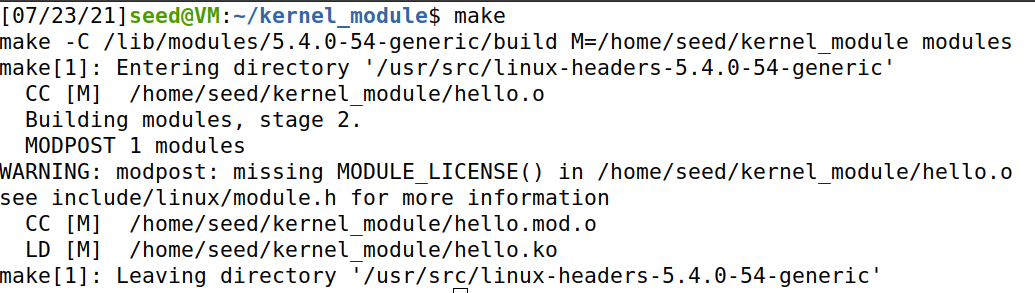
Firewall Exploration Lab

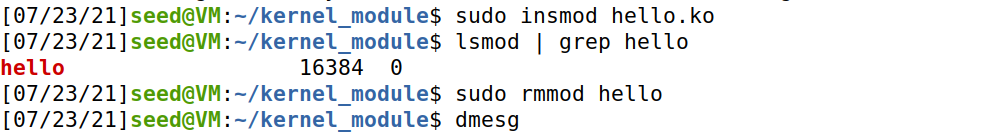
学号：57118223 姓名：齐天一

**Task1.A**

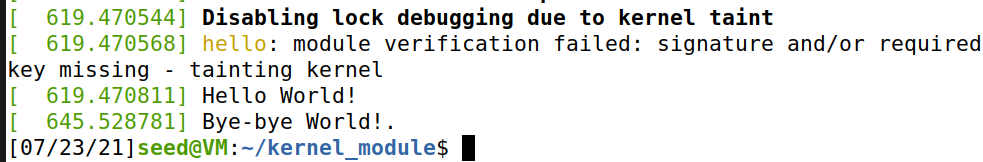
（1）将kernel\_module移动到没有空格的文件夹，并编译



（2）输入指令添加模块

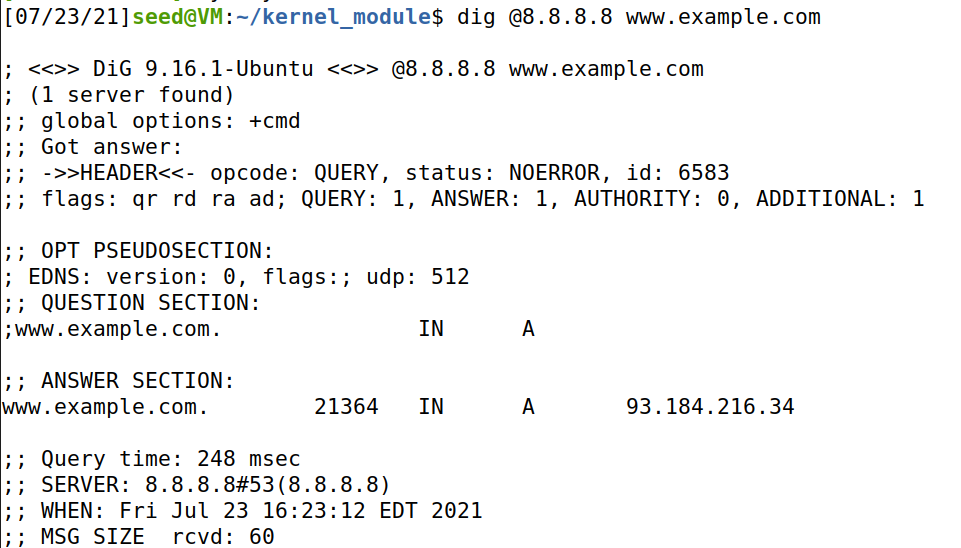


（3）提示语显示安装成功

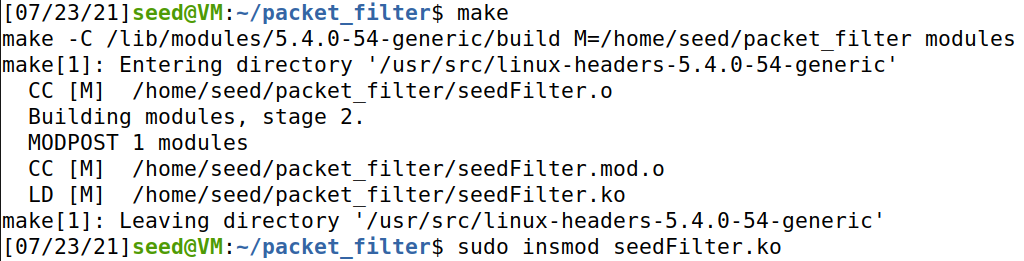


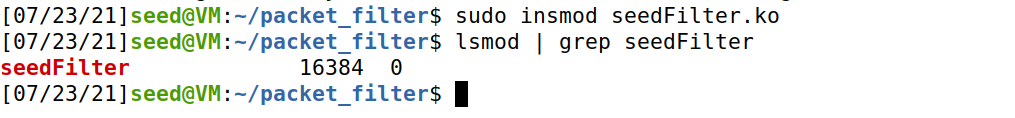
**Task1.B.1**

1. 攻击前使用dig指令进行查看

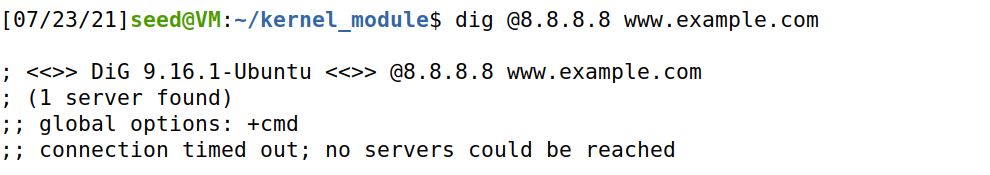


（2）编译packet\_filter内核

（3）输入指令加载模块



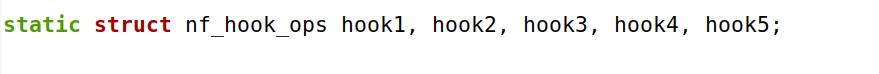
1. 重新输入dig指令

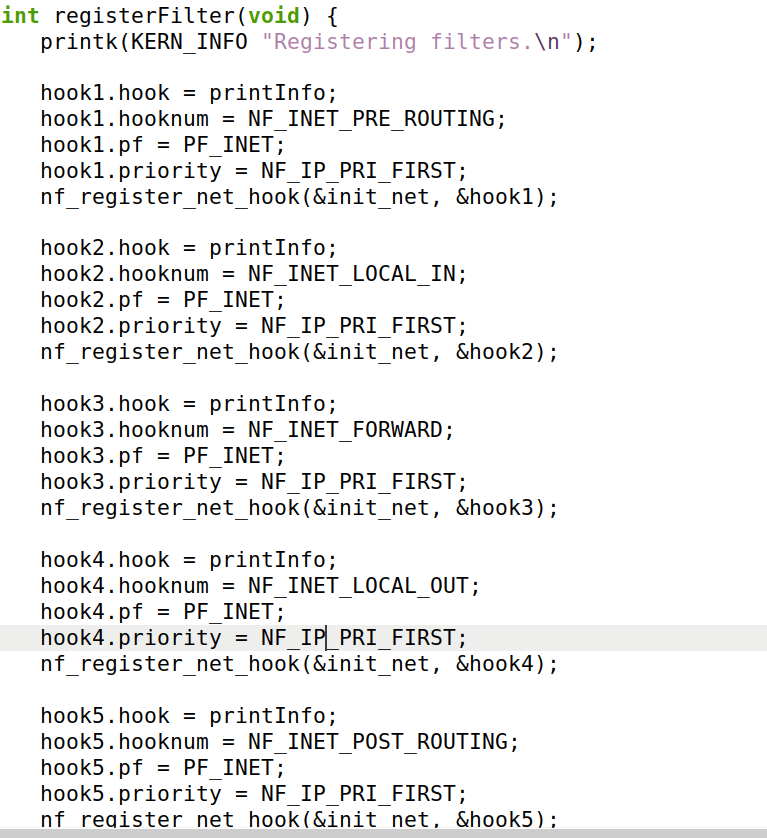


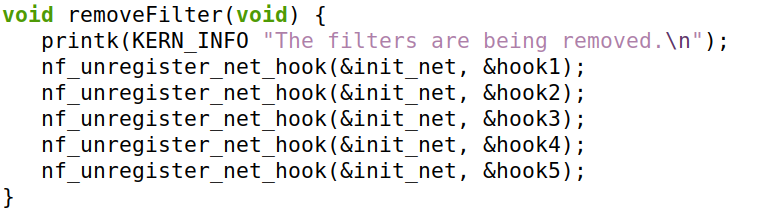
可以看到无法连接

**Task1.B.2**

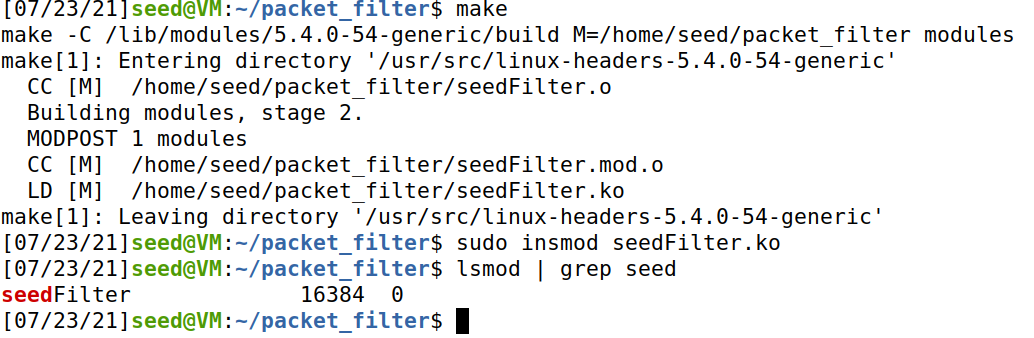
1. 修改程序，新增hook并将其与printInfo关联



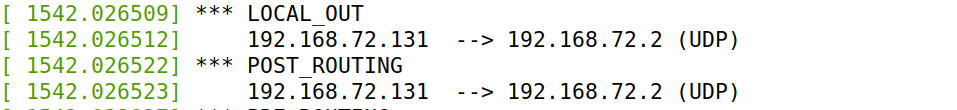




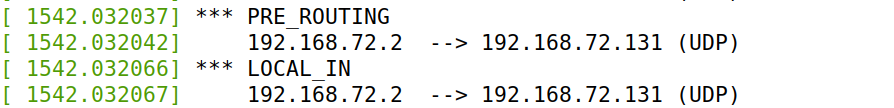
1. 重新编译和加载

（3）输入dmesg查看日志

可以看到，对于本地产生向外部网络发出的数据包，会依次经过NF\_INET\_LOCAL\_OUT和NF\_INET\_POST\_ROUTING两个hook点的处理。推测前者是本地产生包时触发，后者是向外发送时触发。



发往本地的数据包，会依次经过NF\_INET\_PRE\_ROUTING和NF\_INET\_LOCAL\_IN两个hook点的处理。推测前者是判断是发往外部还是本地接收时触发，后者是发往本地时触发。



在日志中没有出现NF\_INET\_FORWARD点的活动

因此得到结论：

NF\_INET\_LOCAL\_OUT 触发条件：由本地产生的包

NF\_INET\_POST\_ROUTING 触发条件：向外部网络发出的包

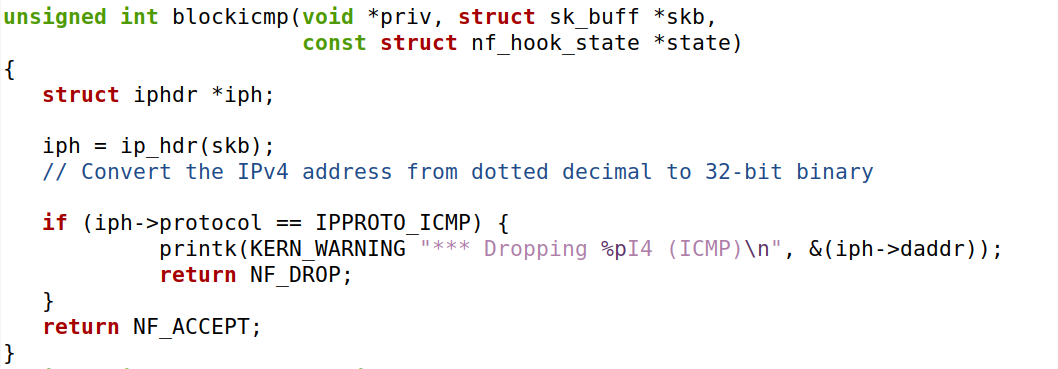
NF\_INET\_PRE\_ROUTING 触发条件：本地接收到的包（不包括本地产生的），用于判断是否向外转发

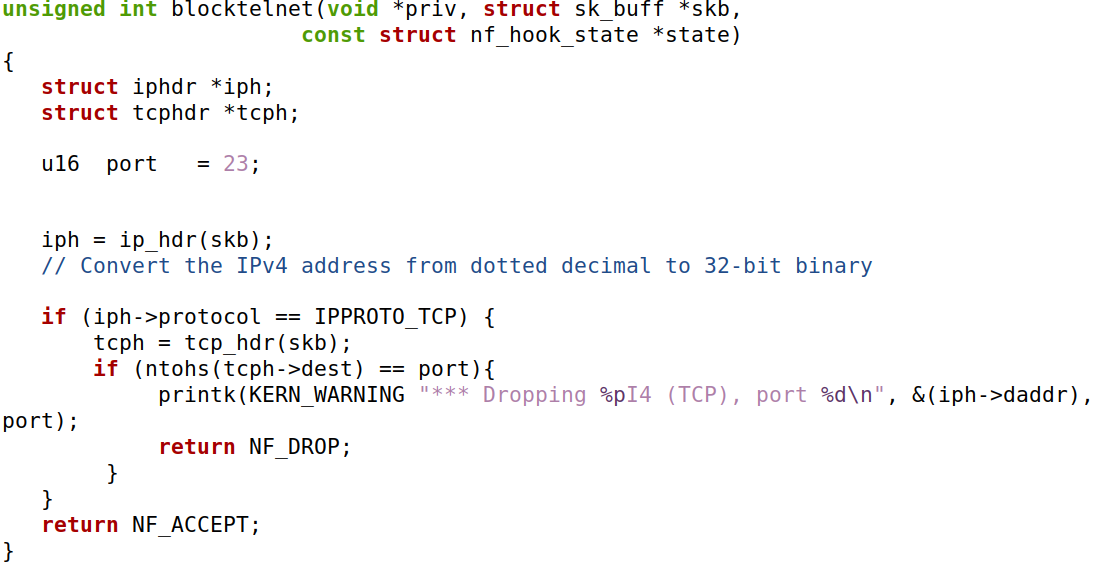
NF\_INET\_LOCAL\_IN 触发条件：发往本地而不需要转发的包

NF\_INET\_FORWARD 触发条件：需要向外转发的数据包

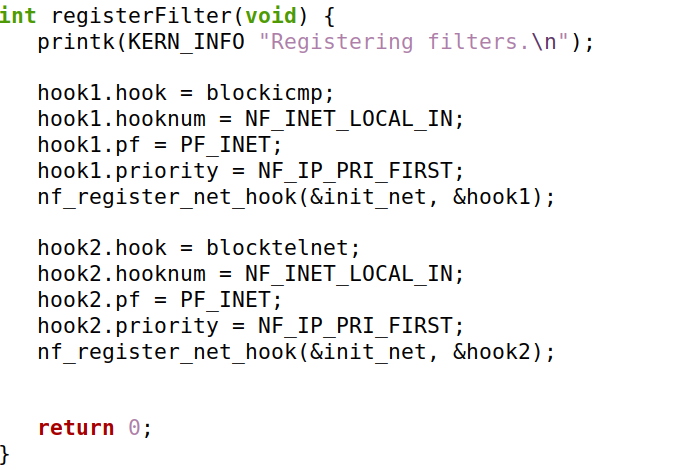
**Task1.B.3**

（1）增加两个bock函数，分别用于拦截icmp报文和telnet报文

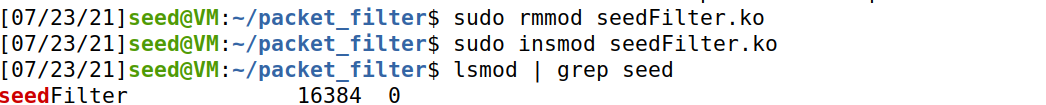




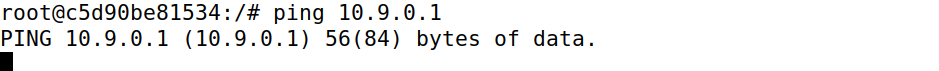
其中可以看到，这里没有判断目标ip是否为主机，这是因为利用task1.B.2中的知识，所有触发NF\_INET\_LOCAL\_IN的都是发往本地的包，因此无需再比较ip，只需要将函数注册为NF\_INET\_LOCAL\_IN点即可。



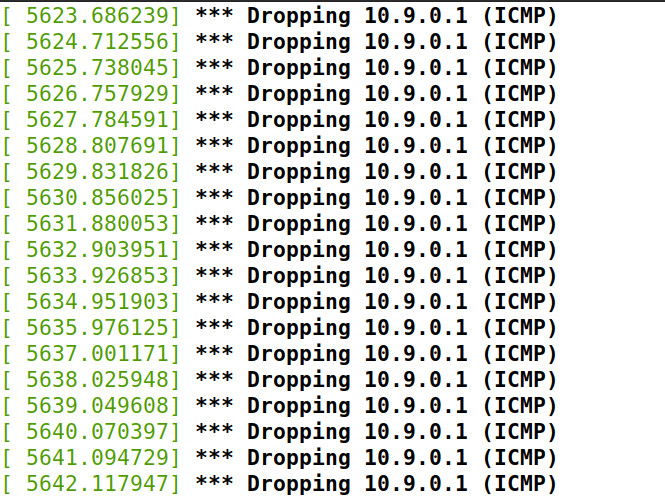
（2）重新编译并加载模块



（3）登录hostA主机，并ping 10.9.0.1（虚拟机），可以看到没有ping通

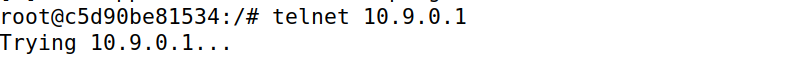


输入dmesg查看

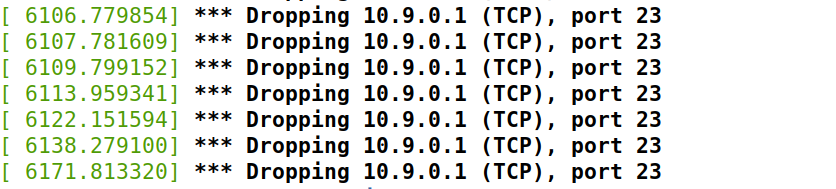


可以看到，icmp报文 都被丢弃了，说明拦截成功

（4）使用telnet 10.9.0.1尝试登录，可以看到登录失败



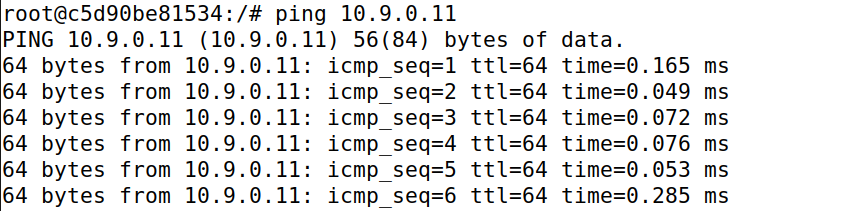
输入dmesg命令查看



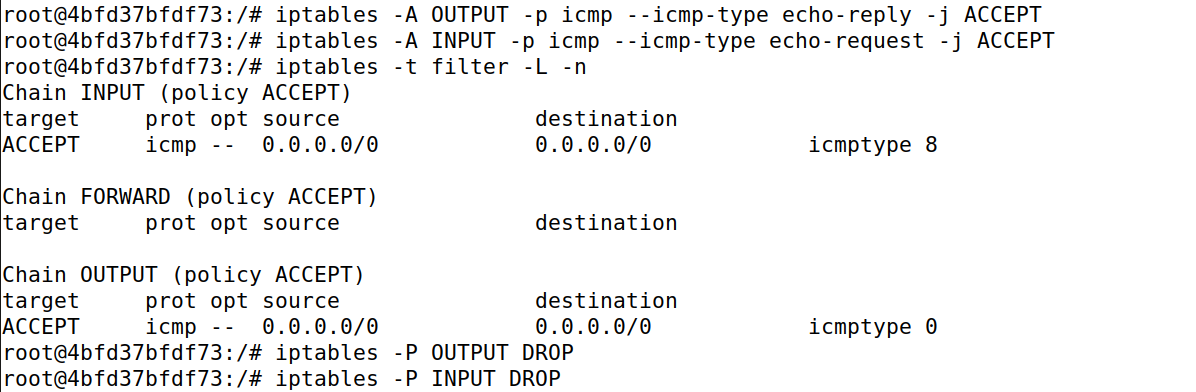
可以看到，丢弃了很多tcp报文，从而阻止了登录，说明拦截成功。

**Task2.A**

（1）未进行操作前，在主机上ping路由器

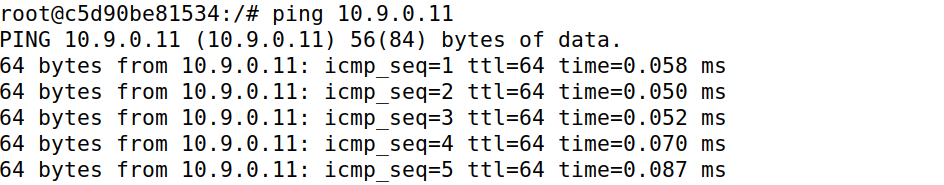


（2）输入四条规则（修改了手册上的规则）

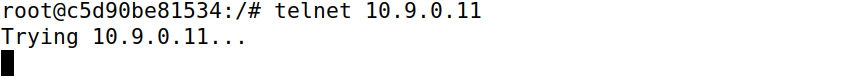


1. 测试

使用ping指令，成功



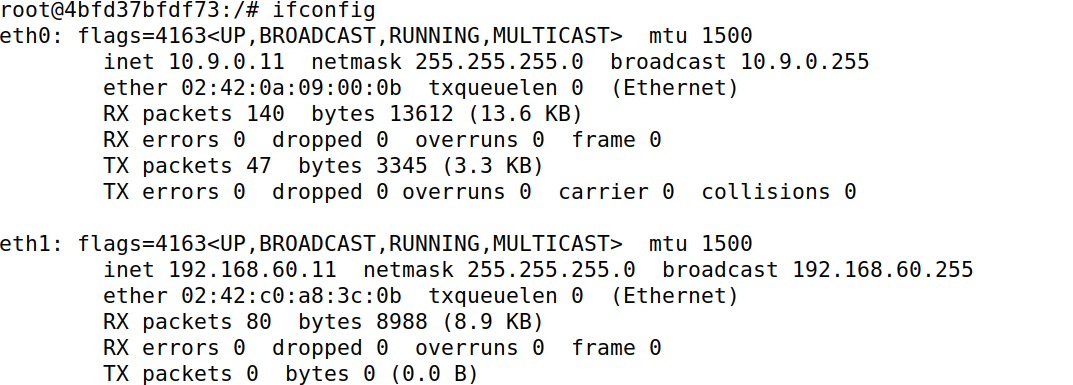
使用telnet指令，失败



说明规则编写正确。

**Task2.B**

1. 在路由器上输入ifconfig查看端口对应ip



可以看到10.9.0.11对应的端口(eth0)是面向外网的，192.168.60.11(eth1)对应的端口是面向内网的。

1. 根据要求指定规则如下：

①OUTPUT INPUT FORWARD丢弃  
iptables -P OUTPUT DROP

iptables -P INPUT DROP

iptables -P FORWARD DROP

（不允许内外流量交互）

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

对于FORWARD 只有icmp应答报文由外部端口（eth0）进入，外部端口（eth1），才接收：

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

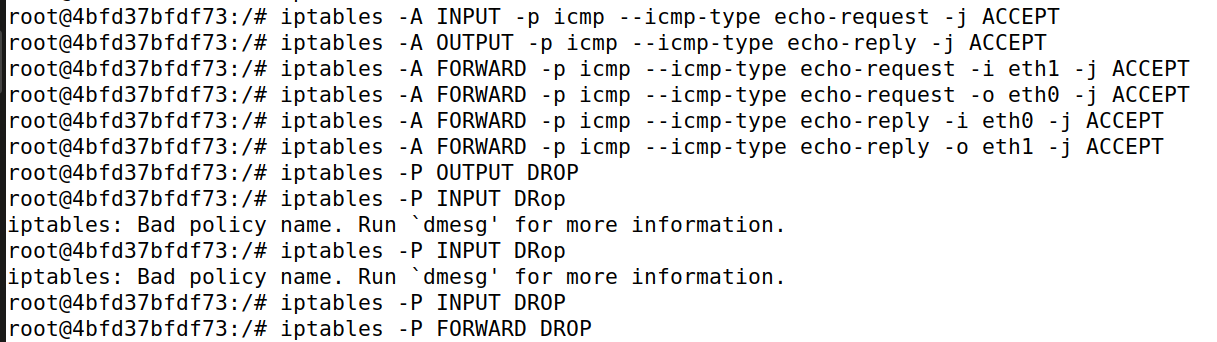
iptables -A FORWARD -p icmp --icmp-type echo-reply -o eth1 -j ACCEPT

（从而保证外部不能ping内部，内部能ping外部）

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

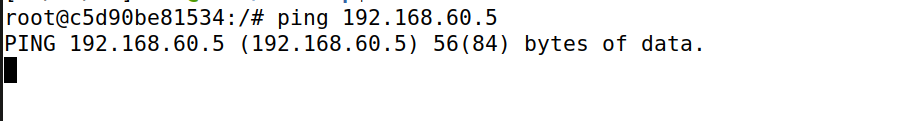
iptables -A INPUT -p icmp --icmp-type echo-request -j ACCEPT

iptables -A OUTPUT -p icmp --icmp-type echo-reply -j ACCEPT



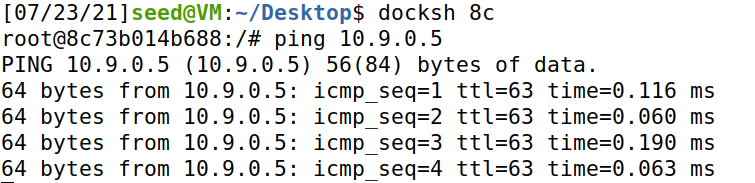
1. 测试

①外网ping内网



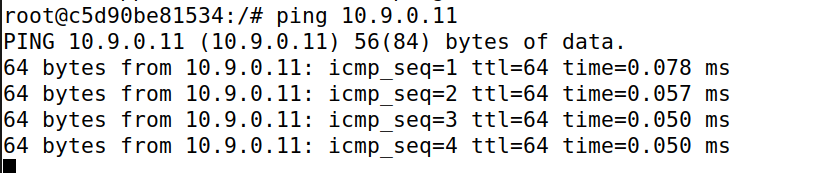
不能实现ping操作，符合要求。

②内网ping外网



能够实现ping操作，符合要求。

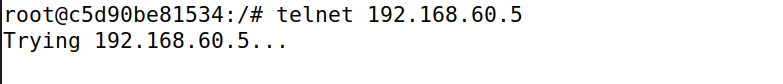
③外网ping路由



能够实现ping操作，符合要求。

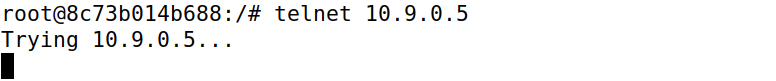
④内外其它流量的交互

外网telnet内网：



无法登录，符合要求。

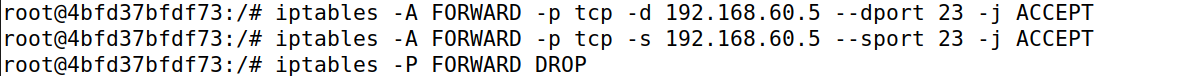
内网telnet外网：



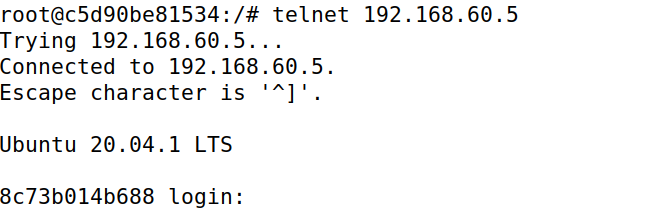
无法登录，符合要求。

**Task2.C**

1. 设计策略如下

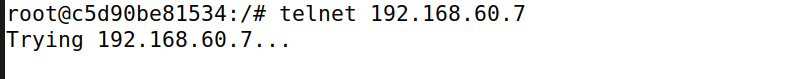
只允许192.168.60.5主机的23端口的流量进行转发，保证其能被外部和内部主机登录，而外部主机到其它内部主机则不行；本地主机的相互访问不需要转发，因此FORWARD对其它报文的丢弃不会影响内部主机间的登录。

①外部telnet192.168.60.5



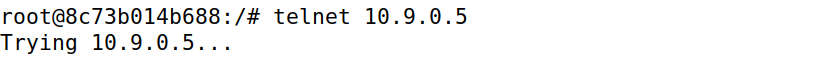
成功。

②外部telnet其它主机



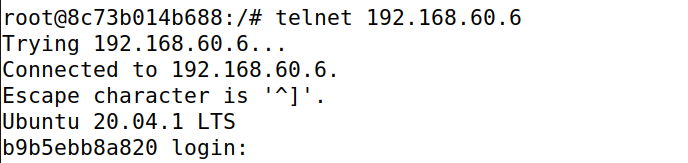
无法登录，符合要求。

③内部telnet外部



无法登录，符合要求。

④内部telnet内部

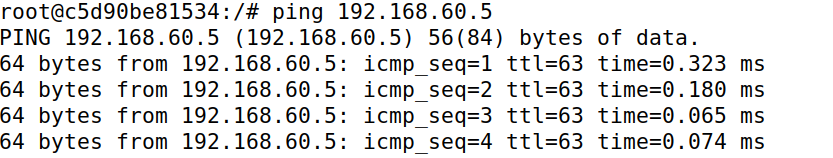


成功。

**Task3.A**

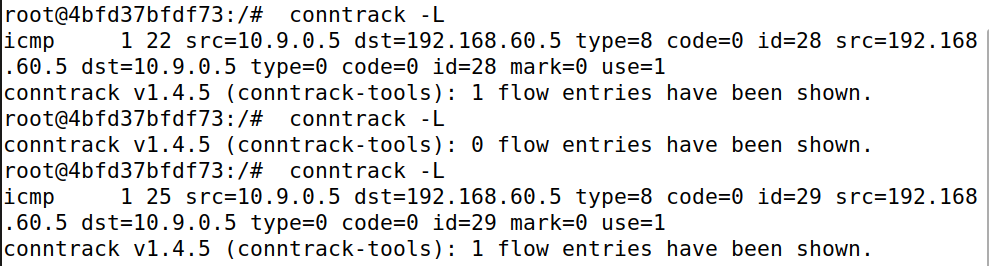
**ICMP**

（1）在10.9.0.5上ping 192.168.60.5主机

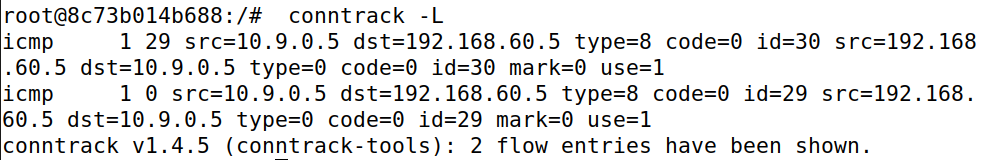


1. 查看追踪信息

可以看到一个icmp持续时间为30s，如果停止icmp，则会进入倒计时直至减少到0。

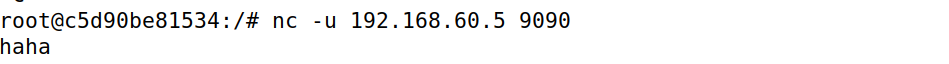


如果30s内再次ping，则会显示两个flow。

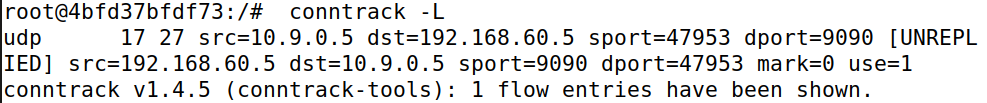
****

**UDP**

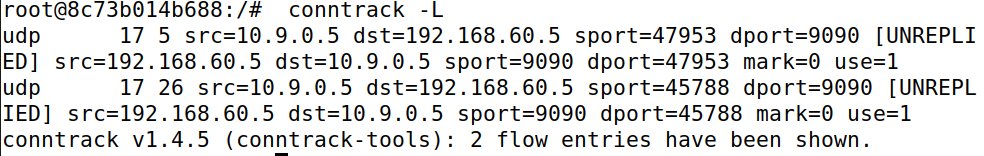
1. 在主机10.9.0.5上输入指令



（2）查看追踪信息

时间同样为30s，且停止登录后会倒计时到0。

同样，30s内再次连接会出现两条记录。



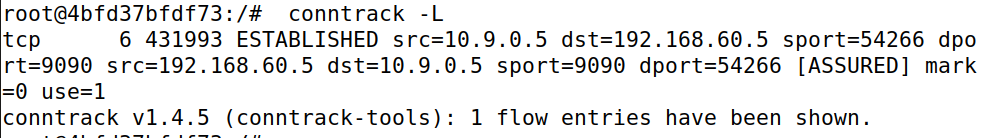
**TCP**

1. 在当前机和目标主机上输入指令，建立连接

J%I3OETI%ZB~3WO%~SU``1W

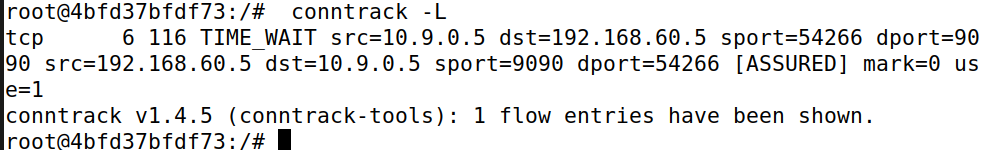
{JY{V]6A5DLBFY2_8)]N{$D

1. 查看追踪



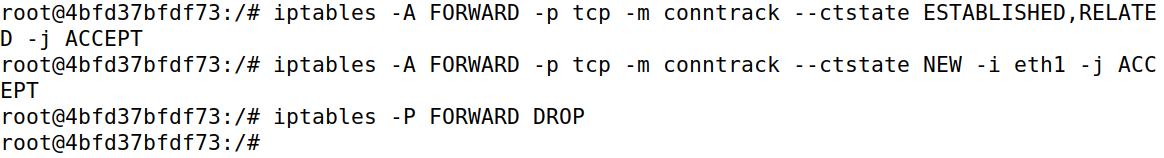
可以看到，存活时间在430000s以上

（3）结束连接，可以看到记录仍然存在，但只存在120s，120s倒计时结束后将消失。



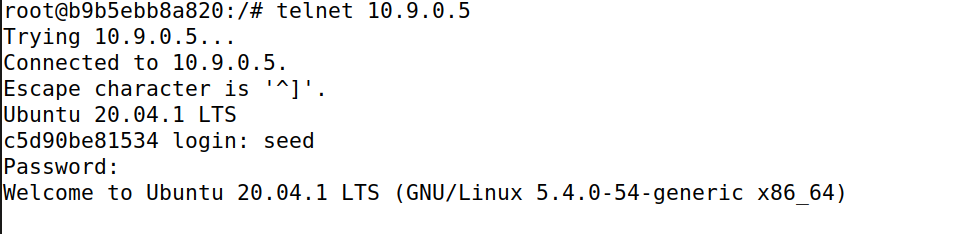
**Task3.B**

1. 编写规则如下

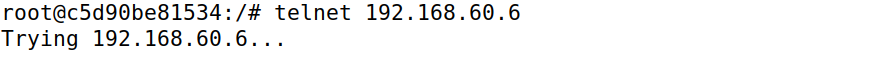


（2）测试

内网telnet外网，成功



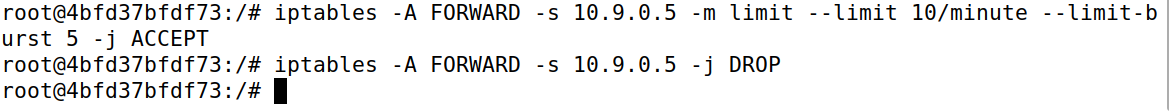
外网telnet内网，无法连接



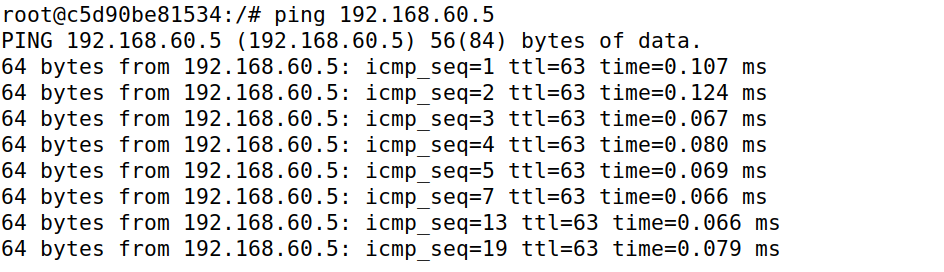
因此规则正确 ，符合实验要求。

**Task4**

（1）在路由器输入规则

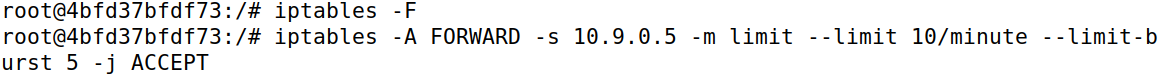


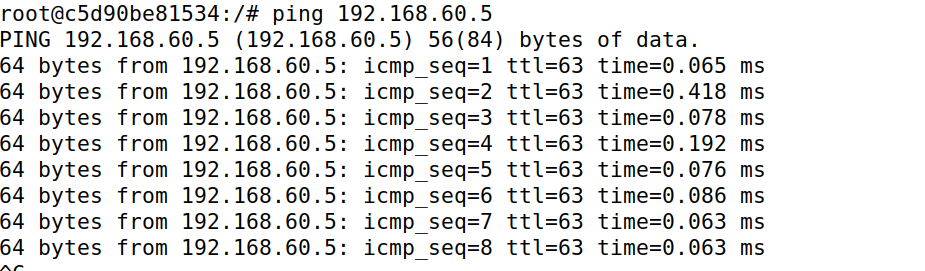
（2）在10.9.0.5主机尝试ping



可以看到，前几个报文速度非常快，后面速度较慢，平均6s一个，符合要求，说明规则发挥了作用

（3）去掉第二条规则，再次尝试ping





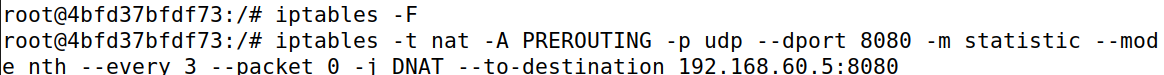
可以看到，报文发送速度很快，并没有减慢到每分钟10个，说明规则失效了。

这是因为没有第二条规则将报文设置为默认DROP，所有报文（无论是否满足第一个规则）都会从默认的ACCEPT规则通过，从而导致第一条规则也失效。

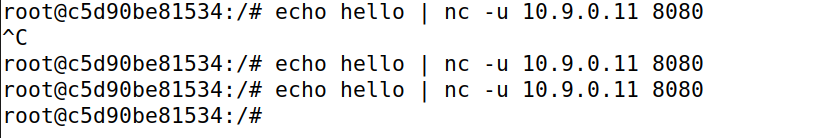
**Task5**

**轮询：**

1. 在路由器上输入规则



（2）在192.168.60.5上开启nc -luk 8080 监听，在10.9.0.5处连接并输入hello

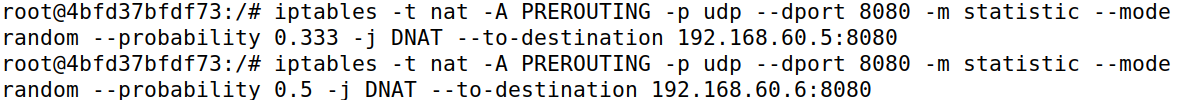


（3）在192.168.60.5上查看，可以看到发送者没发送3次，接收者再会收到1次路由器转发的报文，说明路由规则正确。

M_4KF_28S%NY~XG38UZUXER

**随机：**

1. 指定规则如下

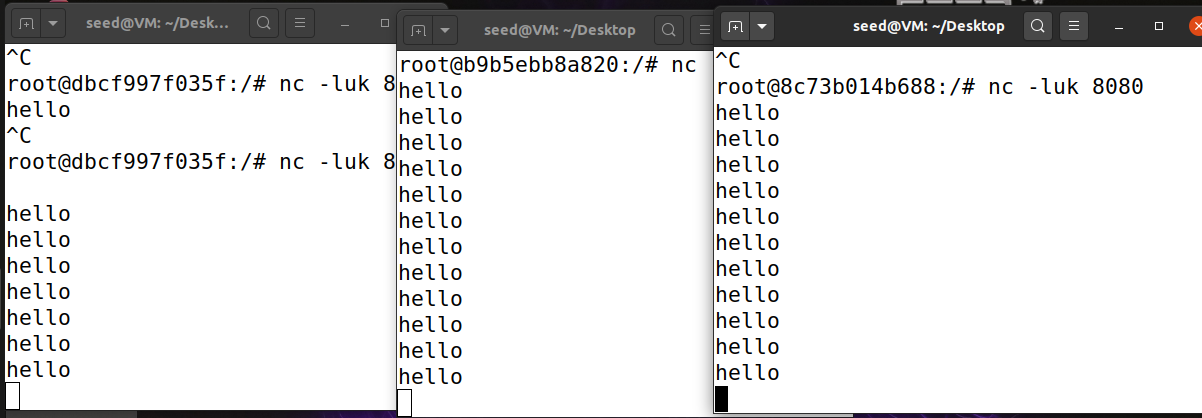


~_BL}FJJ6)V[G7)K_ZN2~CG

其中第一个的概率为0.333，第二个概率则应当在第一个规则没命中基础上考虑，即为0.5，

当前两个没有命中，则默认命中第三条

（2）在三台主机上都输入nc -luk 8080打开监听，在10.9.0.5主机上建立连接并不断输入hello，得到三台主机上的输出结果如下：



可以看到路由确实实现了分发，由于样本数量太少，不能做到每个主机上收到的报文各1/3,但总体上符合负载均衡的要求。