Lab 6. Report

57118105

Task 1.A

①在 kernel-module 中编译内核：

```
[07/27/21]seed@VM:~/.../kernel_module$ make
make -C /lib/modules/5.4.0-54-generic/build M=/home/seed/Desktop/Labs_20.04/
Network/Firewall/Labsetup/Files/kernel_module modules
make[1]: Entering directory '/usr/src/linux-headers-5.4.0-54-generic'
  Building modules, stage 2.
  MODPOST 1 modules
WARNING: modpost: missing MODULE_LICENSE() in /home/seed/Desktop/Labs_20.04/
Network/Firewall/Labsetup/Files/kernel_module/hello.o
see include/linux/module.h for more information
make[1]: Leaving directory '/usr/src/linux-headers-5.4.0-54-generic'
```

②加载模块：

```
[07/27/21]seed@VM:~/.../kernel_module$ sudo insmod hello.ko
```

③使用 dmesg 命令查看：

```
[07/27/21]seed@VM:~/.../kernel_module$ dmesg
[    0.000000] Linux version 5.4.0-54-generic (buildd@lcy01-amd64-024) (gcc
version 9.3.0 (Ubuntu 9.3.0-17ubuntu1~20.04)) #60-Ubuntu SMP Fri Nov 6 10:37
:59 UTC 2020 (Ubuntu 5.4.0-54.60-generic 5.4.65)
[    0.000000] Command line: BOOT_IMAGE=/boot/vmlinuz-5.4.0-54-generic root=
UUID=a91f1a43-2770-4684-9fc3-b7abfd786c1d ro quiet splash
[    0.000000] KERNEL supported cpus:
[    0.000000]   Intel GenuineIntel
[    0.000000]   AMD AuthenticAMD
[    0.000000]   Hygon HygonGenuine
[ 1828.753410] Hello World!
[ 1846.436888] Bye-bye World!.
[ 2062.857048] Hello World!
```

可见 Hello World! 信息。

④列出模块：

```
[07/27/21]seed@VM:~/.../kernel_module$ lsmod | grep hello
hello                  16384  0
```

⑤查看模块信息：

```
[07/27/21]seed@VM:~/.../kernel_module$ modinfo hello.ko
filename:       /home/seed/Desktop/Labs_20.04/Network/Firewall/Labsetup/File
s/kernel_module/hello.ko
srcversion:     75A5408065DE2CED836C338
depends:
retpoline:      Y
name:           hello
vermagic:       5.4.0-54-generic SMP mod_unload
```

⑥删除模块：

```
[07/27/21]seed@VM:~/.../kernel_module$ sudo rmmod hello.ko
```

⑦使用 dmesg 查看：

```
[07/27/21]seed@VM:~/.../kernel_module$ dmesg | grep World
[ 1828.753410] Hello World!
[ 1846.436888] Bye-bye World!.
[ 2062.857048] Hello World!
[ 2116.969576] Bye-bye World!.
[07/27/21]seed@VM:~/.../kernel_module$
```

Task 1.B.1

①首先对 8.8.8.8 进行 dig 尝试：

```
[07/27/21]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
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 8192
;; 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.        20111   IN      A       93.184.216.34

;; Query time: 256 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Tue Jul 27 12:46:14 EDT 2021
;; MSG SIZE  rcvd: 60
```

②加载内核模块：

```
[07/27/21]seed@VM:~/.../packet_filter$ make
make -C /lib/modules/5.4.0-54-generic/build M=/home/seed/Desktop/Labs_20.04/
Network/Firewall/Labsetup/Files/packet_filter modules
make[1]: Entering directory '/usr/src/linux-headers-5.4.0-54-generic'
  CC [M]  /home/seed/Desktop/Labs_20.04/Network/Firewall/Labsetup/Files/pack
et_filter/seedFilter.o
  Building modules, stage 2.
  MODPOST 1 modules
  CC [M]  /home/seed/Desktop/Labs_20.04/Network/Firewall/Labsetup/Files/pack
et_filter/seedFilter.mod.o
  LD [M]  /home/seed/Desktop/Labs_20.04/Network/Firewall/Labsetup/Files/pack
et_filter/seedFilter.ko
make[1]: Leaving directory '/usr/src/linux-headers-5.4.0-54-generic'
[07/27/21]seed@VM:~/.../packet_filter$ sudo insmod seedFilter.ko
```

③再次尝试 dig 8.8.8.8：

```
[07/27/21]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

[07/27/21]seed@VM:~/.../packet_filter$
```

可见被拒绝，防火墙已生效。

④从内核中移除：

```
[07/27/21]seed@VM:~/.../packet_filter$ sudo rmmod seedFilter
[07/27/21]seed@VM:~/.../packet_filter$ lsmod | grep seedFilter
[07/27/21]seed@VM:~/.../packet_filter$
```

Task 1.B.2

①定义 5 个一一对应的钩子：

```
hook1.hook = printInfo;
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 = printInfo;
hook2.hooknum = NF_INET_POST_ROUTING;
hook2.pf = PF_INET;
hook2.priority = NF_IP_PRI_FIRST;
nf_register_net_hook(&init_net, &hook2);

hook3.hook = printInfo;
hook3.hooknum = NF_INET_POST_ROUTING;
hook3.pf = PF_INET;
hook3.priority = NF_IP_PRI_FIRST;
nf_register_net_hook(&init_net, &hook3);

hook4.hook = printInfo;
hook4.hooknum = NF_INET_POST_ROUTING;
hook4.pf = PF_INET;
hook4.priority = NF_IP_PRI_FIRST;
nf_register_net_hook(&init_net, &hook4);

hook5.hook = printInfo;
hook5.hooknum = NF_INET_POST_ROUTING;
hook5.pf = PF_INET;
hook5.priority = NF_IP_PRI_FIRST;
nf_register_net_hook(&init_net, &hook5);
```

②使用 make 编译内核：

```
[07/27/21]seed@VM:~/.../packet_filter$ make
make -C /lib/modules/5.4.0-54-generic/build M=/home/seed/Desktop/Labs_20.04/
Network/Firewall/Labsetup/Files/packet_filter modules
make[1]: Entering directory '/usr/src/linux-headers-5.4.0-54-generic'
  CC [M]  /home/seed/Desktop/Labs_20.04/Network/Firewall/Labsetup/Files/pack
et_filter/seedFilter.o
  Building modules, stage 2.
  MODPOST 1 modules
  CC [M]  /home/seed/Desktop/Labs_20.04/Network/Firewall/Labsetup/Files/pack
et_filter/seedFilter.mod.o
  LD [M]  /home/seed/Desktop/Labs_20.04/Network/Firewall/Labsetup/Files/pack
et_filter/seedFilter.ko
make[1]: Leaving directory '/usr/src/linux-headers-5.4.0-54-generic'
```

③使用 sudo insmod seedFilter.ko 加载内核：

```
[07/28/21]seed@VM:~/.../packet_filter$ sudo insmod seedFilter.ko
```

④使用 lsmod | grep seedFilter 查看模块：

```
[07/28/21]seed@VM:~/.../packet_filter$ lsmod | grep seedFilter
seedFilter             16384  0
```

⑤dig @8.8.8.8：

```
[07/28/21]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
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 41183
;; 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.        15708   IN      A       93.184.216.34

;; Query time: 48 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Wed Jul 28 06:36:49 EDT 2021
;; MSG SIZE  rcvd: 60
```

⑥使用 sudo dmesg -c 查看：

```
[22004.111468] Registering filters.
[22024.821770] *** LOCAL_OUT
[22024.821772]     127.0.0.1  --> 127.0.0.1 (UDP)
[22024.821778] *** POST_ROUTING
[22024.821779]     127.0.0.1  --> 127.0.0.1 (UDP)
[22024.821779] *** POST_ROUTING
[22024.821779]     127.0.0.1  --> 127.0.0.1 (UDP)
[22024.821780] *** POST_ROUTING
[22024.821780]     127.0.0.1  --> 127.0.0.1 (UDP)
[22024.821780] *** POST_ROUTING
[22024.821780]     127.0.0.1  --> 127.0.0.1 (UDP)
[22024.822114] *** LOCAL_OUT
[22024.822115]     192.168.115.130  --> 8.8.8.8 (UDP)
[22024.822120] *** POST_ROUTING
[22024.822120]     192.168.115.130  --> 8.8.8.8 (UDP)
[22024.822121] *** POST_ROUTING
[22024.822121]     192.168.115.130  --> 8.8.8.8 (UDP)
[22024.822121] *** POST_ROUTING
[22024.822122]     192.168.115.130  --> 8.8.8.8 (UDP)
[22024.822122] *** POST_ROUTING
[22024.822122]     192.168.115.130  --> 8.8.8.8 (UDP)
```

⑦运行 sudo rmmod seedFilter 从内核中移除模块：

```
[07/28/21]seed@VM:~/.../packet_filter$ sudo rmmod seedFilter
[07/28/21]seed@VM:~/.../packet_filter$
```

Task 1.C.3

①改编 seedFilter.c：

```c
#include <linux/kernel.h>
#include <linux/module.h>
#include <linux/netfilter.h>
#include <linux/netfilter_ipv4.h>
#include <linux/ip.h>
#include <linux/tcp.h>
#include <linux/udp.h>
#include <linux/icmp.h>
#include <linux/if_ether.h>
#include <linux/inet.h>

static struct nf_hook_ops hook1, hook2, hook3, hook4;

unsigned int blockUDP(void *priv, struct sk_buff *skb, const struct nf_hook_state *state)
{
    struct iphdr *iph;
    struct udphdr *udph;

    u16 port = 53;
    char ip[16] = "8.8.8.8";
    u32 ip_addr;

    if (!skb) return NF_ACCEPT;

    iph = ip_hdr(skb);
    // Convert the IPv4 address from dotted decimal to 32-bit binary
    in4_pton(ip, -1, (u8 *)&ip_addr, '\0', NULL);

    if (iph->protocol == IPPROTO_UDP) {
        udph = udp_hdr(skb);
        if (iph->daddr == ip_addr && ntohs(udph->dest) == port){
            printk(KERN_WARNING "*** Dropping %pI4 (UDP), port %d\n", &(iph->daddr), port);
            return NF_DROP;
        }
    }
    return NF_ACCEPT;
}

unsigned int blockTCP(void *priv, struct sk_buff *skb, const struct nf_hook_state *state)
{
    struct iphdr *iph;
    struct tcphdr *tcph;

    u16 port = 23;
    char ip[16] = "10.9.0.1";
    u32 ip_addr;

    if (!skb) return NF_ACCEPT;

    iph = ip_hdr(skb);
    // Convert the IPv4 address from dotted decimal to 32-bit binary
    in4_pton(ip, -1, (u8 *)&ip_addr, '\0', NULL);

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

```
61      return NF_ACCEPT;
62  }
63
64  unsigned int blockICMP(void *priv, struct sk_buff *skb, const struct
    nf_hook_state *state)
65  {
66      struct iphdr *iph;
67      struct icmphdr *icmph;
68
69      char ip[16] = "10.9.0.1";
70      u32 ip_addr;
71
72      if (!skb) return NF_ACCEPT;
73
74      iph = ip_hdr(skb);
75      // Convert the IPv4 address from dotted decimal to 32-bit binary
76      in4_pton(ip, -1, (u8 *)&ip_addr, '\0', NULL);
77
78      if (iph->protocol == IPPROTO_ICMP) {
79          icmph = icmp_hdr(skb);
80          if (iph->daddr == ip_addr){
81              printk(KERN_WARNING "*** Dropping %pI4 (ICMP)\n", &(iph-
    >daddr));
82              return NF_DROP;
83          }
84      }
85      return NF_ACCEPT;
86  }
87
88  unsigned int printInfo(void *priv, struct sk_buff *skb, const struct
    nf_hook_state *state)
89  {
90      struct iphdr *iph;
91      char *hook;
Wireshark char *protocol;
93
94      switch (state->hook){
95          case NF_INET_LOCAL_IN:     hook = "LOCAL_IN";     break;
96          case NF_INET_LOCAL_OUT:    hook = "LOCAL_OUT";    break;
97          case NF_INET_PRE_ROUTING:  hook = "PRE_ROUTING";  break;
98          case NF_INET_POST_ROUTING: hook = "POST_ROUTING"; break;
99          case NF_INET_FORWARD:      hook = "FORWARD";      break;
100         default:                   hook = "IMPOSSIBLE";   break;
101     }
102     printk(KERN_INFO "*** %s\n", hook); // Print out the hook info
103
104     iph = ip_hdr(skb);
105     switch (iph->protocol){
106         case IPPROTO_UDP:  protocol = "UDP";   break;
107         case IPPROTO_TCP:  protocol = "TCP";   break;
108         case IPPROTO_ICMP: protocol = "ICMP";  break;
109         default:           protocol = "OTHER"; break;
110     }
111     // Print out the IP addresses and protocol
112     printk(KERN_INFO " %pI4 --> %pI4 (%s)\n", &(iph->saddr), &(iph->daddr),
    protocol);
113
114     return NF_ACCEPT;
115 }
116
117 int registerFilter(void) {
118     printk(KERN_INFO "Registering filters.\n");
119
120     hook1.hook = printInfo;
```

```
121     hook1.hooknum = NF_INET_LOCAL_OUT;
122     hook1.pf = PF_INET;
123     hook1.priority = NF_IP_PRI_FIRST;
124     nf_register_net_hook(&init_net, &hook1);
125
126     hook2.hook = blockUDP;
127     hook2.hooknum = NF_INET_POST_ROUTING;
128     hook2.pf = PF_INET;
129     hook2.priority = NF_IP_PRI_FIRST;
130     nf_register_net_hook(&init_net, &hook2);
131
132     hook3.hook = blockICMP;
133     hook3.hooknum = NF_INET_PRE_ROUTING;
134     hook3.pf = PF_INET;
135     hook3.priority = NF_IP_PRI_FIRST;
136     nf_register_net_hook(&init_net, &hook3);
137
138     hook4.hook = blockTCP;
139     hook4.hooknum = NF_INET_PRE_ROUTING;
140     hook4.pf = PF_INET;
141     hook4.priority = NF_IP_PRI_FIRST;
142     nf_register_net_hook(&init_net, &hook4);
143
144     return 0;
145 }
146
147 void removeFilter(void) {
148     printk(KERN_INFO "The filters are being removed.\n");
149     nf_unregister_net_hook(&init_net, &hook1);
150     nf_unregister_net_hook(&init_net, &hook2);
151     nf_unregister_net_hook(&init_net, &hook3);
152     nf_unregister_net_hook(&init_net, &hook4);
153 }
154
155 module_init(registerFilter);
156 module_exit(removeFilter);
157
158 MODULE_LICENSE("GPL");
```

② 编译并加载：

```
[07/28/21]seed@VM:~/.../packet_filter$ make
make -C /lib/modules/5.4.0-54-generic/build M=/home/seed/Desktop/Labs_20.04/Netw
ork/Firewall/Labsetup/Files/packet_filter modules
make[1]: Entering directory '/usr/src/linux-headers-5.4.0-54-generic'
  CC [M]  /home/seed/Desktop/Labs_20.04/Network/Firewall/Labsetup/Files/packet_f
ilter/seedFilter.o
  Building modules, stage 2.
  MODPOST 1 modules
  CC [M]  /home/seed/Desktop/Labs_20.04/Network/Firewall/Labsetup/Files/packet_f
ilter/seedFilter.mod.o
  LD [M]  /home/seed/Desktop/Labs_20.04/Network/Firewall/Labsetup/Files/packet_f
ilter/seedFilter.ko
make[1]: Leaving directory '/usr/src/linux-headers-5.4.0-54-generic'
[07/28/21]seed@VM:~/.../packet_filter$
```

③使用 sudo insmod seedFilter.ko 加载内核：

```
[07/28/21]seed@VM:~/.../packet_filter$ sudo insmod seedFilter.ko
```

④登录到 host A，ping 10.9.0.1：

```
root@322d0f45bcc1:/# 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 ---
23 packets transmitted, 0 received, 100% packet loss, time 22538ms
```

⑤在 host A 上尝试 telnet 10.9.0.1：

```
root@322d0f45bcc1:/# telnet 10.9.0.1
Trying 10.9.0.1...
^C
root@322d0f45bcc1:/#
```

⑥在本机查看内核缓存：

```
[  388.819204] *** Dropping 10.9.0.1 (ICMP)
[  389.842582] *** Dropping 10.9.0.1 (ICMP)
[  390.865724] *** Dropping 10.9.0.1 (ICMP)
[  391.888967] *** Dropping 10.9.0.1 (ICMP)
[  392.915382] *** Dropping 10.9.0.1 (ICMP)
[  393.938733] *** Dropping 10.9.0.1 (ICMP)
[  394.963865] *** Dropping 10.9.0.1 (ICMP)
[  395.985284] *** Dropping 10.9.0.1 (ICMP)
[  397.009321] *** Dropping 10.9.0.1 (ICMP)
[  398.035376] *** Dropping 10.9.0.1 (ICMP)
[  399.059702] *** Dropping 10.9.0.1 (ICMP)
[  400.082763] *** Dropping 10.9.0.1 (ICMP)
[  401.106931] *** Dropping 10.9.0.1 (ICMP)
[  402.129169] *** Dropping 10.9.0.1 (ICMP)
[  403.153094] *** Dropping 10.9.0.1 (ICMP)
[  404.178075] *** Dropping 10.9.0.1 (ICMP)
[  405.202360] *** Dropping 10.9.0.1 (ICMP)
[  406.227900] *** Dropping 10.9.0.1 (ICMP)
[  407.252062] *** Dropping 10.9.0.1 (ICMP)
[  415.782518] *** Dropping 10.9.0.1 (TCP), port 23
[  416.788187] *** Dropping 10.9.0.1 (TCP), port 23
[  418.801977] *** Dropping 10.9.0.1 (TCP), port 23
[07/28/21]seed@VM:~/.../packet_filter$
```

⑦运行 sudo rmmod seedFilter 从内核中移除模块：

```
[07/28/21]seed@VM:~/.../packet_filter$ sudo rmmod seedFilter
[07/28/21]seed@VM:~/.../packet_filter$
```

Task 2.A

①在实验开始前尝试连接 10.9.0.1，发现可以连接：

```
root@322d0f45bcc1:/# 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=0.094 ms
64 bytes from 10.9.0.11: icmp_seq=2 ttl=64 time=0.044 ms
64 bytes from 10.9.0.11: icmp_seq=3 ttl=64 time=0.043 ms
^C
--- 10.9.0.11 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2043ms
rtt min/avg/max/mdev = 0.043/0.060/0.094/0.023 ms
```

```
root@322d0f45bcc1:/# telnet 10.9.0.11
Trying 10.9.0.11...
Connected to 10.9.0.11.
Escape character is '^]'.
Ubuntu 20.04.1 LTS
f76115f7c874 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.
```

②在路由器内输入命令：

```
[07/28/21]seed@VM:~$ docksh f7
root@f76115f7c874:/# iptables -A OUTPUT -p icmp --icmp-type echo-reply -j ACCEP
T
root@f76115f7c874:/# iptables -A INPUT -p icmp --icmp-type echo-request -j ACCE
PT
root@f76115f7c874:/# iptables -P OUTPUT DROP
root@f76115f7c874:/# iptables -P INPUT DROP
root@f76115f7c874:/#
```

③尝试在 host A 中 ping 10.9.0.1：

```
root@322d0f45bcc1:/# 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=0.098 ms
64 bytes from 10.9.0.11: icmp_seq=2 ttl=64 time=0.121 ms
64 bytes from 10.9.0.11: icmp_seq=3 ttl=64 time=0.044 ms
^C
--- 10.9.0.11 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2053ms
rtt min/avg/max/mdev = 0.044/0.087/0.121/0.032 ms
```

能够 ping 通。

④尝试 telnet 10.9.0.1：

```
root@322d0f45bcc1:/# telnet 10.9.0.11
Trying 10.9.0.11...
^C
root@322d0f45bcc1:/#
```

无法连接。

Task 2.B

①在路由器内设置如下规则：

```
root@6807541a3fa9:/# iptables -A FORWARD -p icmp --icmp-type echo-request -j A
CCEPT -i eth1
root@6807541a3fa9:/# iptables -A FORWARD -p icmp --icmp-type echo-reply -j ACC
EPT -i eth0
root@6807541a3fa9:/# iptables -P FORWARD DROP
root@6807541a3fa9:/#
```

②用外部主机 ping 内部主机：

```
root@64cd50783dfe:/# 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 ---
6 packets transmitted, 0 received, 100% packet loss, time 5116ms
```

无法 ping 通。

③用外部主机 ping 路由器：

```
root@64cd50783dfe:/# 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=0.104 ms
64 bytes from 10.9.0.11: icmp_seq=2 ttl=64 time=0.043 ms
64 bytes from 10.9.0.11: icmp_seq=3 ttl=64 time=0.057 ms
^C
--- 10.9.0.11 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2028ms
rtt min/avg/max/mdev = 0.043/0.068/0.104/0.026 ms
```

可以 ping 通。

④用外部主机 telnet 内部主机：

```
root@64cd50783dfe:/# telnet 192.168.60.5
Trying 192.168.60.5...
^C
root@64cd50783dfe:/#
```

无法联通。

⑤用内部主机 ping 外部主机：

```
root@41f654f0272d:/# 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=0.152 ms
64 bytes from 10.9.0.5: icmp_seq=2 ttl=63 time=0.055 ms
64 bytes from 10.9.0.5: icmp_seq=3 ttl=63 time=0.053 ms
^C
--- 10.9.0.5 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2035ms
rtt min/avg/max/mdev = 0.053/0.086/0.152/0.046 ms
root@41f654f0272d:/#
```

可以 ping 通。

⑥用内部主机 telnet 外部主机：

```
root@41f654f0272d:/# telnet 10.9.0.5
Trying 10.9.0.5...
^C
root@41f654f0272d:/#
```

不能连通。

Task 2.C

①在路由器内设置规则：

```
[07/29/21]seed@VM:~$ docksh 9e
root@9e5d2a513f5a:/# iptables -A FORWARD -i eth0 -p tcp --dport 23 -d 192.168.
60.5 -j ACCEPT
root@9e5d2a513f5a:/# iptables -A FORWARD -o eth0 -p tcp --sport 23 -s 192.168.
60.5 -j ACCEPT
root@9e5d2a513f5a:/# iptables -P FORWARD DROP
root@9e5d2a513f5a:/#
```

②在外部主机尝试 telnet 内部主机：

```
root@6b1b088047f0:/# telnet 192.168.60.5
Trying 192.168.60.5...
Connected to 192.168.60.5.
Escape character is '^]'.
Ubuntu 20.04.1 LTS
d11890f06ae1 login: seed
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-54-generic x86_64)
```

连接成功。

③在外部主机尝试 telnet host2：

```
root@6b1b088047f0:/# telnet 192.168.60.6
Trying 192.168.60.6...
^C
root@6b1b088047f0:/#
```

无法连接。

④在内部主机尝试 telnet 外部主机：

```
root@d11890f06ae1:/# telnet 10.9.0.5
Trying 10.9.0.5...
^C
```

无法连接。

⑤在内部主机尝试 telnet 另一内部主机：

```
root@d11890f06ae1:/# telnet 192.168.60.6
Trying 192.168.60.6...
Connected to 192.168.60.6.
Escape character is '^]'.
Ubuntu 20.04.1 LTS
64037fee1f11 login: seed
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-54-generic x86_64)
```

连接成功。

Task 3.A

①在外部主机上 ping 内部主机：

```
root@3ba11a11335b:/# 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=0.083 ms
64 bytes from 192.168.60.5: icmp_seq=2 ttl=63 time=0.055 ms
64 bytes from 192.168.60.5: icmp_seq=3 ttl=63 time=0.053 ms
^C
--- 192.168.60.5 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2028ms
rtt min/avg/max/mdev = 0.053/0.063/0.083/0.013 ms
```

②ICMP 的连接持续时间约为 30s：

```
root@6bb676b4bfa0:/# conntrack -L
icmp     1 25 src=10.9.0.5 dst=192.168.60.5 type=8 code=0 id=31 src=192.168.60
.5 dst=10.9.0.5 type=0 code=0 id=31 mark=0 use=1
conntrack v1.4.5 (conntrack-tools): 1 flow entries have been shown.
root@6bb676b4bfa0:/#
```

③使用 nc -lu 命令连接外部和内部主机：

```
root@3ba11a11335b:/# nc -u 192.168.60.5 9090
lkdj
ldkfjlkje
^C
root@3ba11a11335b:/#
```

```
root@b52b8b2104c7:/# nc -lu 9090
lkdj
ldkfjlkje
^C
root@b52b8b2104c7:/#
```

④UDP 的连接持续时间约为 30s:

```
root@6bb676b4bfa0:/# conntrack -L
udp      17 19 src=10.9.0.5 dst=192.168.60.5 sport=45350 dport=9090 [UNREPLIED
] src=192.168.60.5 dst=10.9.0.5 sport=9090 dport=45350 mark=0 use=1
conntrack v1.4.5 (conntrack-tools): 1 flow entries have been shown.
root@6bb676b4bfa0:/#
```

⑤使用 nc -l 命令连接外部和内部主机：

```
root@3ba11a11335b:/# nc 192.168.60.5 9090
lkjejal
lkejaoijkvjlan;
^C
root@3ba11a11335b:/#
```

```
root@b52b8b2104c7:/# nc -l 9090
lkjejal
lkejaoijkvjlan;
^C
root@b52b8b2104c7:/#
```

⑥TCP 的连接持续时间约 120s：

```
root@6bb676b4bfa0:/# conntrack -L
tcp      6 114 TIME_WAIT src=10.9.0.5 dst=192.168.60.5 sport=42530 dport=9090
src=192.168.60.5 dst=10.9.0.5 sport=9090 dport=42530 [ASSURED] mark=0 use=1
conntrack v1.4.5 (conntrack-tools): 1 flow entries have been shown.
root@6bb676b4bfa0:/#
```

Task 3.B

①在路由器内设置规则：

```
root@adeecb2424a5:/# iptables -F
root@adeecb2424a5:/# iptables -A FORWARD -p tcp -m conntrack --ctstate ESTABLI
SHED,RELATED -j ACCEPT
root@adeecb2424a5:/# iptables -A FORWARD -p tcp --dport 23 -d 192.168.60.5 --s
yn -m conntrack --ctstate NEW -j ACCEPT
root@adeecb2424a5:/# iptables -A FORWARD -p tcp --dport 23 -d 10.9.0.0/24 --sy
n -m conntrack --ctstate NEW -j ACCEPT
root@adeecb2424a5:/# iptables -P FORWARD DROP
root@adeecb2424a5:/#
```

②在外部主机上 telnet 内部主机：

```
root@85771dc13f31:/# telnet 192.168.60.5
Trying 192.168.60.5...
Connected to 192.168.60.5.
Escape character is '^]'.
Ubuntu 20.04.1 LTS
6865a3e90c04 login: seed
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-54-generic x86_64)
```

连接成功。

③在外部主机上 telnet 另一内部主机：

```
root@85771dc13f31:/# telnet 192.168.60.6
Trying 192.168.60.6...
^C
```

无法连接。

④在内部主机上 telnet 另一主机：

```
root@6865a3e90c04:/# telnet 192.168.60.6
Trying 192.168.60.6...
Connected to 192.168.60.6.
Escape character is '^]'.
Ubuntu 20.04.1 LTS
09d103e2c066 login: seed
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-54-generic x86_64)
```

连接成功。

⑤在内部主机上 telnet 外部主机：

```
root@6865a3e90c04:/# telnet 10.9.0.5
Trying 10.9.0.5...
Connected to 10.9.0.5.
Escape character is '^]'.
Ubuntu 20.04.1 LTS
85771dc13f31 login: seed
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-54-generic x86_64)
```

连接成功。

Task 4

①在路由器上设置规则：

```
root@515a5da0102c:/# iptables -A FORWARD -s 10.9.0.5 -m limit --limit 10/minut
 --limit-burst 5 -j ACCEPT
root@515a5da0102c:/# iptables -A FORWARD -s 10.9.0.5 -j DROP
root@515a5da0102c:/#
```

②在外部主机上 ping 内部主机：

```
root@990c415c127c:/# 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=0.150 ms
64 bytes from 192.168.60.5: icmp_seq=2 ttl=63 time=0.055 ms
64 bytes from 192.168.60.5: icmp_seq=3 ttl=63 time=0.057 ms
64 bytes from 192.168.60.5: icmp_seq=4 ttl=63 time=0.054 ms
64 bytes from 192.168.60.5: icmp_seq=5 ttl=63 time=0.054 ms
64 bytes from 192.168.60.5: icmp_seq=7 ttl=63 time=0.073 ms
64 bytes from 192.168.60.5: icmp_seq=13 ttl=63 time=0.055 ms
64 bytes from 192.168.60.5: icmp_seq=19 ttl=63 time=0.053 ms
^C
--- 192.168.60.5 ping statistics ---
23 packets transmitted, 8 received, 65.2174% packet loss, time 22509ms
rtt min/avg/max/mdev = 0.053/0.068/0.150/0.031 ms
root@990c415c127c:/#
```

可见前六个包发送速度很快，之后每隔 6s 发送一个。

Task 5

①发送命令：

```
root@515a5da0102c:/# iptables -t nat -A PREROUTING -p udp --dport 8080 -m stati
stic --mode nth --every 3 --packet 0 -j DNAT --to-destination 192.168.60.5:8080
root@515a5da0102c:/# iptables -t nat -A PREROUTING -p udp --dport 8080 -m stati
stic --mode nth --every 3 --packet 1 -j DNAT --to-destination 192.168.60.6:8080
root@515a5da0102c:/# iptables -t nat -A PREROUTING -p udp --dport 8080 -m stati
stic --mode nth --every 3 --packet 2 -j DNAT --to-destination 192.168.60.7:8080
```

②查看：

```
root@6e9418ee15a8:/# nc -luk 8080
hello_1
```

```
[07/29/21]seed@VM:~$ docksh 8d
root@8dcf9d4cf702:/# nc -luk 8080
hello_2
```

```
root@0ae6eca57f73:/# nc -luk 8080
hello_3
```

③发送命令：

④查看：

```
root@0ae6eca57f73:/# nc -luk 8080
hello_1
hello_2
hello_5
hello_6
hello_9
hello_10
hello_11
hello_14
```

```
root@8dcf9d4cf702:/# nc -luk 8080
hello_7
hello_12
hello_13
hello_15
```

```
root@6e9418ee15a8:/# nc -luk 8080
hello_3
hello_4
hello_8
```

可见有50%的概率发向192.168.60.7，有25%的概率发向192.168.60.6，有25%的概率发向192.168.60.5。