    hook3.priority = NF_IP_PRI_FIRST;
93    nf_register_net_hook(&init_net, &hook3);
94
95    hook4.hook = printInfo;
96    hook4.hooknum = NF_INET_FORWARD;
97    hook4.pf = PF_INET;
98    hook4.priority = NF_IP_PRI_FIRST;
99    nf_register_net_hook(&init_net, &hook4);
```

```
101    hook5.hook = printInfo;
102    hook5.hooknum = NF_INET_PRE_FIRST;
103    hook5.pf = PF_INET;
104    hook5.priority = NF_IP_PRI_FIRST;
105    nf_register_net_hook(&init_net, &hook5);
106
107    hook6.hook = printInfo;
108    hook6.hooknum = NF_INET_POST_ROUTING;
109    hook6.pf = PF_INET;
110    hook6.priority = NF_IP_PRI_FIRST;
111    nf_register_net_hook(&init_net, &hook6);
112
113    return 0;
114 }
115
116 void removeFilter(void) {
117    printk(KERN_INFO "The filters are being removed.\n");
118    nf_unregister_net_hook(&init_net, &hook1);
119    nf_unregister_net_hook(&init_net, &hook2);
120    nf_unregister_net_hook(&init_net, &hook3);
121    nf_unregister_net_hook(&init_net, &hook4);
122    nf_unregister_net_hook(&init_net, &hook5);
123    nf_unregister_net_hook(&init_net, &hook6);
124 }
125
```

重新编译并加载：

```
[07/26/21]seed@VM:~/.../packet_filter$ make
make -C /lib/modules/5.4.0-54-generic/build M=/home/seed/Desktop/packet_filter m
odules
make[1]: Entering directory '/usr/src/linux-headers-5.4.0-54-generic'
  CC [M]  /home/seed/Desktop/packet_filter/seedFilter.o
  Building modules, stage 2.
  MODPOST 1 modules
  CC [M]  /home/seed/Desktop/packet_filter/seedFilter.mod.o
  LD [M]  /home/seed/Desktop/packet_filter/seedFilter.ko
make[1]: Leaving directory '/usr/src/linux-headers-5.4.0-54-generic'
[07/26/21]seed@VM:~/.../packet_filter$ sudo insmod seedFilter.ko
[07/26/21]seed@VM:~/.../packet_filter$ lsmod | grep seedFilter
seedFilter             16384  0
```

进入 docker。

```
[07/26/21]seed@VM:~/.../Labsetup$ dockps
901343cf23a1  host3-192.168.60.7
5bc871c247c0  host1-192.168.60.5
e10d24587811  hostA-10.9.0.5
ea55371f5c62  host2-192.168.60.6
6bba5316370b  seed-router
```

在 10.9.0.5 中 ping 10.9.0.1：

```
[07/26/21]seed@VM:~/.../Labsetup$ docksh e1
root@e10d24587811:/# ping 10.9.0.1
PING 10.9.0.1 (10.9.0.1) 56(84) bytes of data.
64 bytes from 10.9.0.1: icmp_seq=1 ttl=64 time=0.093 ms
64 bytes from 10.9.0.1: icmp_seq=2 ttl=64 time=0.177 ms
64 bytes from 10.9.0.1: icmp_seq=3 ttl=64 time=0.148 ms
64 bytes from 10.9.0.1: icmp_seq=4 ttl=64 time=0.169 ms
64 bytes from 10.9.0.1: icmp_seq=5 ttl=64 time=0.197 ms
^C
--- 10.9.0.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4069ms
rtt min/avg/max/mdev = 0.093/0.156/0.197/0.035 ms
```

通过 dmesg 查看：

```
[69663.128894] *** PRE_ROUTING
[69663.128896]     192.168.225.2  --> 192.168.225.136 (UDP)
[69663.128902] *** LOCAL_IN
[69663.128903]     192.168.225.2  --> 192.168.225.136 (UDP)
[69663.129147] *** LOCAL_OUT
[69663.129148]     127.0.0.53  --> 127.0.0.1 (UDP)
[69663.129152] *** POST_ROUTING
[69663.129153]     127.0.0.53  --> 127.0.0.1 (UDP)
```

在 10.9.0.5 上 ping 192.168.60.5：

```
root@e10d24587811:/# ping 192.168.60.5
PING 192.168.60.5 (192.168.60.5) 56(84) bytes of data.
64 bytes from 192.168.60.5: icmp_seq=1 ttl=63 time=0.205 ms
64 bytes from 192.168.60.5: icmp_seq=2 ttl=63 time=0.247 ms
64 bytes from 192.168.60.5: icmp_seq=3 ttl=63 time=0.250 ms
64 bytes from 192.168.60.5: icmp_seq=4 ttl=63 time=0.113 ms
64 bytes from 192.168.60.5: icmp_seq=5 ttl=63 time=0.192 ms
^C
--- 192.168.60.5 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4099ms
rtt min/avg/max/mdev = 0.113/0.201/0.250/0.049 ms
```

通过 dmesg 查看信息：

```
[69823.860163] *** PRE_ROUTING
[69823.860163]     10.9.0.5  --> 192.168.60.5 (ICMP)
[69823.860165] *** FORWARD
[69823.860166]     10.9.0.5  --> 192.168.60.5 (ICMP)
[69823.860167] *** POST_ROUTING
[69823.860168]     10.9.0.5  --> 192.168.60.5 (ICMP)
```

```
[69824.885717] *** PRE_ROUTING
[69824.885719]     192.168.60.5  --> 10.9.0.5 (ICMP)
[69824.885721] *** FORWARD
[69824.885722]     192.168.60.5  --> 10.9.0.5 (ICMP)
[69824.885724] *** POST_ROUTING
[69824.885725]     192.168.60.5  --> 10.9.0.5 (ICMP)
```

可以推测出 hook 的触发条件分别为：

NF_INET_LOCAL_OUT	由本地产生的包
NF_INET_POST_ROUTING	向外部网络发送的包
NF_INET_PRE_ROUTING	本地接收到的包，用于判断是否向外转发
NF_INET_LOCAL_IN	发往本地而不需要转发的包
NF_INET_FORWARD	需要向外转发的数据包

3. 增加两个 hook 函数，分别用于拦截 icmp 报文和 telnet 报文：

```
static struct nf_hook_ops hook1, hook2;

unsigned int telnetFilter(void* priv, struct sk_buff* skb, const struct nf_hook_state*
state)
{
        struct iphdr* iph;
        struct tcphdr* tcph;
        u16 port=23;
        iph=ip_hdr(skb);
        //tcph=(void*)iph+iph->ihl*4;

        if(iph->protocol==IPPROTO_TCP)
        {
                tcph=tcp_hdr(skb);
                if(ntohs(tcph->dest)==port)
                {
                        printk(KERN_WARNING"***Dropping %pI4 (TCP), port %d\n",&(iph-
>daddr),port);
                }
                return NF_DROP;
        }
        else
        {
                return NF_ACCEPT;
        }
}
```

```
unsigned int icmpFilter(void* priv, struct sk_buff* skb, const struct nf_hook_state* state)
{
        struct iphdr* iph;
        iph=ip_hdr(skb);
        if(iph->protocol==IPPROTO_ICMP)
        {
                printk(KERN_WARNING"***Dropping %pI4 (ICMP)\n",&(iph->daddr));
                return NF_DROP;
        }
        return NF_ACCEPT;
}
```

修改 registerFilter 和 removeFilter 函数：

```
int registerFilter(void) {
   printk(KERN_INFO "Registering filters.\n");

   hook1.hook = telnetFilter;
   hook1.hooknum = NF_INET_LOCAL_OUT;
   hook1.pf = PF_INET;
   hook1.priority = NF_IP_PRI_FIRST;
   nf_register_net_hook(&init_net, &hook1);

   hook2.hook = icmpFilter;
   hook2.hooknum = NF_INET_LOCAL_OUT;
   hook2.pf = PF_INET;
   hook2.priority = NF_IP_PRI_FIRST;
   nf_register_net_hook(&init_net, &hook2);

   return 0;
}

void removeFilter(void) {
   printk(KERN_INFO "The filters are being removed.\n");
   nf_unregister_net_hook(&init_net, &hook1);
   nf_unregister_net_hook(&init_net, &hook2);
}
```

重新编译并加载模块：

```
[07/26/21]seed@VM:~/.../packet_filter$ sudo rmmod seedFilter
[07/26/21]seed@VM:~/.../packet_filter$ sudo insmod seedFilter.ko
[07/26/21]seed@VM:~/.../packet_filter$ lsmod | grep seedFilter
seedFilter             16384  0
```

在 hostA 上 telnet 10.9.0.1：

```
root@e10d24587811:/# telnet 10.9.0.1
Trying 10.9.0.1...
^C
```

在 hostA 上 ping 10.9.0.1：

```
root@e10d24587811:/# ping 10.9.0.1
PING 10.9.0.1 (10.9.0.1) 56(84) bytes of data.
^C
--- 10.9.0.1 ping statistics ---
11 packets transmitted, 0 received, 100% packet loss, time 10245ms
```

均失败。

通过 dmesg 查看：

```
[76941.102031] ***Dropping 10.9.0.5 (ICMP)
[76942.126874] ***Dropping 10.9.0.5 (ICMP)
[76943.149366] ***Dropping 10.9.0.5 (ICMP)
[76944.173381] ***Dropping 10.9.0.5 (ICMP)
[76945.197553] ***Dropping 10.9.0.5 (ICMP)
[76946.220916] ***Dropping 10.9.0.5 (ICMP)
[76947.247259] ***Dropping 10.9.0.5 (ICMP)
[76948.267393] ***Dropping 10.9.0.5 (ICMP)
[76949.293856] ***Dropping 10.9.0.5 (ICMP)
[76950.317017] ***Dropping 10.9.0.5 (ICMP)
[76951.341457] ***Dropping 10.9.0.5 (ICMP)
```

发现包均被丢弃。拦截成功。

Task 2.A

在 hostA 上 ping 网关，可以 ping 通：

```
root@e10d24587811:/# ping 10.9.0.11
PING 10.9.0.11 (10.9.0.11) 56(84) bytes of data.
64 bytes from 10.9.0.11: icmp_seq=1 ttl=64 time=13.7 ms
64 bytes from 10.9.0.11: icmp_seq=2 ttl=64 time=0.065 ms
64 bytes from 10.9.0.11: icmp_seq=3 ttl=64 time=0.061 ms
64 bytes from 10.9.0.11: icmp_seq=4 ttl=64 time=0.075 ms
64 bytes from 10.9.0.11: icmp_seq=5 ttl=64 time=0.073 ms
64 bytes from 10.9.0.11: icmp_seq=6 ttl=64 time=0.121 ms
^C
--- 10.9.0.11 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5104ms
rtt min/avg/max/mdev = 0.061/2.354/13.731/5.087 ms
```

输入如下规则：

```
root@6bba5316370b:/# iptables -A OUTPUT -p icmp --icmp-type echo-reply -j ACCEPT
root@6bba5316370b:/# iptables -A INPUT -p icmp --icmp-type echo-request -j ACCEPT
root@6bba5316370b:/# iptables -t filter -L -n
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     icmp --  0.0.0.0/0            0.0.0.0/0            icmptype 8

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     icmp --  0.0.0.0/0            0.0.0.0/0            icmptype 0
root@6bba5316370b:/# iptables -P OUTPUT DROP
root@6bba5316370b:/# iptables -P INPUT DROP
```

在 hostA 中 ping 10.9.0.11：

```
root@c80df0d479ac:/# ping 10.9.0.11
PING 10.9.0.11 (10.9.0.11) 56(84) bytes of data.
64 bytes from 10.9.0.11: icmp_seq=1 ttl=64 time=0.097 ms
64 bytes from 10.9.0.11: icmp_seq=2 ttl=64 time=0.077 ms
64 bytes from 10.9.0.11: icmp_seq=3 ttl=64 time=0.063 ms
64 bytes from 10.9.0.11: icmp_seq=4 ttl=64 time=0.070 ms
^C
--- 10.9.0.11 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3071ms
rtt min/avg/max/mdev = 0.063/0.076/0.097/0.012 ms
```

在 hostA 中 telnet 10.9.0.11：

```
root@c80df0d479ac:/# telnet 10.9.0.11
Trying 10.9.0.11...
^C
```

规则编写正确。

Task 2.B

为了保护内网，需要对 ICMP 流量做出如下限制：

（1） 外部主机不能 ping 内部主机；

（2） 外部主机可以 ping 网关；

（3） 内部主机可以 ping 外部主机；

（4） 其他在内外网之间的包应被阻塞。

规则如下：

（1） OUTPUT INPUT FORWARD 丢弃，不允许内外流量交互：

```
iptables -P OUTPUT DROP
iptables -P INPUT DROP
iptables -P FORWARD DROP
```

（2） 对于 FORWARD 只有 icmp 请求报文由内部端口 eth1 进入，外部端口 eth0 流出，才接收：

```
iptables -A FORWARD -p icmp --icmp-type echo-request -i eth1 -j ACCEPT
iptables -A FORWARD -p icmp --icmp-type echo-request -o eth0 -j ACCEPT
```

只有 icmp 应答报文由外部端口进入，内部端口流出，才接收：

```
iptables -A FORWARD -p icmp --icmp-type echo-reply -i eth0 -j ACCEPT
iptables -A FORWARD -p icmp --icmp-type echo-reply -o eth1 -j ACCEPT
```

（3） 对于 input 和 output，允许输入的 icmp 请求和应答报文，保证外部主机能够 ping 路由器：

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

```
root@f039eaeb3765:/# iptables -A INPUT -p icmp --icmp-type echo-request -j ACCEPT
root@f039eaeb3765:/# iptables -A OUTPUT -p icmp --icmp-type echo-reply -j ACCEPT
root@f039eaeb3765:/# iptables -A FORWARD -p icmp --icmp-type echo-request -i eth1 -j ACCEPT
root@f039eaeb3765:/# iptables -A FORWARD -p icmp --icmp-type echo-request -o eth0 -j ACCEPT
root@f039eaeb3765:/# iptables -A FORWARD -p icmp --icmp-type echo-reply -i eth0 -j ACCEPT
root@f039eaeb3765:/# iptables -A FORWARD -p icmp --icmp-type echo-reply -o eth1 -j ACCEPT
root@f039eaeb3765:/# iptables -P OUTPUT DROP
root@f039eaeb3765:/# iptables -P INPUT DROP
root@f039eaeb3765:/# iptables -P FORWARD DROP
root@f039eaeb3765:/# iptables -L
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     icmp --  anywhere             anywhere             icmp echo-request

Chain FORWARD (policy DROP)
target     prot opt source               destination
ACCEPT     icmp --  anywhere             anywhere             icmp echo-request
ACCEPT     icmp --  anywhere             anywhere             icmp echo-request
ACCEPT     icmp --  anywhere             anywhere             icmp echo-reply
ACCEPT     icmp --  anywhere             anywhere             icmp echo-reply

Chain OUTPUT (policy DROP)
target     prot opt source               destination
ACCEPT     icmp --  anywhere             anywhere             icmp echo-reply
```

测试规则：

```
[07/26/21]seed@VM:~/.../Labsetup$ dockps
6c95b2d2174e  host3-192.168.60.7
c80df0d479ac  hostA-10.9.0.5
1832cf083396  host1-192.168.60.5
36c63fa8bd7b  host2-192.168.60.6
f039eaeb3765  seed-router
```

（1） 外网 ping 内网：

```
root@c80df0d479ac:/# ping 192.168.60.5
PING 192.168.60.5 (192.168.60.5) 56(84) bytes of data.

--- 192.168.60.5 ping statistics ---
15 packets transmitted, 0 received, 100% packet loss, time 14318ms
```

ping 不成功

（2） 内网 ping 外网：

```
root@1832cf083396:/# ping 10.9.0.5
PING 10.9.0.5 (10.9.0.5) 56(84) bytes of data.
64 bytes from 10.9.0.5: icmp_seq=1 ttl=63 time=47.1 ms
64 bytes from 10.9.0.5: icmp_seq=2 ttl=63 time=0.062 ms
64 bytes from 10.9.0.5: icmp_seq=3 ttl=63 time=0.129 ms
64 bytes from 10.9.0.5: icmp_seq=4 ttl=63 time=0.114 ms
64 bytes from 10.9.0.5: icmp_seq=5 ttl=63 time=0.125 ms
64 bytes from 10.9.0.5: icmp_seq=6 ttl=63 time=0.286 ms
^C
--- 10.9.0.5 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5098ms
rtt min/avg/max/mdev = 0.062/7.975/47.137/17.513 ms
```

成功。

（3） 外网 ping 网关：

```
^Croot@c80df0d479ac:/# ping 10.9.0.11
PING 10.9.0.11 (10.9.0.11) 56(84) bytes of data.
64 bytes from 10.9.0.11: icmp_seq=1 ttl=64 time=0.105 ms
64 bytes from 10.9.0.11: icmp_seq=2 ttl=64 time=0.063 ms
64 bytes from 10.9.0.11: icmp_seq=3 ttl=64 time=0.063 ms
64 bytes from 10.9.0.11: icmp_seq=4 ttl=64 time=0.065 ms
^C
--- 10.9.0.11 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3054ms
rtt min/avg/max/mdev = 0.063/0.074/0.105/0.017 ms
```

成功。

（4） 外网 telnet 内网：

```
root@c80df0d479ac:/# telnet 192.168.60.5
Trying 192.168.60.5...
^C
```

telnet 失败。

（5） 内网 telnet 内网：

```
root@1832cf083396:/# telnet 10.9.0.5
Trying 10.9.0.5...
^C
```

telnet 失败。

Task 2.c

规则：

只允许 192.168.60.5 的 23 端口的流量进行转发，保证能被外部和内部主机登录，而外部主机无法连接到内部主机；

本地主机的相互访问不需要转发，因此 FORWARD 对其他报文的丢弃不会影响内部主机间的登录。

```
root@f039eaeb3765:/# iptables -A FORWARD -p tcp --sport 23 -d 192.168.60.5 -j ACCEPT
root@f039eaeb3765:/# iptables -A FORWARD -p tcp --dport 23 -d 192.168.60.5 -j ACCEPT
root@f039eaeb3765:/# iptables -A FORWARD -p tcp -s 192.168.60.0/24 -d 192.168.60.0/24 -j ACCEPT
root@f039eaeb3765:/# iptables -A FORWARD -p tcp -i eth1 -o eth0 --sport 23 -j ACCEPT
root@f039eaeb3765:/# iptables -t filter -L -n
Chain INPUT (policy DROP)
target     prot opt source               destination

Chain FORWARD (policy DROP)
target     prot opt source               destination
ACCEPT     tcp  --  0.0.0.0/0            192.168.60.5         tcp spt:23
ACCEPT     tcp  --  0.0.0.0/0            192.168.60.5         tcp dpt:23
ACCEPT     tcp  --  192.168.60.0/24      192.168.60.0/24
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0            tcp spt:23

Chain OUTPUT (policy DROP)
target     prot opt source               destination
```

测试：

（1） 内部主机 telnet 内部服务器：

```
root@1832cf083396:/# telnet 192.168.60.6
Trying 192.168.60.6...
Connected to 192.168.60.6.
Escape character is '^]'.
Ubuntu 20.04.1 LTS
36c63fa8bd7b login: seed
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-54-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

seed@36c63fa8bd7b:~$
```

成功。

（2） 内部主机 ping 内部主机：

```
root@1832cf083396:/# ping 192.168.60.6
PING 192.168.60.6 (192.168.60.6) 56(84) bytes of data.
64 bytes from 192.168.60.6: icmp_seq=1 ttl=64 time=30.5 ms
64 bytes from 192.168.60.6: icmp_seq=2 ttl=64 time=0.062 ms
64 bytes from 192.168.60.6: icmp_seq=3 ttl=64 time=0.096 ms
^C
--- 192.168.60.6 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2032ms
rtt min/avg/max/mdev = 0.062/10.221/30.506/14.343 ms
```

成功。

（3） 外部主机 ping 内部主机：

```
root@c80df0d479ac:/# ping 192.168.60.5
PING 192.168.60.5 (192.168.60.5) 56(84) bytes of data.
^C
--- 192.168.60.5 ping statistics ---
7 packets transmitted, 0 received, 100% packet loss, time 6179ms
```

失败。

（4） 内部主机 ping 外部主机：

```
root@1832cf083396:/# ping 10.9.0.5
PING 10.9.0.5 (10.9.0.5) 56(84) bytes of data.
^C
--- 10.9.0.5 ping statistics ---
5 packets transmitted, 0 received, 100% packet loss, time 4088ms
```

失败。

（5） 外部主机 telnet192.168.60.5：

```
root@c80df0d479ac:/# telnet 192.168.60.5
Trying 192.168.60.5...
Connected to 192.168.60.5.
Escape character is '^]'.
Ubuntu 20.04.1 LTS
1832cf083396 login: seed
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-54-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

seed@1832cf083396:~$
```

成功。

（6） 外部主机 telnet 内部其他主机：

```
root@c80df0d479ac:/# telnet 192.168.60.6
Trying 192.168.60.6...
^C
```

失败。

Task 3.A：

ICMP：

在 10.9.0.5 上 ping 192.168.60.5：

```
root@33efd669b6b5:/# ping 192.168.60.5
PING 192.168.60.5 (192.168.60.5) 56(84) bytes of data.
64 bytes from 192.168.60.5: icmp_seq=1 ttl=63 time=0.315 ms
64 bytes from 192.168.60.5: icmp_seq=2 ttl=63 time=0.115 ms
64 bytes from 192.168.60.5: icmp_seq=3 ttl=63 time=0.131 ms
64 bytes from 192.168.60.5: icmp_seq=4 ttl=63 time=0.131 ms
64 bytes from 192.168.60.5: icmp_seq=5 ttl=63 time=0.099 ms
64 bytes from 192.168.60.5: icmp_seq=6 ttl=63 time=0.107 ms
```

在网关查看连接信息，持续时间为 29 秒：

```
root@bca5690e4073:/# conntrack -L
icmp     1 29 src=10.9.0.5 dst=192.168.60.5 type=8 code=0 id=29 src=192.168.60.5 dst=10.9.0.5 ty
pe=0 code=0 id=29 mark=0 use=1
conntrack v1.4.5 (conntrack-tools): 1 flow entries have been shown.
root@bca5690e4073:/#
```

UDP：

在 192.168.60.5 中输入如下命令：

```
[07/26/21]seed@VM:~/.../Labsetup$ docksh b7
root@b7c274bb50eb:/# nc -lu 9090
```

在 10.9.0.5 中输入如下命令：

```
root@33efd669b6b5:/# nc -u 192.168.60.5 9090
```

连接后，在网关查看连接信息，持续时间为 29 秒：

```
root@bca5690e4073:/# conntrack -L
icmp     1 29 src=10.9.0.5 dst=192.168.60.5 type=8 code=0 id=29 src=192.168.60.5 dst=10.9.0.5 ty
```

TCP：

在 192.168.60.5 输入如下命令：

```
root@b7c274bb50eb:/# nc -l 9090
```

在 10.9.0.5 输入如下命令：

```
root@33efd669b6b5:/# nc 192.168.60.5 9090
abc
```

建立连接之后，在网关查看连接信息：

```
root@bca5690e4073:/# conntrack -L
tcp      6 431995 ESTABLISHED src=10.9.0.5 dst=192.168.60.5 sport=51658 dport=9090 src=192.168.6
0.5 dst=10.9.0.5 sport=9090 dport=51658 [ASSURED] mark=0 use=1
conntrack v1.4.5 (conntrack-tools): 1 flow entries have been shown.
```

持续时间为 431995 秒。

Task 3.B

编写如下规则：

```
root@bca5690e4073:/# iptables -F
root@bca5690e4073:/# iptables -A FORWARD -p tcp -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
root@bca5690e4073:/# iptables -A FORWARD -p tcp --dport 23 -d 192.168.60.5 --syn -m conntrack --ctstate NEW
 -j ACCEPT
root@bca5690e4073:/# iptables -A FORWARD -p tcp --dport 23 -d 10.9.0.0/24 --syn -m conntrack --ctstate NEW
-j ACCEPT
root@bca5690e4073:/# iptables -P FORWARD DROP
```

测试：

（1）在 10.9.0.5 上 telnet 192.168.60.5：

```
root@33efd669b6b5:/# telnet 192.168.60.5
Trying 192.168.60.5...
Connected to 192.168.60.5.
Escape character is '^]'.
Ubuntu 20.04.1 LTS
b7c274bb50eb login: seed
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-54-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.
```

成功。

（2） 在 10.9.0.5 上 telnet 内网其他主机（192.168.60.6）：

```
root@33efd669b6b5:/# telnet 192.168.60.6
Trying 192.168.60.6...
^C
```

（3）内网主机 telnet 外网主机：

```
root@b7c274bb50eb:/# telnet 10.9.0.5
Trying 10.9.0.5...
Connected to 10.9.0.5.
Escape character is '^]'.
Ubuntu 20.04.1 LTS
33efd669b6b5 login: seed
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-54-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.
```

成功。

（4） 网内主机 telnet 网内主机：

```
root@b7c274bb50eb:/# telnet 192.168.60.6
Trying 192.168.60.6...
Connected to 192.168.60.6.
Escape character is '^]'.
Ubuntu 20.04.1 LTS
91e730419bf5 login: seed
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-54-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.
```

成功。

（5） 外部主机无法与内部主机建立连接：

```
root@b7c274bb50eb:/# nc -l 9090
```

```
root@33efd669b6b5:/# nc 192.168.60.5 9090
abc
```

Task 4：

编写如下规则：

```
root@bca5690e4073:/# iptables -F
root@bca5690e4073:/# iptables -A FORWARD -s 10.9.0.5 -m limit --limit 10/minute --limit-burst 5 -j ACCEPT
root@bca5690e4073:/# iptables -A FORWARD -s 10.9.0.5 -j DROP
root@bca5690e4073:/# iptables -P FORWARD ACCEPT
```

在 10.9.0.5 上 ping 192.168.60.5：

```
root@33efd669b6b5:/# ping 192.168.60.5
PING 192.168.60.5 (192.168.60.5) 56(84) bytes of data.
64 bytes from 192.168.60.5: icmp_seq=1 ttl=63 time=0.133 ms
64 bytes from 192.168.60.5: icmp_seq=2 ttl=63 time=0.201 ms
64 bytes from 192.168.60.5: icmp_seq=3 ttl=63 time=0.140 ms
64 bytes from 192.168.60.5: icmp_seq=4 ttl=63 time=0.084 ms
64 bytes from 192.168.60.5: icmp_seq=5 ttl=63 time=0.122 ms
64 bytes from 192.168.60.5: icmp_seq=7 ttl=63 time=0.113 ms
64 bytes from 192.168.60.5: icmp_seq=13 ttl=63 time=0.130 ms
64 bytes from 192.168.60.5: icmp_seq=19 ttl=63 time=0.173 ms
64 bytes from 192.168.60.5: icmp_seq=25 ttl=63 time=0.098 ms
^C
--- 192.168.60.5 ping statistics ---
28 packets transmitted, 9 received, 67.8571% packet loss, time 27637ms
rtt min/avg/max/mdev = 0.084/0.132/0.201/0.034 ms
```

前几个报文速度较快，后面开始速度变慢，平均 6s 一个，说明规则设定正确。

去掉第二条规则：

```
root@bca5690e4073:/# iptables -F
root@bca5690e4073:/# iptables -A FORWARD -s 10.9.0.5 -m limit --limit 10/minute --limit-burst 5 -j ACCEPT
root@bca5690e4073:/# iptables -P FORWARD ACCEPT
```

重新 ping：

```
root@33efd669b6b5:/# ping 192.168.60.5
PING 192.168.60.5 (192.168.60.5) 56(84) bytes of data.
64 bytes from 192.168.60.5: icmp_seq=1 ttl=63 time=0.136 ms
64 bytes from 192.168.60.5: icmp_seq=2 ttl=63 time=0.138 ms
64 bytes from 192.168.60.5: icmp_seq=3 ttl=63 time=0.105 ms
64 bytes from 192.168.60.5: icmp_seq=4 ttl=63 time=0.564 ms
64 bytes from 192.168.60.5: icmp_seq=5 ttl=63 time=0.156 ms
64 bytes from 192.168.60.5: icmp_seq=6 ttl=63 time=0.138 ms
64 bytes from 192.168.60.5: icmp_seq=7 ttl=63 time=0.099 ms
64 bytes from 192.168.60.5: icmp_seq=8 ttl=63 time=0.106 ms
64 bytes from 192.168.60.5: icmp_seq=9 ttl=63 time=0.089 ms
64 bytes from 192.168.60.5: icmp_seq=10 ttl=63 time=0.126 ms
64 bytes from 192.168.60.5: icmp_seq=11 ttl=63 time=0.135 ms
64 bytes from 192.168.60.5: icmp_seq=12 ttl=63 time=0.096 ms
64 bytes from 192.168.60.5: icmp_seq=13 ttl=63 time=0.200 ms
64 bytes from 192.168.60.5: icmp_seq=14 ttl=63 time=0.118 ms
64 bytes from 192.168.60.5: icmp_seq=15 ttl=63 time=0.095 ms
^C
--- 192.168.60.5 ping statistics ---
15 packets transmitted, 15 received, 0% packet loss, time 14305ms
rtt min/avg/max/mdev = 0.089/0.153/0.564/0.113 ms
```

可以看出，报文发送速度较快，并没有减慢，说明规则失效。

原因：第二条规则的作用是将报文默认设置为 DROP，去掉之后，所有保温都会从默认的 ACCEPT 规则中通过。

Task 5：

在 192.168.60.5，192.168.60.6，192.168.60.7 中均输入如下命令：

```
root@b7c274bb50eb:/# nc -luk 8080
```

（1） 轮询模式：

在网关中输入如下规则：

```
root@bca5690e4073:/# iptables -t nat -A PREROUTING -p udp --dport 8080 -m statistic --mode nth --every 3 --packet 0 -j DNAT --to-destination 192.168.60.5:8080
root@bca5690e4073:/# iptables -t nat -A PREROUTING -p udp --dport 8080 -m statistic --mode nth --every 2 --packet 0 -j DNAT --to-destination 192.168.60.6:8080
root@bca5690e4073:/# iptables -t nat -A PREROUTING -p udp --dport 8080 -j DNAT --to-destination 192.168.60.7:8080
```

每三个报文中，第一个发送到 192.168.60.5 的 8080 端口，第二个发送到 192.168.60.6 的 8080 端口，第三个发送到 192.168.60.7 的 8080 端口。

在 10.9.0.5 上输入如下命令：

```
root@33efd669b6b5:/# echo hello1 | nc -u 10.9.0.11 8080
^C
root@33efd669b6b5:/# echo hello2 | nc -u 10.9.0.11 8080
^C
root@33efd669b6b5:/# echo hello3 | nc -u 10.9.0.11 8080
^C
root@33efd669b6b5:/# echo hello4 | nc -u 10.9.0.11 8080
^C
root@33efd669b6b5:/# echo hello5 | nc -u 10.9.0.11 8080
^C
root@33efd669b6b5:/# echo hello6 | nc -u 10.9.0.11 8080
^C
```

192.168.60.5 中收到的内容：

```
root@b7c274bb50eb:/# nc -luk 8080
hello1
hello4
```

192.168.60.6 中收到的内容：

```
root@91e730419bf5:/# nc -luk 8080
hello2
hello5
```

192.168.60.7 中收到的内容：

```
root@c2cd3eb50a75:/# nc -luk 8080
hello3
hello6
```

（2） 随机模式：

```
root@bca5690e4073:/# iptables -F
root@bca5690e4073:/# iptables -t nat -A PREROUTING -p udp --dport 8080 -m statistic --mode random --probability 0.33 -j DNAT --to-destination 192.168.60.5:8080
root@bca5690e4073:/# iptables -t nat -A PREROUTING -p udp --dport 8080 -m statistic --mode random --probability 0.5 -j DNAT --to-destination 192.168.60.6:8080
root@bca5690e4073:/# iptables -t nat -A PREROUTING -p udp --dport 8080 -j DNAT --to-destination 192.168.60.7:8080
```

到达的报文以 0.33 的概率发送到 192.168.60.5 的 8080 端口。

如果没有发送到 192.168.60.5 的 8080 端口，则以 0.5 的概率发送到 192.168.60.6 的 8080 端口。

如果没有发送到 192.168.60.5 或 192.168.60.6，则发送到 192.168.60.7 的 8080 端口。

在 10.9.0.5 上输入如下命令：

```
root@33efd669b6b5:/# echo hello1 | nc -u 10.9.0.11 8080
^C
root@33efd669b6b5:/# echo hello2 | nc -u 10.9.0.11 8080
^C
root@33efd669b6b5:/# echo hello3 | nc -u 10.9.0.11 8080
^C
root@33efd669b6b5:/# echo hello4 | nc -u 10.9.0.11 8080
^C
root@33efd669b6b5:/# echo hello5 | nc -u 10.9.0.11 8080
^C
root@33efd669b6b5:/# echo hello6 | nc -u 10.9.0.11 8080
^C
root@33efd669b6b5:/# echo hello7 | nc -u 10.9.0.11 8080
^C
root@33efd669b6b5:/# echo hello8 | nc -u 10.9.0.11 8080
^C
root@33efd669b6b5:/# echo hello9 | nc -u 10.9.0.11 8080
^C
```

192.168.60.5 中收到的内容：

```
root@b7c274bb50eb:/# nc -luk 8080
hello1
hello4
hello7
```

192.168.60.6 中收到的内容：

```
root@91e730419bf5:/# nc -luk 8080
hello2
hello5
hello8
```

192.168.60.7 中收到的内容：

```
root@c2cd3eb50a75:/# nc -luk 8080
hello3
hello6
hello9
```

实现了负载均衡。