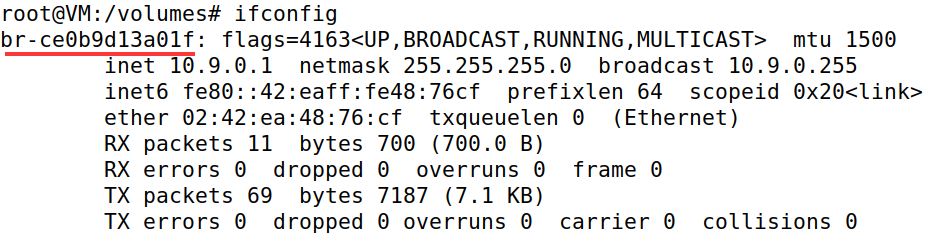
**Packet Sniffifing and Spoofifing Lab**

**57118214陈佳杰**

**Lab Task Set 1: Using Scapy to Sniff and Spoof Packets**

**Task 1.1: Sniffifing Packets**

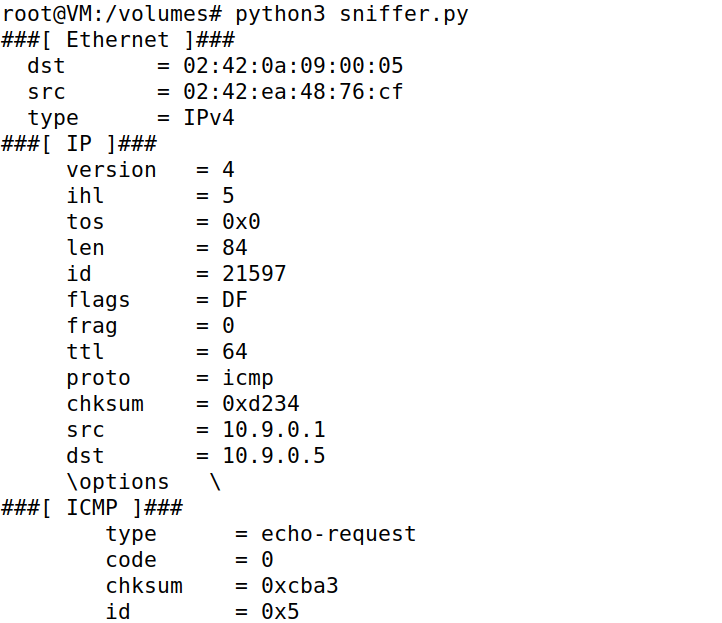
获取接口名称

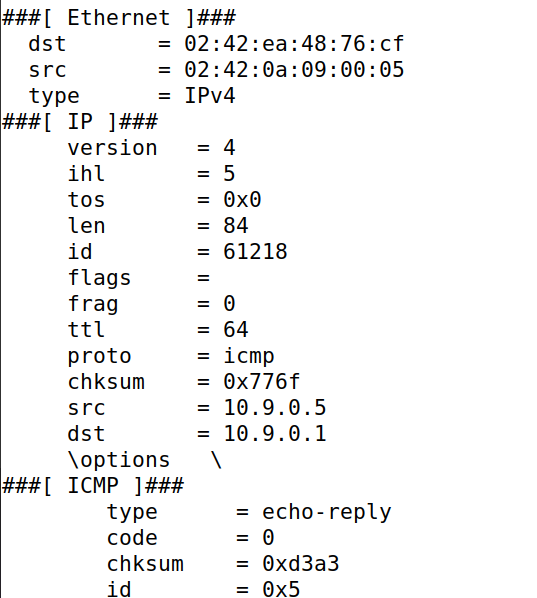
  
使用Scapy进行包嗅探的Python程序

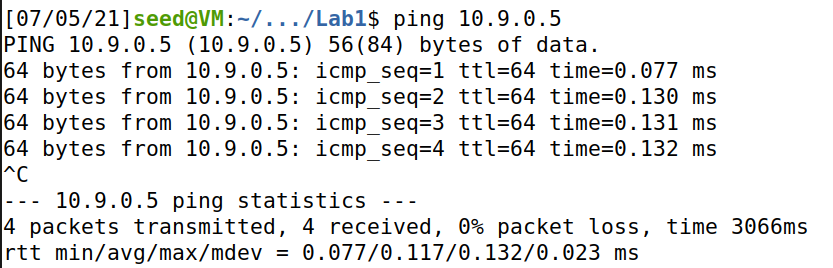


**Task 1.1A.**

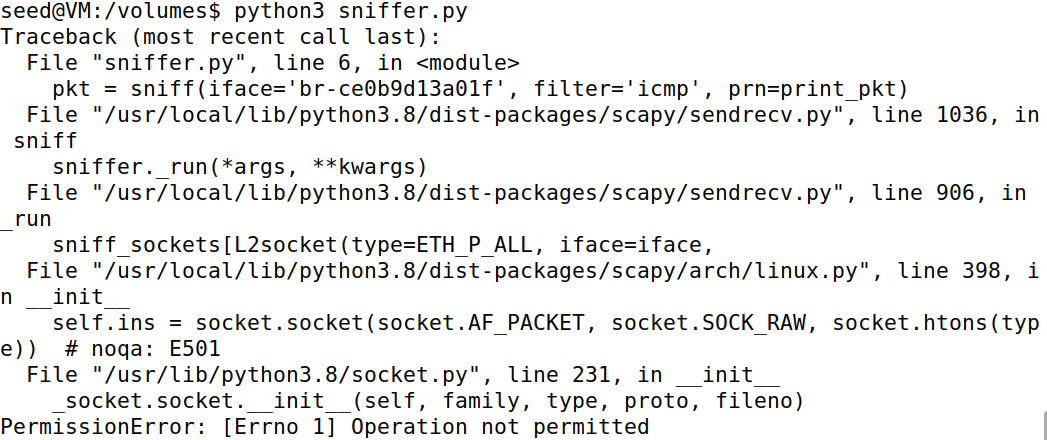
在docker中运行sniffer.py，同时ping 10.9.0.5，可以嗅探到数据包，结果如下所示。







用seed用户来运行sniffer.py，发现无法运行，因为sniff函数需要较高的权限才能运行。



**Task 1.1B.**

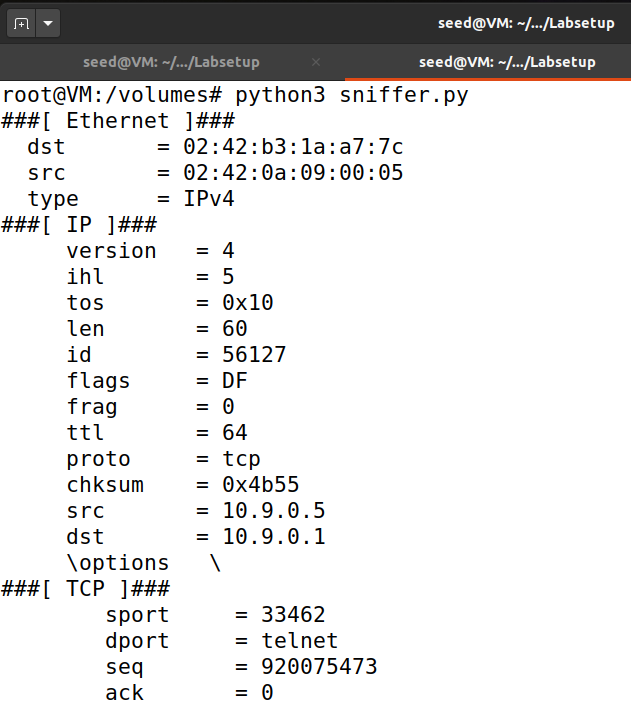
①Capture only the ICMP packet

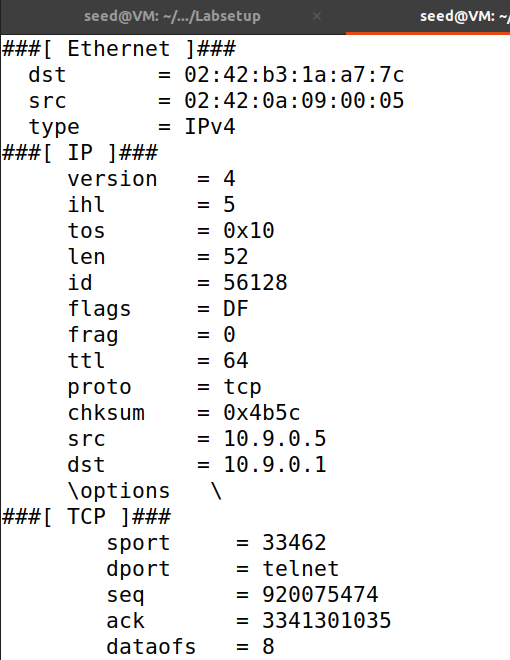
实验内容与Task 1.1A.一样，filter='icmp'

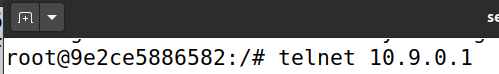
②Capture any TCP packet that comes from a particular IP and with a destination port number 23.

设置过滤器为filter='src host 10.9.0.5 and tcp dst port 23'

运行sniffer.py后在host中telnet 10.9.0.1可获取以下报文





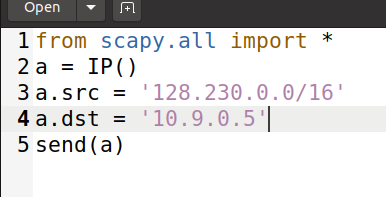


③Capture packets comes from or to go to a particular subnet. You can pick any subnet, such as 128.230.0.0/16; you should not pick the subnet that your VM is attached to.

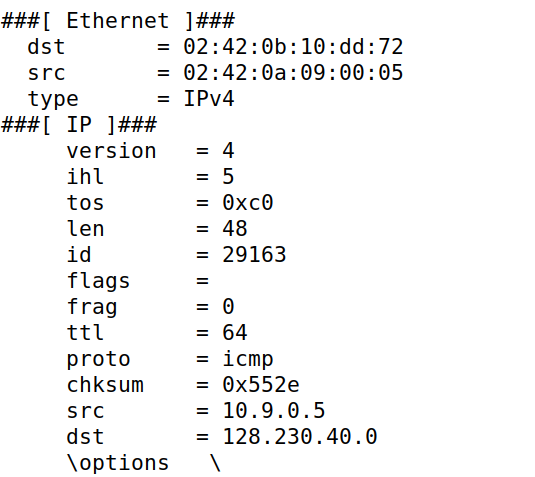
设置过滤器为filter='net 128.230.0.0/16'

能捕获来自或转到128.230.0.0/16这个子网的数据包

构造发包程序

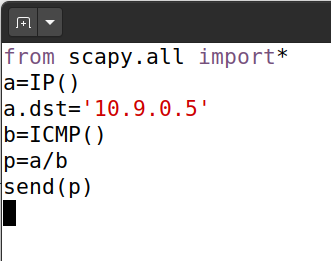


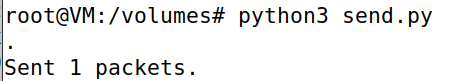
运行sniffer.py的结果



**Task 1.2: Spoofifing ICMP Packets**

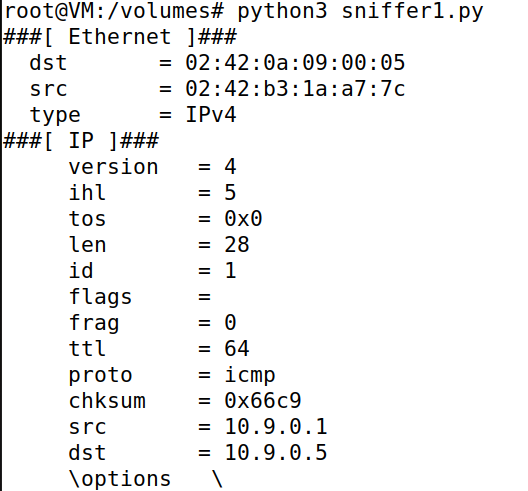
写发送数据包的程序sned.py并运行

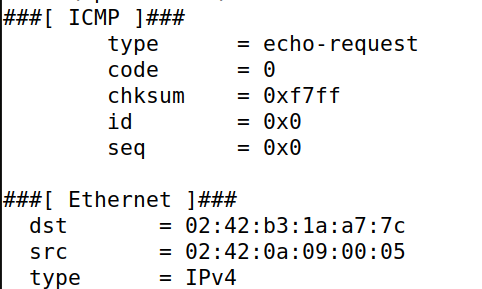


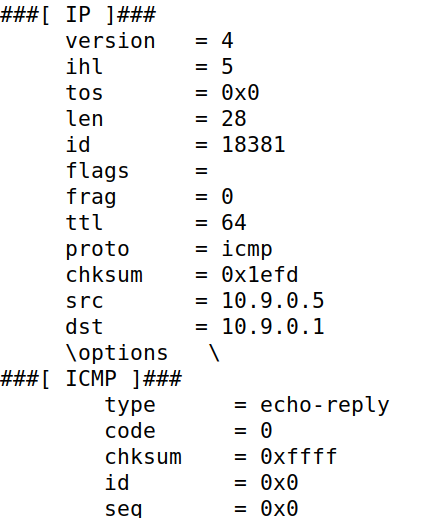


此时发送一个ICMP数据包到10.9.0.5

攻击者运行的嗅探程序能捕获到这个数据包

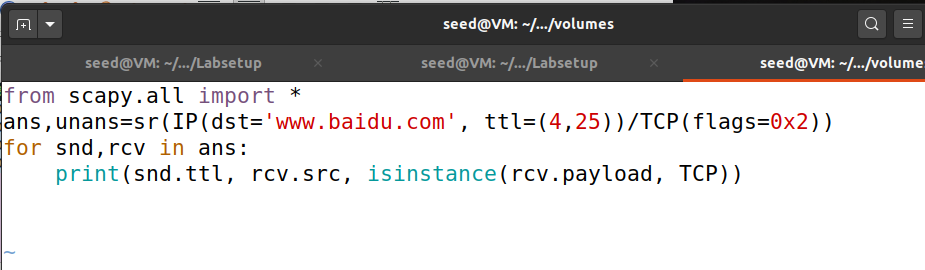




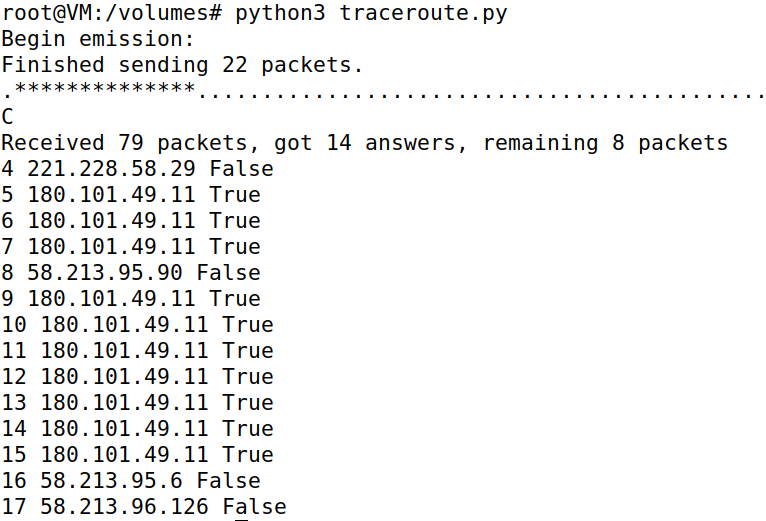


**Task 1.3: Traceroute**

写一个脚本来发送数据包

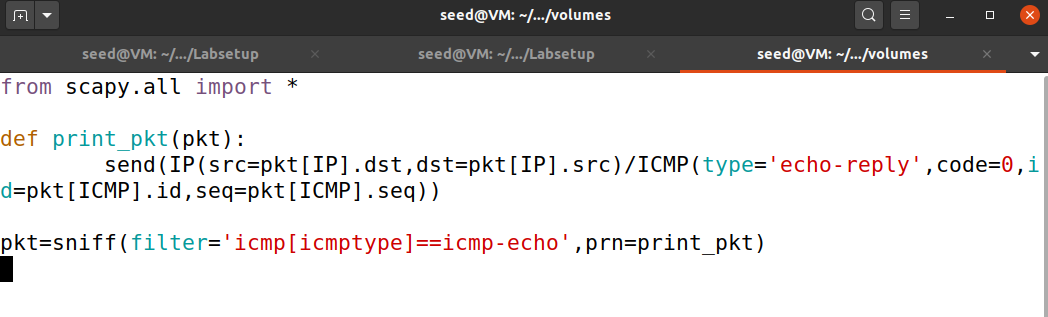


运行后得到的结果如下所示

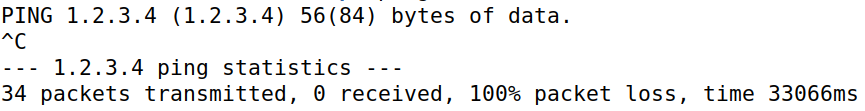


**Task 1.4: Sniffifing and-then Spoofifing**

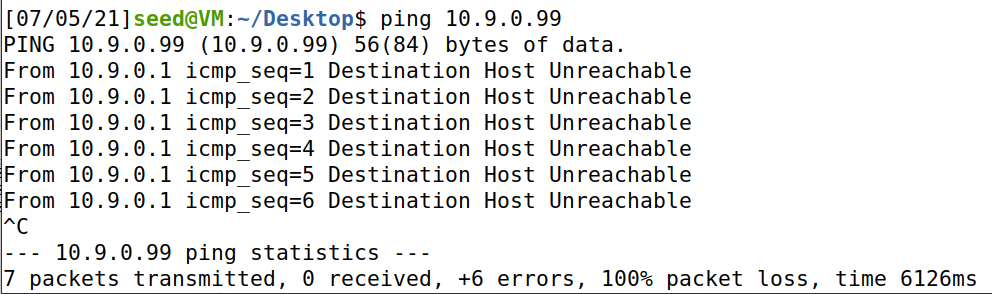
编写如下程序



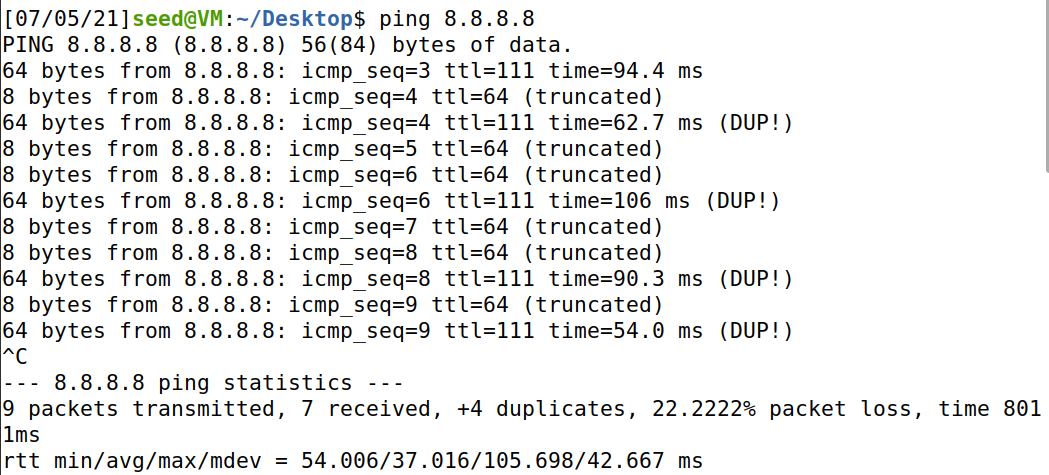
在运行程序的情况下用主机依次ping 1.2.3.4；10.9.0.99；8.8.8.8三个ip得到的结果如下：



无效ip无法连接



局域网内不存在的ip地址无法通过路由转发



一个正常的地址