Lab6-Report

57117136 孙伟杰

Task 1

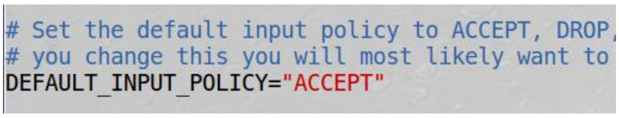
A 的 IP 地址为 10.0.2.5

B 的 IP 地址为 10.0.2.6

实验流程：

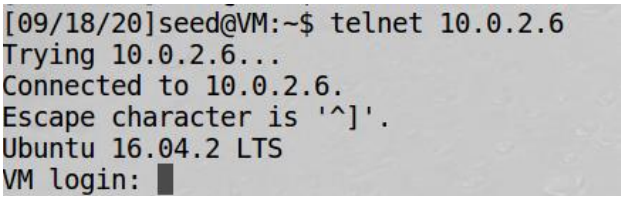
Prevent A from doing telnet to Machine B

1. 首先修改/etc/default/ufw 下的默认设置



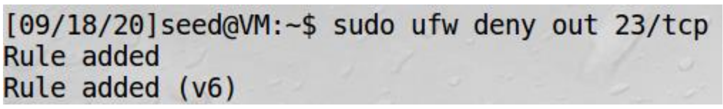
```
# Set the default input policy to ACCEPT, DROP,
# you change this you will most likely want to
DEFAULT_INPUT_POLICY="ACCEPT"
```

2.未进行任何设置前，A 可以正常 Telnet B



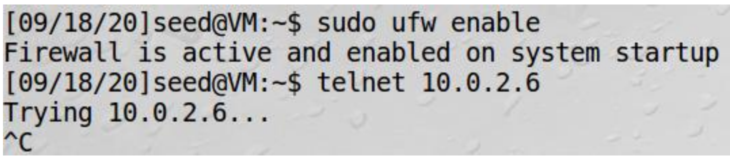
```
[09/18/20]seed@VM:~$ telnet 10.0.2.6
Trying 10.0.2.6...
Connected to 10.0.2.6.
Escape character is '^]'.
Ubuntu 16.04.2 LTS
VM login: █
```

3.设置相应 ufw 规则，禁止 tcp 23 端口的流量流出，设置完成后启动防火墙



```
[09/18/20]seed@VM:~$ sudo ufw deny out 23/tcp
Rule added
Rule added (v6)
```

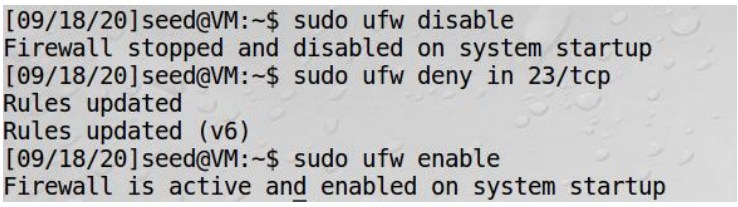
4.此时无法 A 正常使用 Telnet B 的服务



```
[09/18/20]seed@VM:~$ sudo ufw enable
Firewall is active and enabled on system startup
[09/18/20]seed@VM:~$ telnet 10.0.2.6
Trying 10.0.2.6...
^C
```

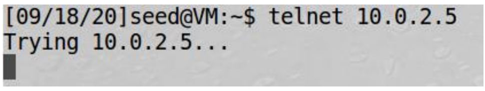
Prevent B from doing telnet to Machine A

1. 设置相应 ufw 规则，禁止 tcp 23 端口的流量流入，设置完成后启动防火墙



```
[09/18/20]seed@VM:~$ sudo ufw disable
Firewall stopped and disabled on system startup
[09/18/20]seed@VM:~$ sudo ufw deny in 23/tcp
Rules updated
Rules updated (v6)
[09/18/20]seed@VM:~$ sudo ufw enable
Firewall is active and enabled on system startup
```

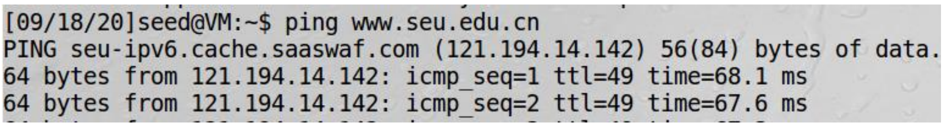
2.B 在开启防火墙后无法正常 Telnet 连接 A



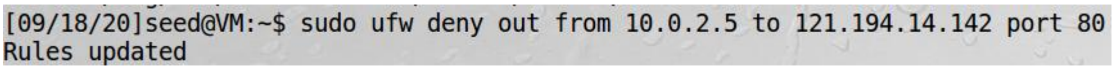
```
[09/18/20]seed@VM:~$ telnet 10.0.2.5
Trying 10.0.2.5...
█
```

Prevent A from visiting an external web site

1.尝试 ping 通 www.seu.edu.cn，获取外部网站 ip 地址

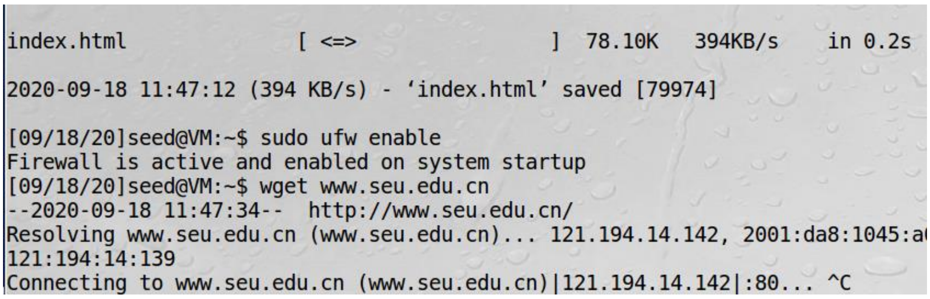


```
[09/18/20]seed@VM:~$ ping www.seu.edu.cn
PING seu-ipv6.cache.saaswaf.com (121.194.14.142) 56(84) bytes of data.
64 bytes from 121.194.14.142: icmp_seq=1 ttl=49 time=68.1 ms
64 bytes from 121.194.14.142: icmp_seq=2 ttl=49 time=67.6 ms
```



```
[09/18/20]seed@VM:~$ sudo ufw deny out from 10.0.2.5 to 121.194.14.142 port 80
Rules updated
```

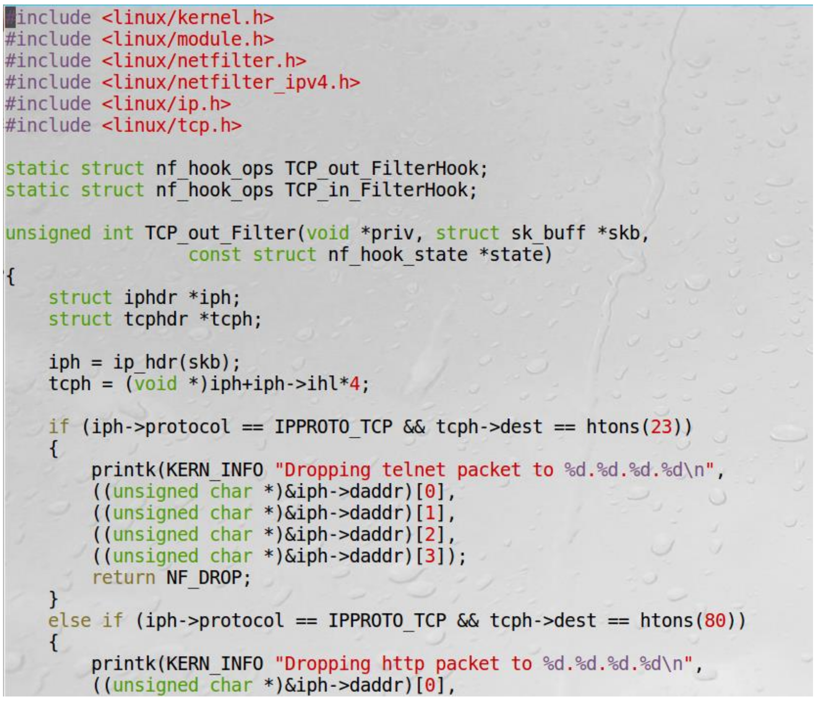
2.设置 10.0.2.5 到 121.194.14.142 从 80 端口流出的流量，无法实现连接



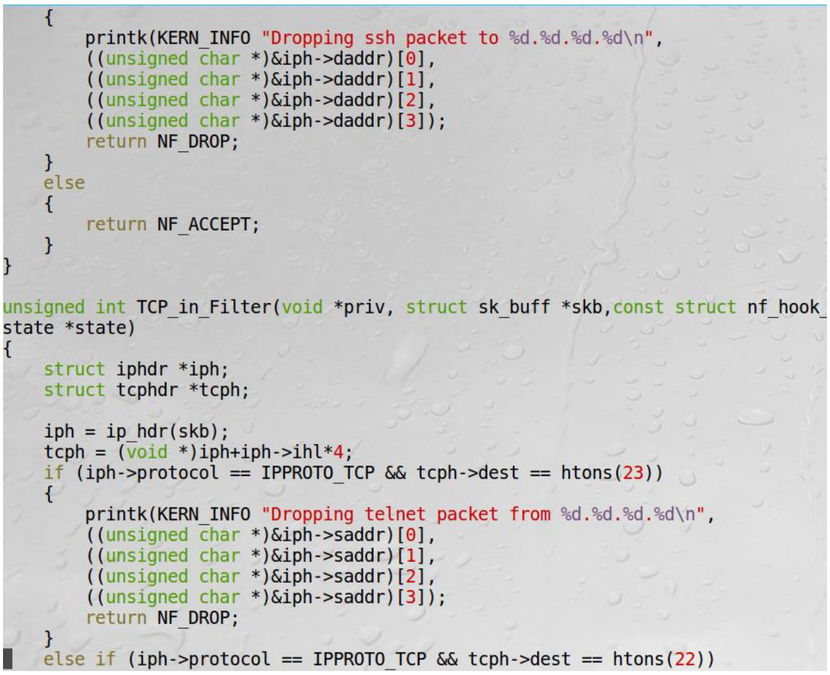
```
index.html              [ <=>                ]  78.10K   394KB/s    in 0.2s

2020-09-18 11:47:12 (394 KB/s) - ‘index.html’ saved [79974]

[09/18/20]seed@VM:~$ sudo ufw enable
Firewall is active and enabled on system startup
[09/18/20]seed@VM:~$ wget www.seu.edu.cn
--2020-09-18 11:47:34--  http://www.seu.edu.cn/
Resolving www.seu.edu.cn (www.seu.edu.cn)... 121.194.14.142, 2001:da8:1045:a
121:194:14:139
Connecting to www.seu.edu.cn (www.seu.edu.cn)|121.194.14.142|:80... ^C
```

Task 2

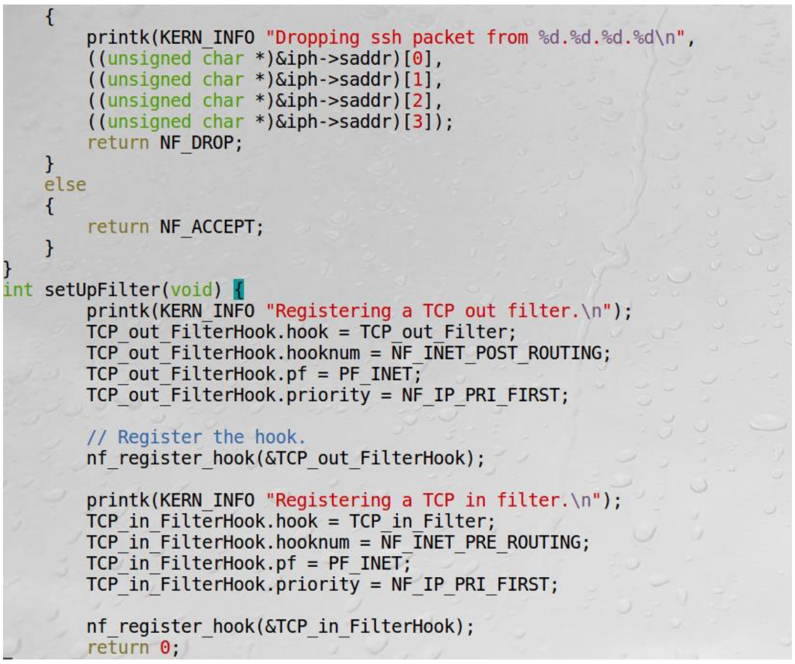
首先编写过滤器程序 Filter.c



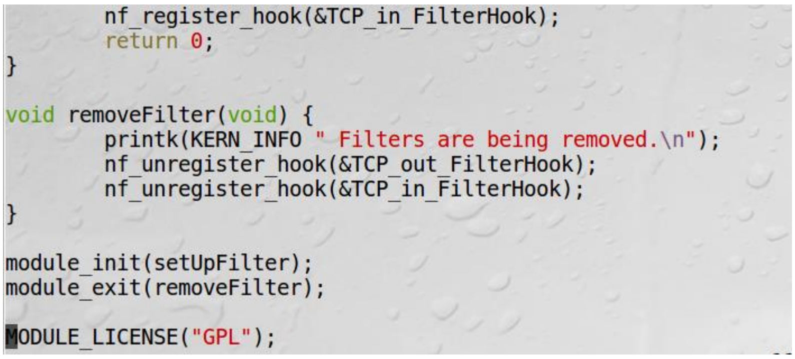
```c
#include <linux/kernel.h>
#include <linux/module.h>
#include <linux/netfilter.h>
#include <linux/netfilter_ipv4.h>
#include <linux/ip.h>
#include <linux/tcp.h>

static struct nf_hook_ops TCP_out_FilterHook;
static struct nf_hook_ops TCP_in_FilterHook;

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

    iph = ip_hdr(skb);
    tcph = (void *)iph+iph->ihl*4;

    if (iph->protocol == IPPROTO_TCP && tcph->dest == htons(23))
    {
        printk(KERN_INFO "Dropping telnet packet to %d.%d.%d.%d\n",
        ((unsigned char *)&iph->daddr)[0],
        ((unsigned char *)&iph->daddr)[1],
        ((unsigned char *)&iph->daddr)[2],
        ((unsigned char *)&iph->daddr)[3]);
        return NF_DROP;
    }
    else if (iph->protocol == IPPROTO_TCP && tcph->dest == htons(80))
    {
        printk(KERN_INFO "Dropping http packet to %d.%d.%d.%d\n",
        ((unsigned char *)&iph->daddr)[0],
```



```c
    {
        printk(KERN_INFO "Dropping ssh packet to %d.%d.%d.%d\n",
        ((unsigned char *)&iph->daddr)[0],
        ((unsigned char *)&iph->daddr)[1],
        ((unsigned char *)&iph->daddr)[2],
        ((unsigned char *)&iph->daddr)[3]);
        return NF_DROP;
    }
    else
    {
        return NF_ACCEPT;
    }
}

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

    iph = ip_hdr(skb);
    tcph = (void *)iph+iph->ihl*4;
    if (iph->protocol == IPPROTO_TCP && tcph->dest == htons(23))
    {
        printk(KERN_INFO "Dropping telnet packet from %d.%d.%d.%d\n",
        ((unsigned char *)&iph->saddr)[0],
        ((unsigned char *)&iph->saddr)[1],
        ((unsigned char *)&iph->saddr)[2],
        ((unsigned char *)&iph->saddr)[3]);
        return NF_DROP;
    }
    else if (iph->protocol == IPPROTO_TCP && tcph->dest == htons(22))
```

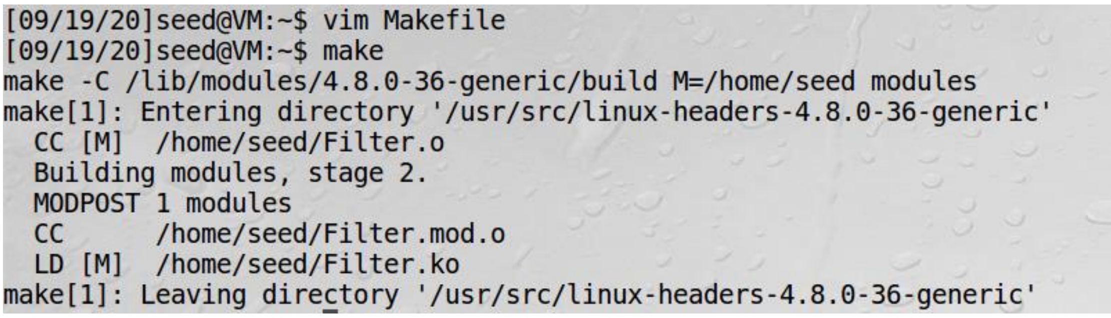


```
    {
        printk(KERN_INFO "Dropping ssh packet from %d.%d.%d.%d\n",
        ((unsigned char *)&iph->saddr)[0],
        ((unsigned char *)&iph->saddr)[1],
        ((unsigned char *)&iph->saddr)[2],
        ((unsigned char *)&iph->saddr)[3]);
        return NF_DROP;
    }
    else
    {
        return NF_ACCEPT;
    }
}
int setUpFilter(void) {
        printk(KERN_INFO "Registering a TCP out filter.\n");
        TCP_out_FilterHook.hook = TCP_out_Filter;
        TCP_out_FilterHook.hooknum = NF_INET_POST_ROUTING;
        TCP_out_FilterHook.pf = PF_INET;
        TCP_out_FilterHook.priority = NF_IP_PRI_FIRST;

        // Register the hook.
        nf_register_hook(&TCP_out_FilterHook);

        printk(KERN_INFO "Registering a TCP in filter.\n");
        TCP_in_FilterHook.hook = TCP_in_Filter;
        TCP_in_FilterHook.hooknum = NF_INET_PRE_ROUTING;
        TCP_in_FilterHook.pf = PF_INET;
        TCP_in_FilterHook.priority = NF_IP_PRI_FIRST;

        nf_register_hook(&TCP_in_FilterHook);
        return 0;
```



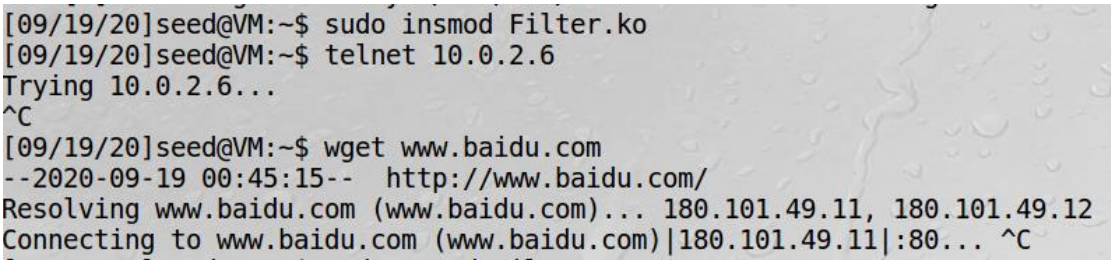
```
        nf_register_hook(&TCP_in_FilterHook);
        return 0;
}

void removeFilter(void) {
        printk(KERN_INFO " Filters are being removed.\n");
        nf_unregister_hook(&TCP_out_FilterHook);
        nf_unregister_hook(&TCP_in_FilterHook);
}

module_init(setUpFilter);
module_exit(removeFilter);

MODULE_LICENSE("GPL");
```

1.编译成功后载入相关模块



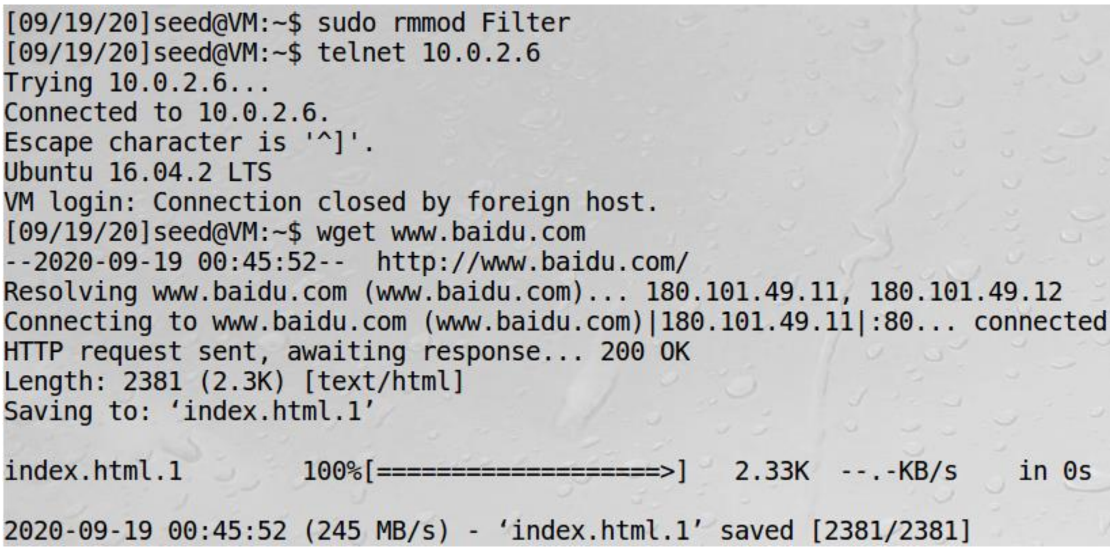
```
[09/19/20]seed@VM:~$ vim Makefile
[09/19/20]seed@VM:~$ make
make -C /lib/modules/4.8.0-36-generic/build M=/home/seed modules
make[1]: Entering directory '/usr/src/linux-headers-4.8.0-36-generic'
  CC [M]  /home/seed/Filter.o
  Building modules, stage 2.
  MODPOST 1 modules
  CC      /home/seed/Filter.mod.o
  LD [M]  /home/seed/Filter.ko
make[1]: Leaving directory '/usr/src/linux-headers-4.8.0-36-generic'
```

2.模块载入后，无法实现 A telnet B，B telnet A 以及 wget 访问 www.baidu.com



```
[09/19/20]seed@VM:~$ sudo insmod Filter.ko
[09/19/20]seed@VM:~$ telnet 10.0.2.6
Trying 10.0.2.6...
^C
[09/19/20]seed@VM:~$ wget www.baidu.com
--2020-09-19 00:45:15--  http://www.baidu.com/
Resolving www.baidu.com (www.baidu.com)... 180.101.49.11, 180.101.49.12
Connecting to www.baidu.com (www.baidu.com)|180.101.49.11|:80... ^C
```

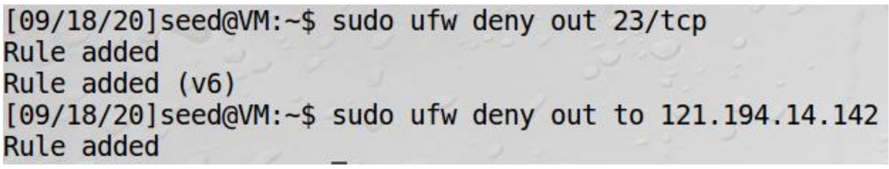
3.移除载入模块后，服务恢复正常



```
[09/19/20]seed@VM:~$ sudo rmmod Filter
[09/19/20]seed@VM:~$ telnet 10.0.2.6
Trying 10.0.2.6...
Connected to 10.0.2.6.
Escape character is '^]'.
Ubuntu 16.04.2 LTS
VM login: Connection closed by foreign host.
[09/19/20]seed@VM:~$ wget www.baidu.com
--2020-09-19 00:45:52--  http://www.baidu.com/
Resolving www.baidu.com (www.baidu.com)... 180.101.49.11, 180.101.49.12
Connecting to www.baidu.com (www.baidu.com)|180.101.49.11|:80... connected
HTTP request sent, awaiting response... 200 OK
Length: 2381 (2.3K) [text/html]
Saving to: ‘index.html.1’

index.html.1        100%[===================>]   2.33K  --.-KB/s    in 0s

2020-09-19 00:45:52 (245 MB/s) - ‘index.html.1’ saved [2381/2381]
```

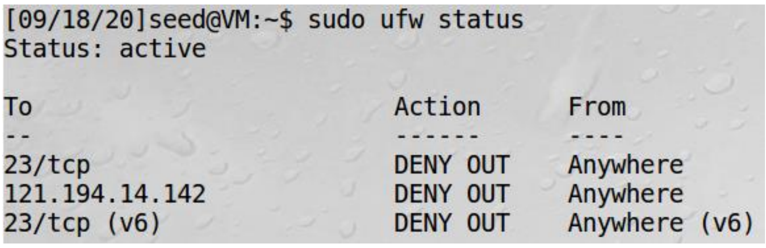
Task 3

1.移除task2中可加载内核模块，删除ufw中之前设置的所有规则，重新添加：

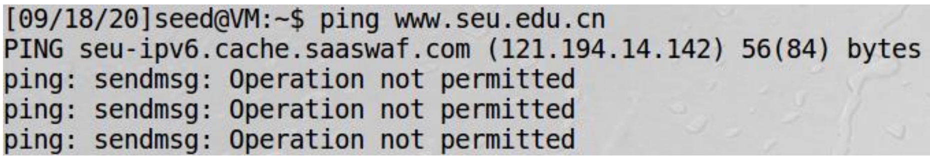


```
[09/18/20]seed@VM:~$ sudo ufw deny out 23/tcp
Rule added
Rule added (v6)
[09/18/20]seed@VM:~$ sudo ufw deny out to 121.194.14.142
Rule added
```

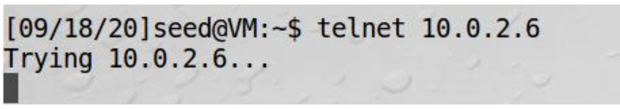
2. 此时A(10.0.2.5)已经无法向外telnet以及访问www.seu.edu.cn



```
[09/18/20]seed@VM:~$ sudo ufw status
Status: active

To                         Action      From
--                         ------      ----
23/tcp                     DENY OUT    Anywhere
121.194.14.142             DENY OUT    Anywhere
23/tcp (v6)                DENY OUT    Anywhere (v6)
```



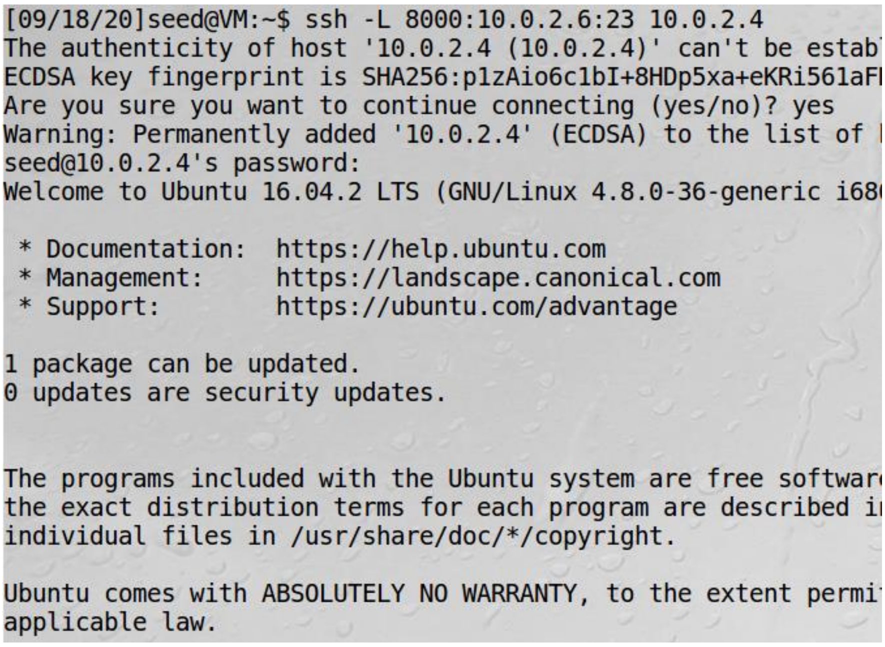
```
[09/18/20]seed@VM:~$ ping www.seu.edu.cn
PING seu-ipv6.cache.saaswaf.com (121.194.14.142) 56(84) bytes
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
```



```
[09/18/20]seed@VM:~$ telnet 10.0.2.6
Trying 10.0.2.6...
```

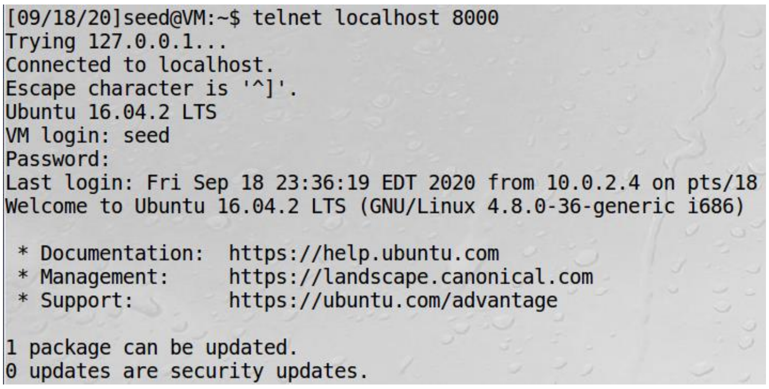
Task 3.a

1.利用C（10.0.2.4）开辟一条以B（10.0.2.6）为telnet通讯目标的ssh隧道



```
[09/18/20]seed@VM:~$ ssh -L 8000:10.0.2.6:23 10.0.2.4
The authenticity of host '10.0.2.4 (10.0.2.4)' can't be estab
ECDSA key fingerprint is SHA256:p1zAio6c1bI+8HDp5xa+eKRi561aF
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.0.2.4' (ECDSA) to the list of 
seed@10.0.2.4's password:
Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.8.0-36-generic i68

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

1 package can be updated.
0 updates are security updates.


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

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

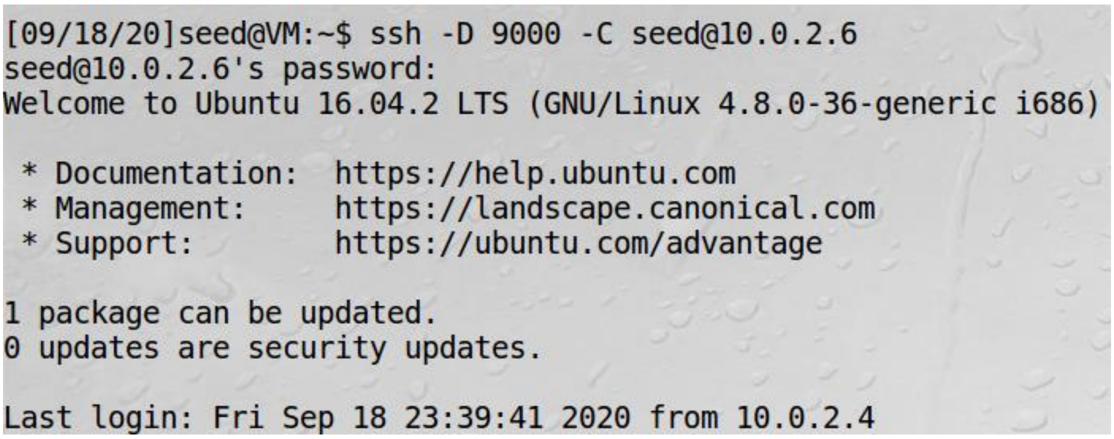
2.隧道开辟完成后，在 A（10.0.2.5）上执行 telnet localhost 8000



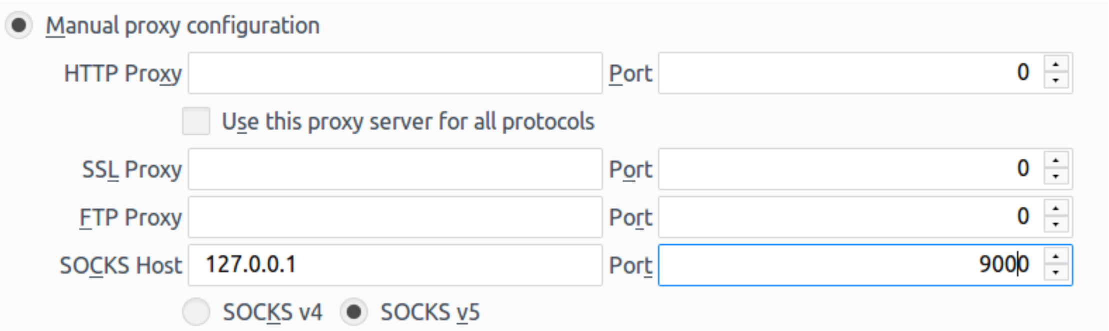
```
[09/18/20]seed@VM:~$ telnet localhost 8000
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
Ubuntu 16.04.2 LTS
VM login: seed
Password:
Last login: Fri Sep 18 23:36:19 EDT 2020 from 10.0.2.4 on pts/18
Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.8.0-36-generic i686)

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

1 package can be updated.
0 updates are security updates.
```

Task 3.b

1.设置 ssh 隧道，同时在 Firefox 浏览器中设置代理：



```
[09/18/20]seed@VM:~$ ssh -D 9000 -C seed@10.0.2.6
seed@10.0.2.6's password:
Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.8.0-36-generic i686)

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

1 package can be updated.
0 updates are security updates.

Last login: Fri Sep 18 23:39:41 2020 from 10.0.2.4
```



● Manual proxy configuration

HTTP Proxy [ ] Port 0

□ Use this proxy server for all protocols

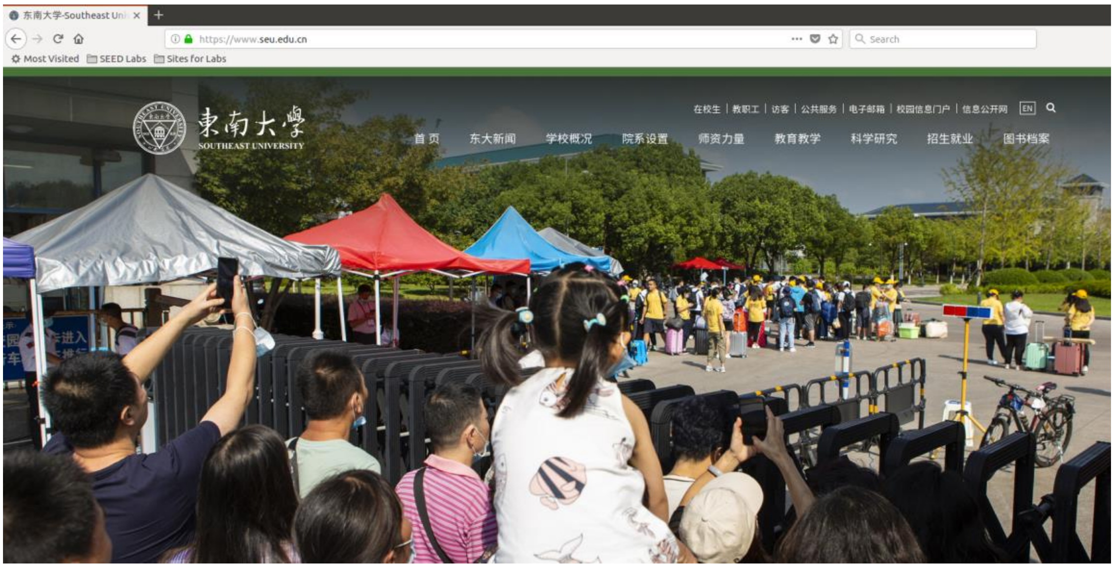
SSL Proxy [ ] Port 0

FTP Proxy [ ] Port 0

SOCKS Host 127.0.0.1 Port 9000

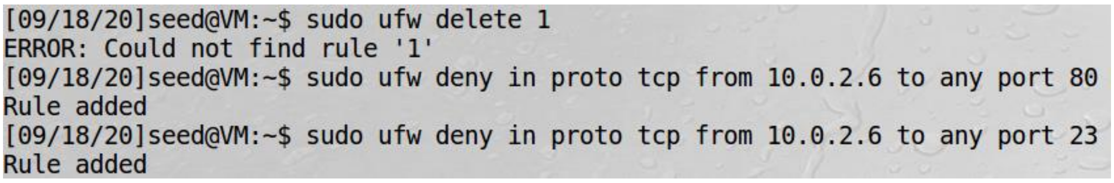
○ SOCKS v4 ● SOCKS v5

2.设置成功后，可以成功访问本应该被防火墙阻隔的 www.seu.edu.cn

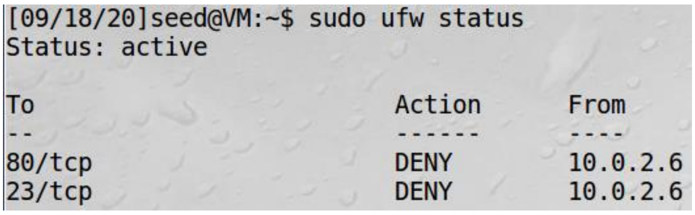


Task 4

1. 对主机 A 进行配置（10.0.2.5），禁止 B（10.0.2.6）访问 23 和 80 端口

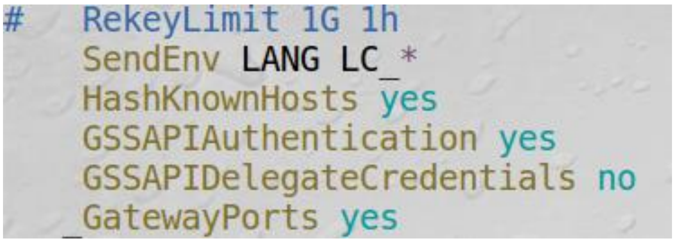


```
[09/18/20]seed@VM:~$ sudo ufw delete 1
ERROR: Could not find rule '1'
[09/18/20]seed@VM:~$ sudo ufw deny in proto tcp from 10.0.2.6 to any port 80
Rule added
[09/18/20]seed@VM:~$ sudo ufw deny in proto tcp from 10.0.2.6 to any port 23
Rule added
```



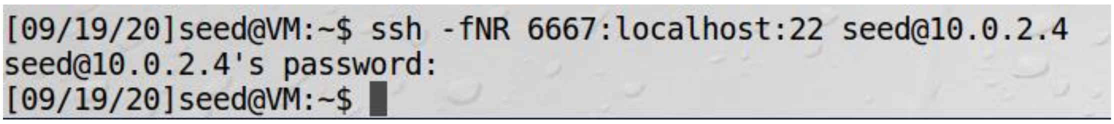
```
[09/18/20]seed@VM:~$ sudo ufw status
Status: active

To                         Action      From
--                         ------      ----
80/tcp                     DENY        10.0.2.6
23/tcp                     DENY        10.0.2.6
```

2. 对主机 C（10.0.2.4）中/etc/ssh/sshd_config 进行配置



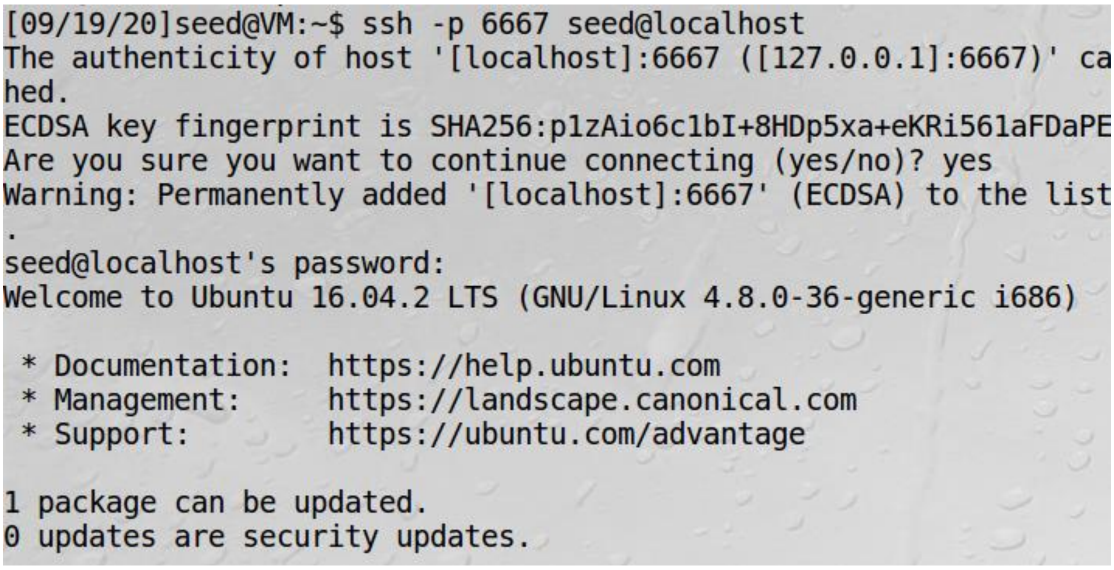
```
#   RekeyLimit 1G 1h
    SendEnv LANG LC_*
    HashKnownHosts yes
    GSSAPIAuthentication yes
    GSSAPIDelegateCredentials no
    GatewayPorts yes
```

3.在 A 中设置逆向的 ssh 隧道，使得主机 B 可以通过访问 C 的 6667 号端口访问 A



```
[09/19/20]seed@VM:~$ ssh -fNR 6667:localhost:22 seed@10.0.2.4
seed@10.0.2.4's password:
[09/19/20]seed@VM:~$ █
```

4.主机 B 可以成功访问 A



```
[09/19/20]seed@VM:~$ ssh -p 6667 seed@localhost
The authenticity of host '[localhost]:6667 ([127.0.0.1]:6667)' ca
hed.
ECDSA key fingerprint is SHA256:p1zAio6c1bI+8HDp5xa+eKRi561aFDaPE
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '[localhost]:6667' (ECDSA) to the list
.
seed@localhost's password:
Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.8.0-36-generic i686)

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

1 package can be updated.
0 updates are security updates.
```