# 一、实验目的

1. 掌握Linux下防火墙的原理和配置方法

2. 掌握如何设计防火墙的高级过滤策略以抵御各种类型的攻击

# 二、实验内容

1. 配置Linux下的一组防火墙规则，对内网、外网以及DMZ区交互的访问进行有效约束

2. 防止端口扫描；防止 ping 攻击；防止 ip 碎片攻击；丢弃坏的 TCP包；防止 SYN攻击

# 三、实验原理

防火墙作为一种访问控制的机制是确保网络安全的重要手段。针对不同的需求和应用环境，可以量身定制出不同的防火墙系统。在众多网络防火墙产品中，Linux操作系统上的防火墙软件特点显著。首先是Linux操作系统在系统稳定性、健壮性及价格的低廉性方面都独具优势；更为重要的是系统包含了建立Internet所需的所有服务软件包。

Linux提供的防火墙软件包内置于Linux内核中，是一种基于包过滤型的防火墙实现技术。其中心思想是根据网络层IP包头中的源地址、目的地址及包类型等信息来控制包的流向。更彻底地过滤规则是检查包中的源端口、目的端口以及连接状态等信息。在Linux 2.4以上版的内核应用里一个全新的包过滤管理工具Iptables，可以加入、插入或删除包过滤表格中的规则。

1）iptables简介

netfilter/iptables（简称为iptables）组成Linux平台下的包过滤防火墙，与大多数的Linux软件一样，这个包过滤防火墙是免费的，它可以代替昂贵的商业防火墙解决方案，完成封包过滤、封包重定向和网络地址转换（NAT）等功能

2）iptables基础

规则（rules）其实就是网络管理员预定义的条件，规则一般的定义为“如果数据包头符合这样的条件，就这样处理这个数据包”。规则存储在内核空间的信息 包过滤表中，这些规则分别指定了源地址、目的地址、传输协议（如TCP、UDP、ICMP）和服务类型（如HTTP、FTP和SMTP）等。当数据包与规 则匹配时，iptables就根据规则所定义的方法来处理这些数据包，如放行（accept）、拒绝（reject）和丢弃（drop）等。配置防火墙的 主要工作就是添加、修改和删除这些规则。

iptables的原理主要是对数据包的控制，如图1所示：

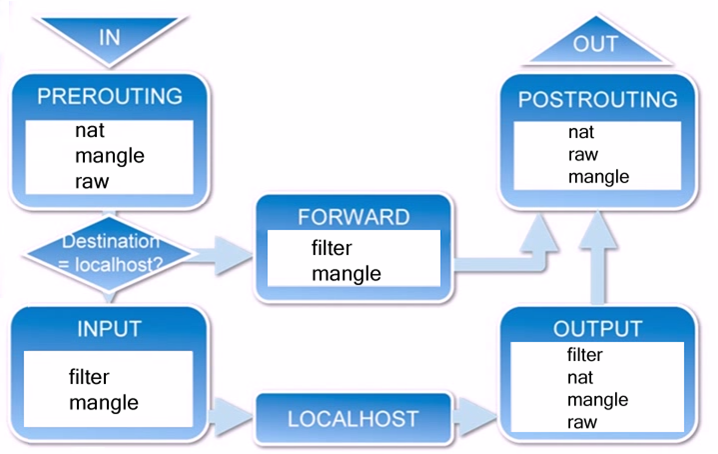


图1 数据包处理流程图

（1）一个数据包进入网卡时，它首先进入PREROUTING链，内核根据数据包目的IP判断是否需要转发出去。

（2）如果数据包就是进入本机的，它就会沿着图向下移动，到达INPUT链。数据包到了INPUT链后，任何进程都会收到它。本机上运行的程序可以发送数据包，这些数据包会经过OUTPUT链，然后到达POSTROUTING链输出。

（3）如果数据包是要转发出去的，且内核允许转发，数据包就会如图所示向右移动，经过FORWARD链，然后到达POSTROUTING链输出。

注：[详细介绍](https://www.jianshu.com/p/ee4ee15d3658)

# 四、实验环境

（1）装有Linux系统的虚拟机

（2）Wireshark等抓包工具

（3）Nmap端口扫描工具

sudo apt-get install nmap

# 五、实验步骤

（1）搭建实验环境

将虚拟机IP地址配置到和宿主机[一个网段](https://blog.csdn.net/oNelson123/article/details/125417115?ops_request_misc=%257B%2522request%255Fid%2522%253A%2522169569161416800215073919%2522%252C%2522scm%2522%253A%252220140713.130102334..%2522%257D&request_id=169569161416800215073919&biz_id=0&utm_medium=distribute.pc_search_result.none-task-blog-2~all~sobaiduend~default-1-125417115-null-null.142%5ev94%5echatsearchT3_1&utm_term=ubuntu%2020.4%E4%BF%AE%E6%94%B9ip%E5%9C%B0%E5%9D%80%E5%90%8E%E5%88%B7%E6%96%B0&spm=1018.2226.3001.4187)即可，注意地址不可与其他地址冲突。

关闭宿主机防火墙后，此时A和B是可以互ping的，如图2与图3所示：

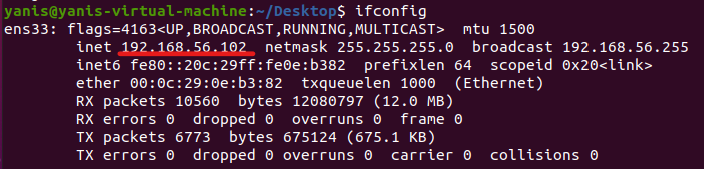


图2 虚拟机IP

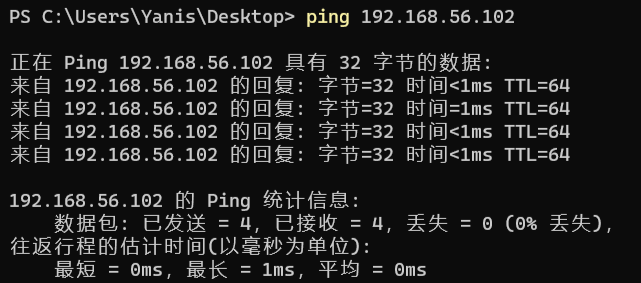


图3 宿主机ping通虚拟机

（2）启动虚拟机的防火墙

输入如下指令，启动成功后的结果如图4所示：

sudo modprobe ip\_tables //加载ip\_table内核模块

sudo ufw enable //启用UFW（Uncomplicated Firewall）并启动防火墙服务

sudo ufw reload //重新加载UFW配置

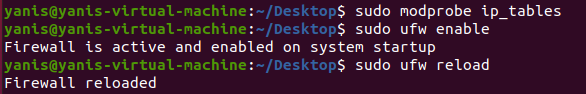


图4 成功启动防火墙

（3）清除filter表中的规则，设置默认规则为丢弃一切报文

sudo iptables -F //清空iptables防火墙规则

sudo iptables -P INPUT DROP //将输入流量的默认策略设置为丢弃

sudo iptables -P OUTPUT DROP //将输出流量的默认策略设置为丢弃

sudo iptables -P FORWARD DROP //将转发流量的默认策略设置为丢弃

（4）进行ping命令限制并检验，

1）其它机器不可以ping主机A，主机A可以 ping其它机器，实验结果如图5和图6所示：

sudo iptables -A INPUT -p icmp -m state --state ESTABLISHED,RELATED -j ACCEPT //允许系统接受来自已经建立的 ICMP 连接或相关 ICMP 流量

sudo iptables -A OUTPUT -s 192.168.56.102 -d 192.168.56.1 -p icmp -j ACCEPT //允许源IP地址的主机发送ICMP流量（如ping）到目标IP地址

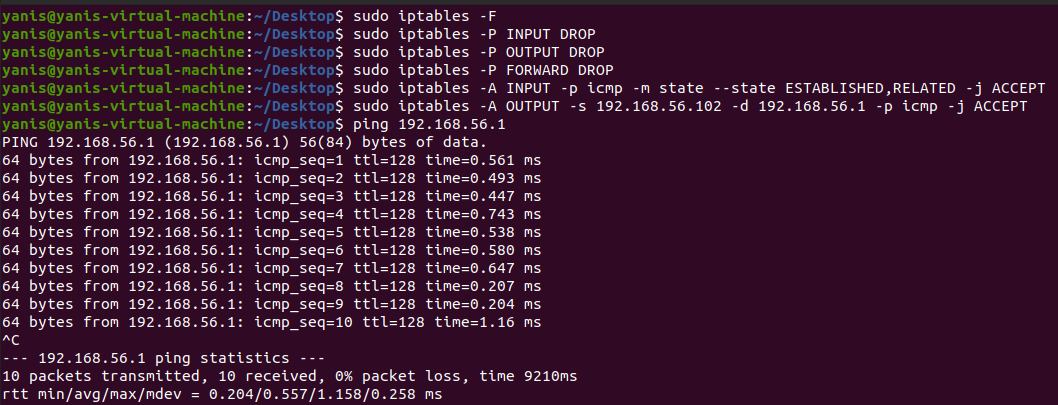


图5 主机A ping 主机B



图6 主机B ping 主机A

2）其它机器可以ping主机A ，阻止主机A ping其它机器，结果如图7和图8所示：

sudo iptables -F OUTPUT

sudo iptables -F INPUT（先要把之前的规则去掉）

sudo iptables -A OUTPUT -p icmp -m state --state ESTABLISHED,RELATED -j ACCEPT

sudo iptables -A INPUT -s 192.168.56.1 -d 192.168.56.102 -p icmp -j ACCEPT

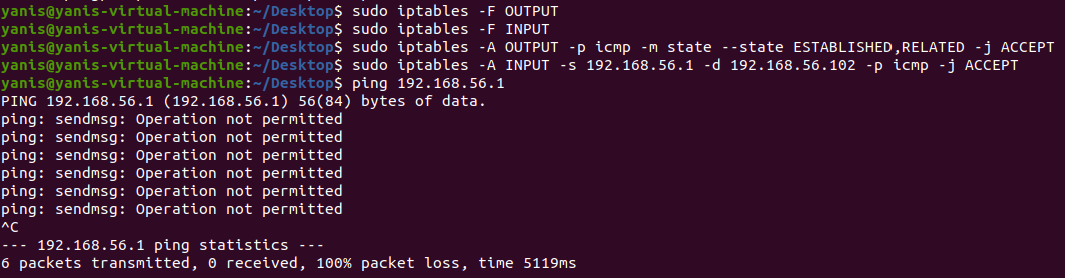


图7 主机A ping 主机B

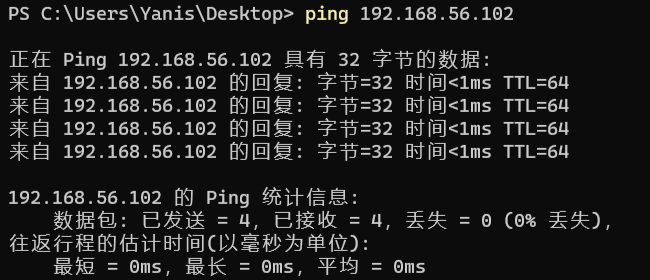


图8 主机B ping 主机A

3）主机A可以ping其它机器，也允许其它机器ping主机A，结果如图9和图10所示：

sudo iptables -F OUTPUT

sudo iptables -F INPUT

sudo iptables -A INPUT -s 192.168.43.91 -d 192.168.43.227 -p icmp -j ACCEPT

sudo iptables -A OUTPUT -s 192.168.56.1 -d 192.168.56.102 -p icmp -j ACCEPT

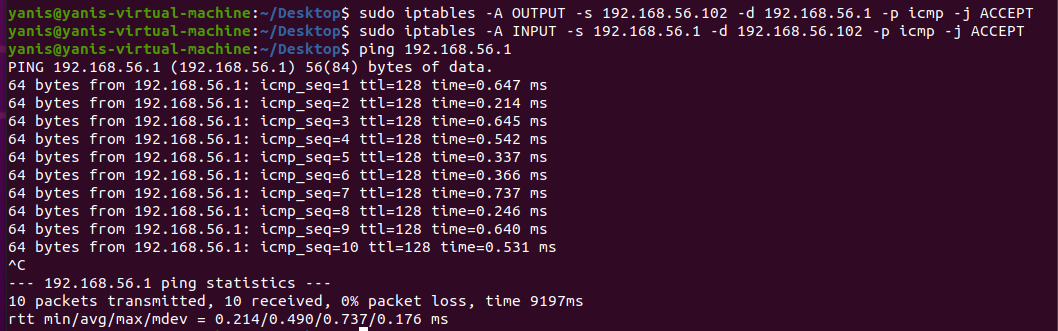


图9 主机A ping 主机 B



图10 主机B ping 主机A

（5）防止端口扫描，在虚拟机中输入（清空之前规则）：

sudo iptables -A INPUT -i ens33 -p tcp --tcp-flags SYN,FIN SYN,FIN -j DROP

先查看是否安装ssh

dpkg -l | grep ssh

sudo apt-get install openssh-server

用[nmap扫描端口](https://blog.csdn.net/jimmyleeee/article/details/127756390?spm=1001.2101.3001.6650.2&utm_medium=distribute.pc_relevant.none-task-blog-2%7Edefault%7EBlogCommendFromBaidu%7ERate-2-127756390-blog-124497882.235%5Ev38%5Epc_relevant_anti_t3&depth_1-utm_source=distribute.pc_relevant.none-task-blog-2%7Edefault%7EBlogCommendFromBaidu%7ERate-2-127756390-blog-124497882.235%5Ev38%5Epc_relevant_anti_t3&utm_relevant_index=3)

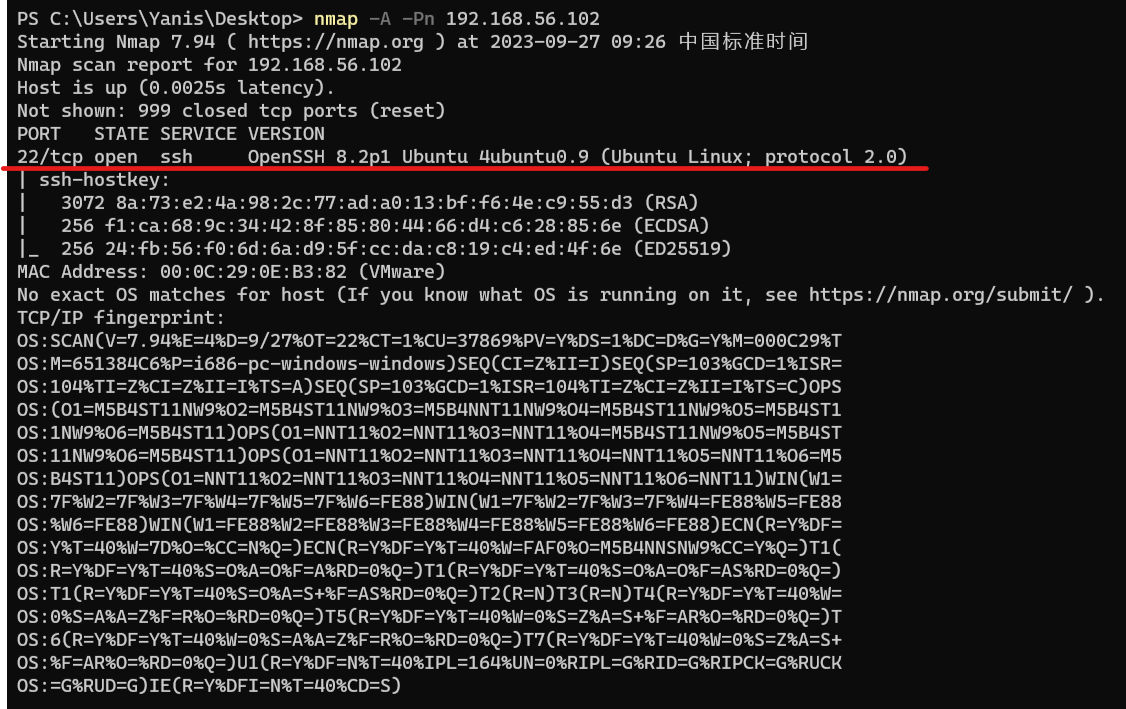


图11 输入指令前的端口扫描情况

sudo iptables -F

sudo iptables -A INPUT -p tcp --tcp-flags ALL FIN,URG,PSH -j REJECT //拒绝带有同时包含FIN、URG和PSH标志的传入TCP连接

sudo iptables -A INPUT -p tcp --tcp-flags SYN,RST SYN,RST -j REJECT //拒绝带有同时包含SYN和RST标志的传入TCP连接（SYN标志用于建立连接，RST标志用于中断连接）

sudo iptables -A INPUT -p tcp --tcp-flags SYN,FIN SYN,FIN -j REJECT // FIN标志用于终止连接

sudo iptables -A INPUT -j REJECT --reject-with icmp-host-prohibited //拒绝所有传入的网络流量，并向源主机发送 ICMP-host-prohibited错误消息

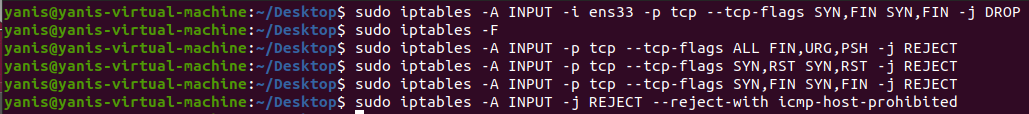


图12 输入防止端口扫描指令

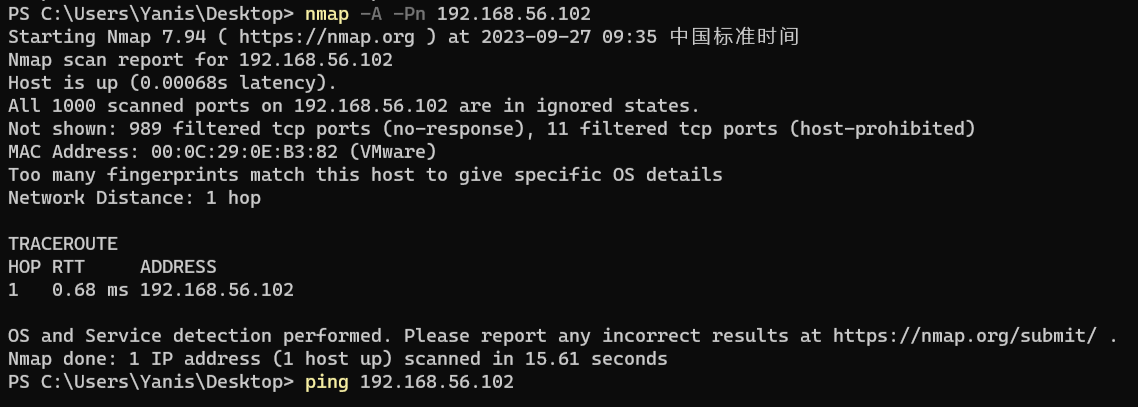


图13 指令输入后端口扫描结果

注：可以用nmap工具可进行端口扫描测

（6）防止 ping 攻击，允许一个包每秒，触发条件是10个包，输入（清空之前规则）：

sudo iptables -A INPUT -p icmp -m limit --limit 1/s --limit-burst 10 -j ACCEPT

sudo iptables -A OUTPUT -s 192.168.19.133 -d 192.168.19.1 -p icmp -j ACCEPT

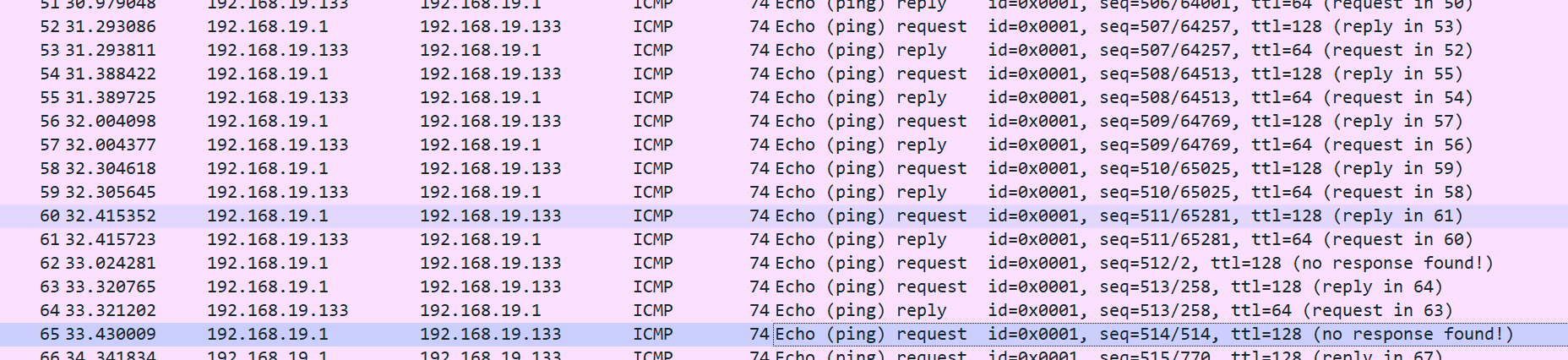


图14 Wireshark抓包情况

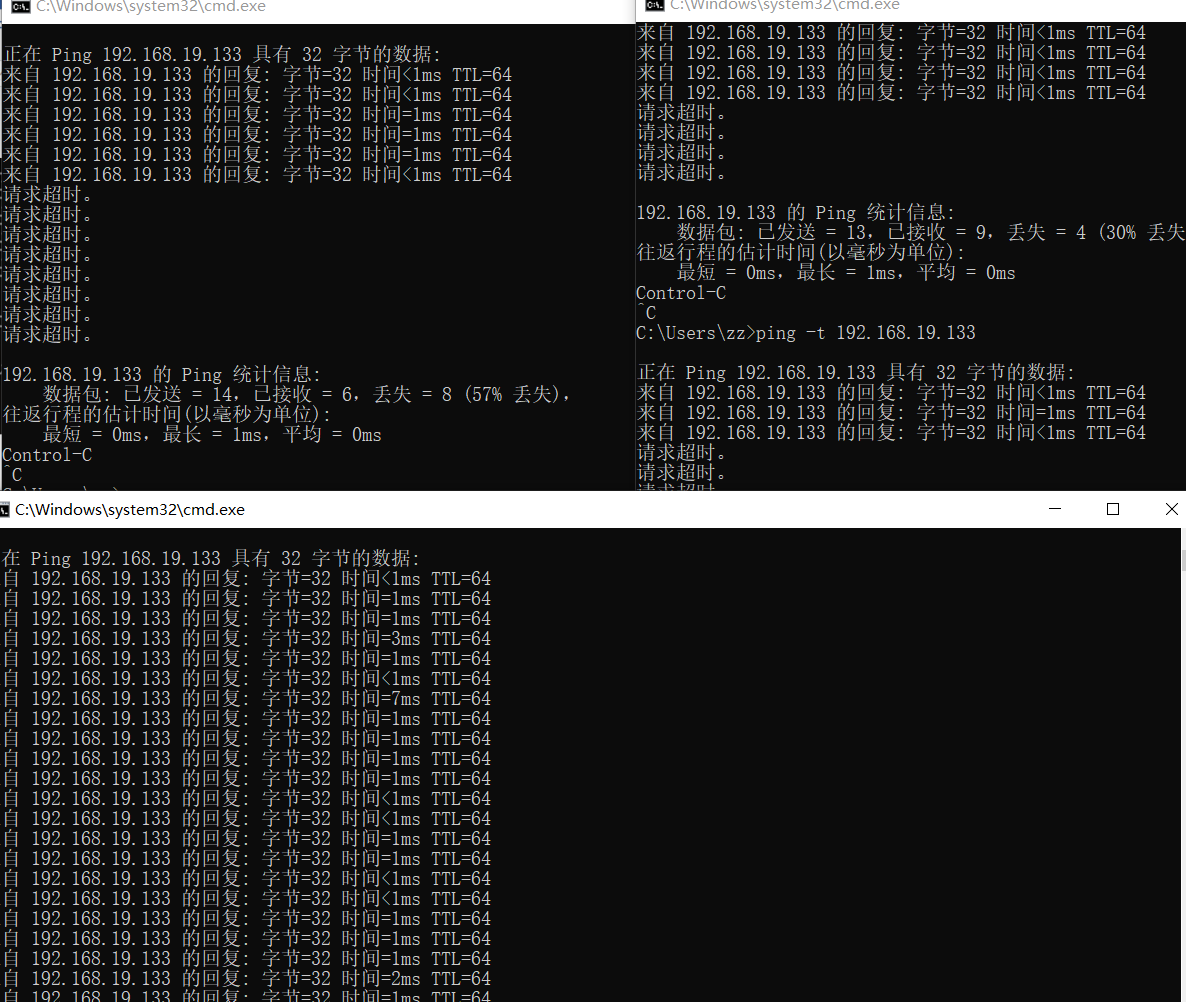


图15 宿主机B ping 主机A

（7）防止 ping 攻击，允许一个包每分钟，触发条件是10个包，输入（清空之前规则）：

sudo iptables -A INPUT -p icmp -m limit --limit 1/m --limit-burst 10 -j ACCEPT

sudo iptables -A OUTPUT -s 192.168.19.133 -d 192.168.19.1 -p icmp -j ACCEPT

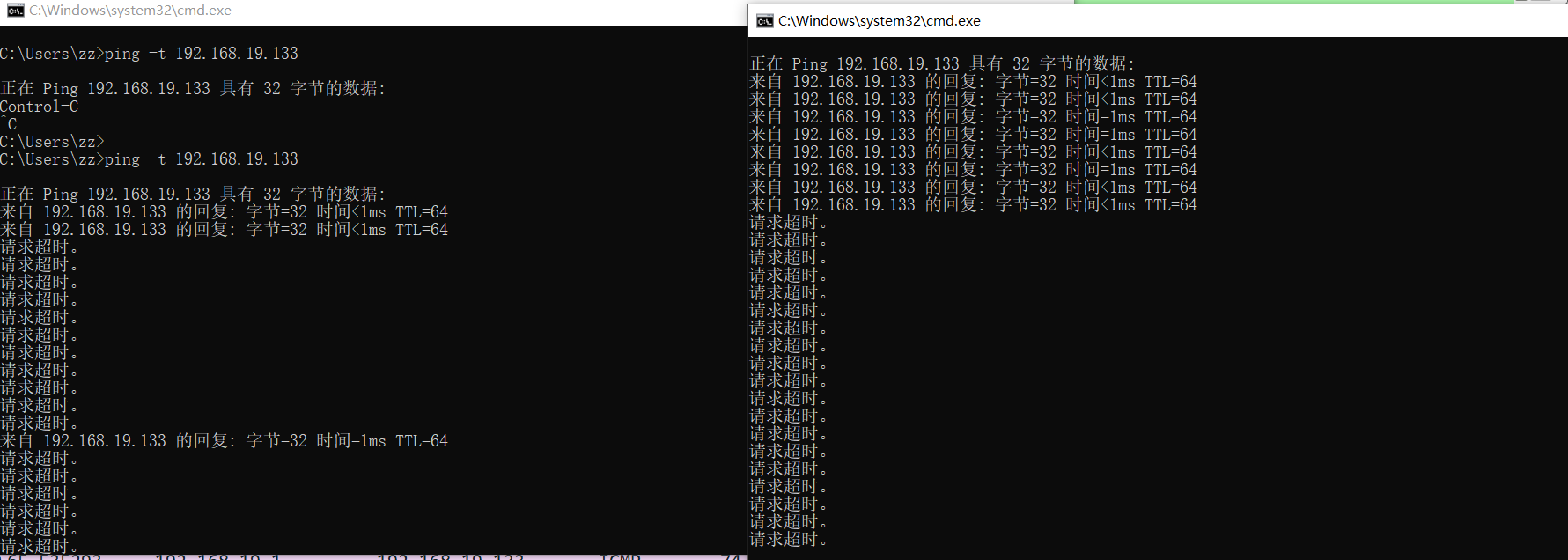
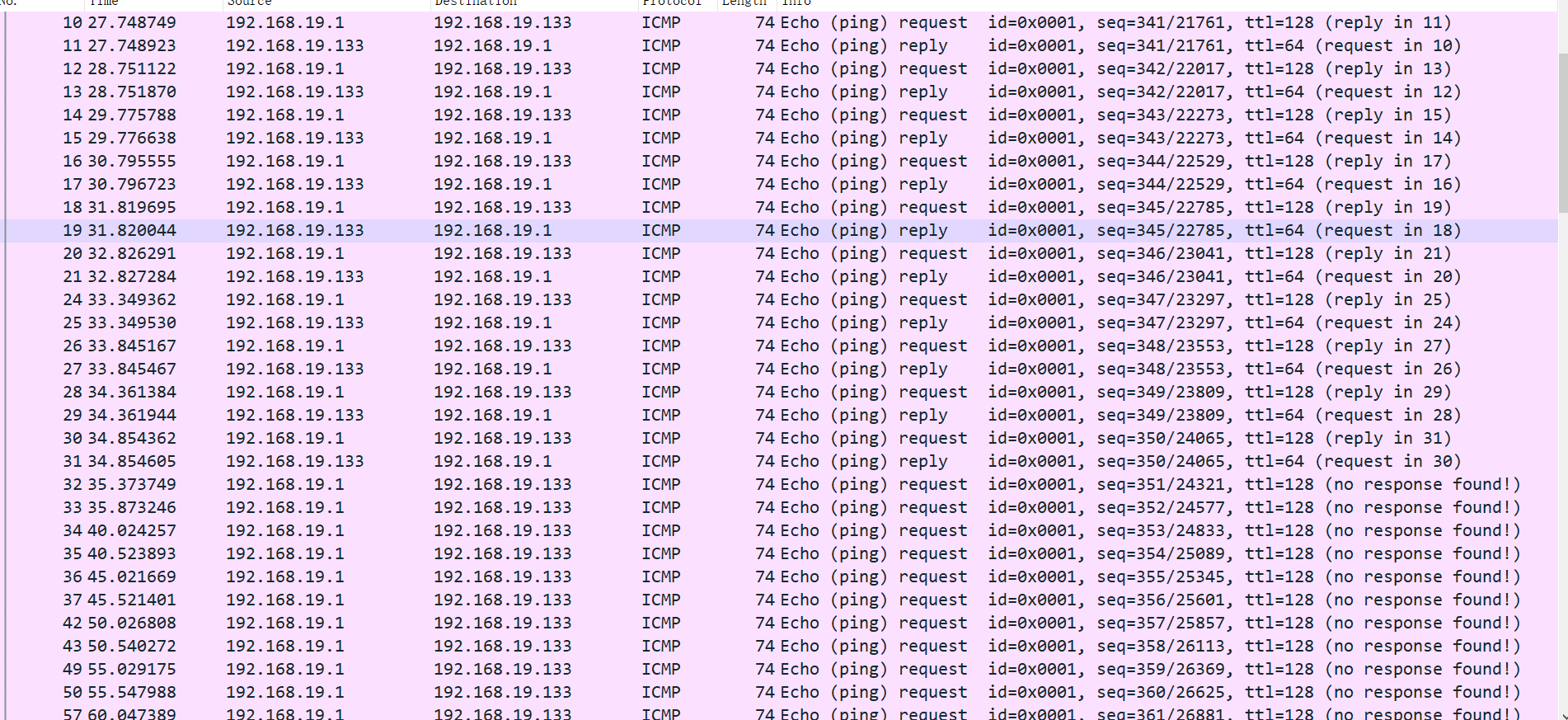


图16 宿主机B ping 主机A



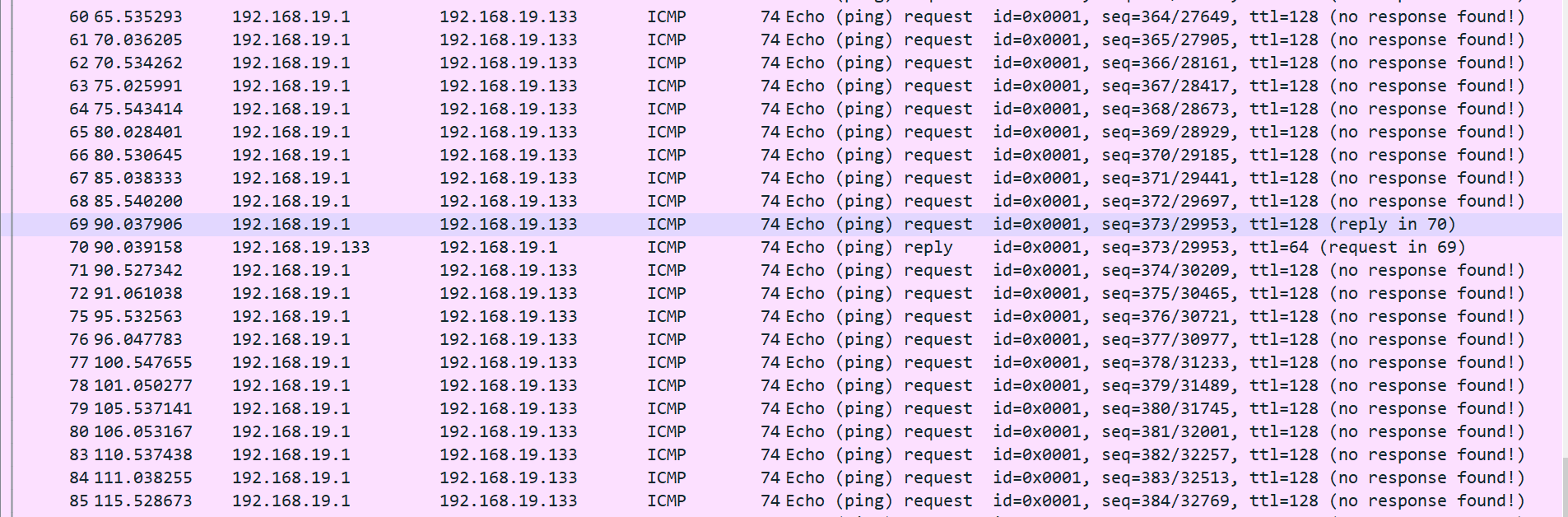


图17 Wireshark抓包情况

（8）对于不管来自哪里的 ip 碎片都进行控制,允许每秒通过 100 个碎片，触发条件是100个包，输入（清空之前规则）：

sudo iptables -A FORWARD -f -m limit --limit 100/s --limit-burst 100 -j ACCEPT

sudo iptables -A OUTPUT -s 192.168.19.133 -d 192.168.19.1 -p icmp -j ACCEPT

（9）丢弃坏的 TCP 包 ，输入（清空之前规则）：

sudo iptables -A FORWARD -p TCP ! --syn -m state --state NEW -j DROP

5#防止 SYN攻击

sudo iptables -N syn-flood

sudo iptables -A INPUT -p tcp --syn -j syn-flood

sudo iptables -A syn-flood -p tcp -m limit --limit 3/s --limit-burst 6 -j RETURN

sudo iptables -A syn-flood -j REJECT

（10）屏蔽指定ip，输入（清空之前规则）：

有时候我们发现某个ip不停的往服务器发包，这时我们可以使用以下命令，将指定ip发来的包丢弃：

BLOCK\_THIS\_IP="x.x.x.x"  
iptables -A INPUT -i eth0 -p tcp -s "$BLOCK\_THIS\_IP" -j DROP

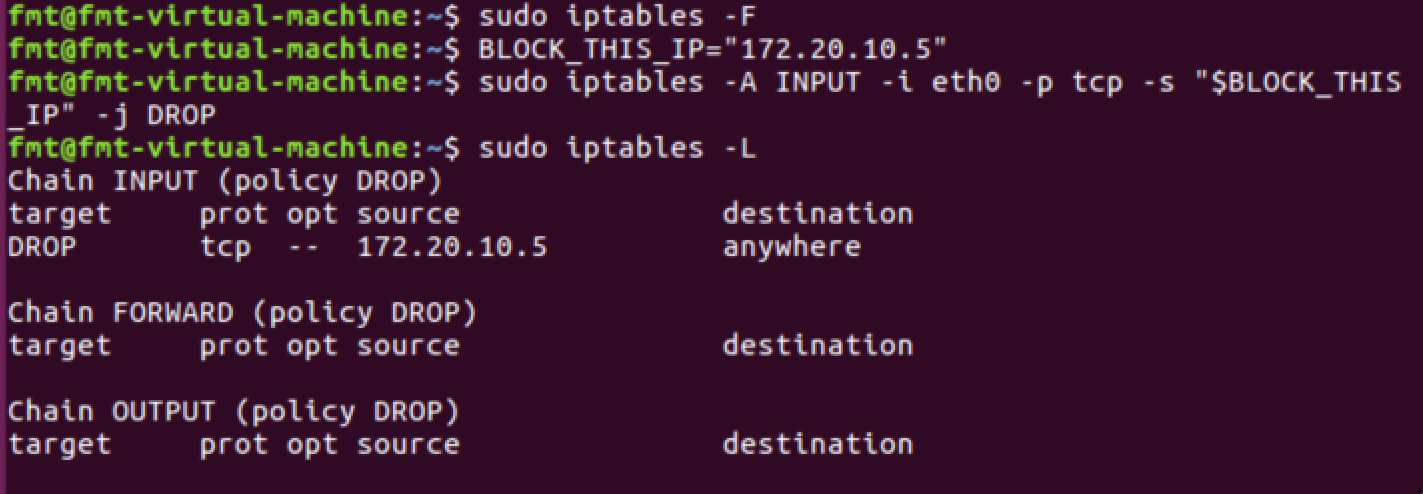


图18 屏蔽指定IP

（11）配置服务项

利用iptables，我们可以对日常用到的服务项进行安全管理，比如设定只能通过指定网段、由指定网口通过SSH连接本机：

iptables -A INPUT -i eth0 -p tcp -s 192.168.100.0/24 --dport 22 -m state --state NEW,ESTABLISHED -j ACCEPT  
iptables -A OUTPUT -o eth0 -p tcp --sport 22 -m state --state ESTABLISHED -j ACCEPT



图19 配置服务项

（12）网口转发配置

对于用作防火墙或网关的服务器，一个网口连接到公网，其他网口的包转发到该网口实现内网向公网通信，假设eth0连接内网，eth1连接公网，配置规则如下：

iptables -A FORWARD -i eth0 -o eth1 -j ACCEPT

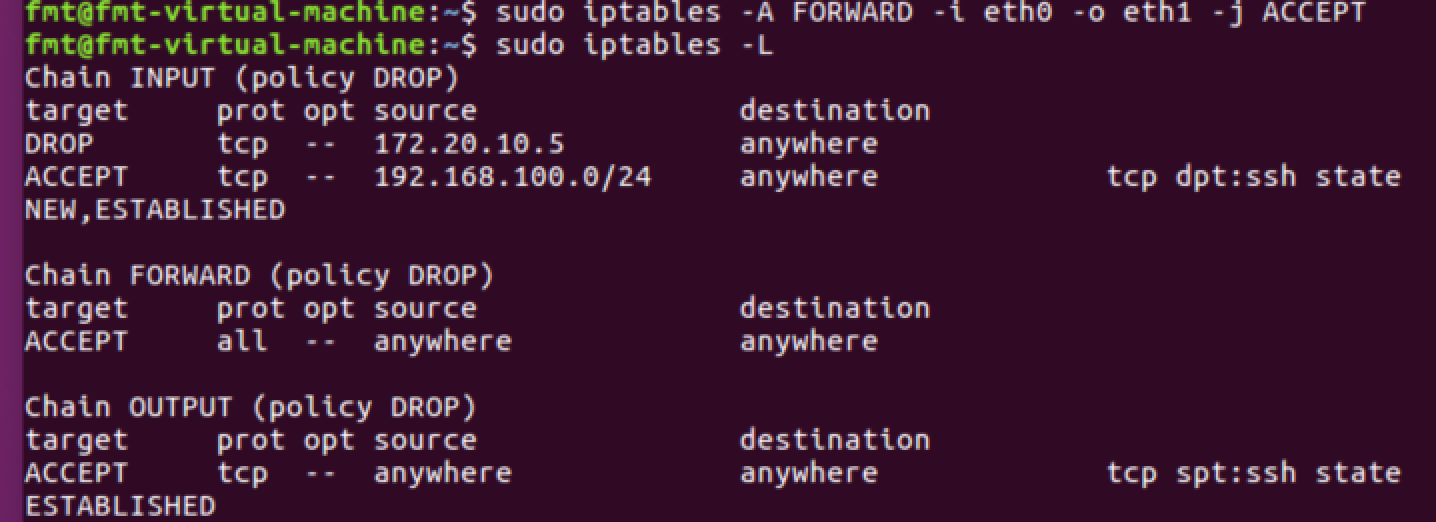


图20 网络转发配置

# 六、思考题

1.所有链的预设规则应该设置为什么？为什么？（DROP、ACCEPT？）

2.包过滤防火墙有什么优缺点？

3.如果规则链中两条规则是互相矛盾的，如前一条是禁止某个端口，后一条是打开这个端口，会出现什么情况？

4.如何限制WPS联网服务？ 如何限制网易云音乐的联网服务？

5.在过滤规则条数很多的情况下，应该如何设计规则链表，来提高 Iptables的过滤效率？

6.分析若干种现有流行的攻击方法，如何有针对性地设计 Iptables 的高级过滤规则？

# 七、考核标准

1.完成基本的实验环境配置，每一块实验内容能完成其中两小问的可得60分，若这个最基本的都没有完成，且对不会演示实验操作过程，直接不及格；

2.完成基本实验内容，每块实验内容既有1-2小题没有实现成功，且回答不上来针对操作提问的得65分；每块实验内容既有1-2小题没有实现成功但是验收过程中对实验原理熟悉操作熟练，现场提问略知一二的得75分；

3.完成大部分实验任务，通篇只有1-2个小环节有错，现场提问基本能回答出来的得80-85分；

4.完成所有实验任务，实验原理掌握清晰，分析细致，现场提问回答较好的得90-100分；