**Packet Sniffing and Spoofing Lab**

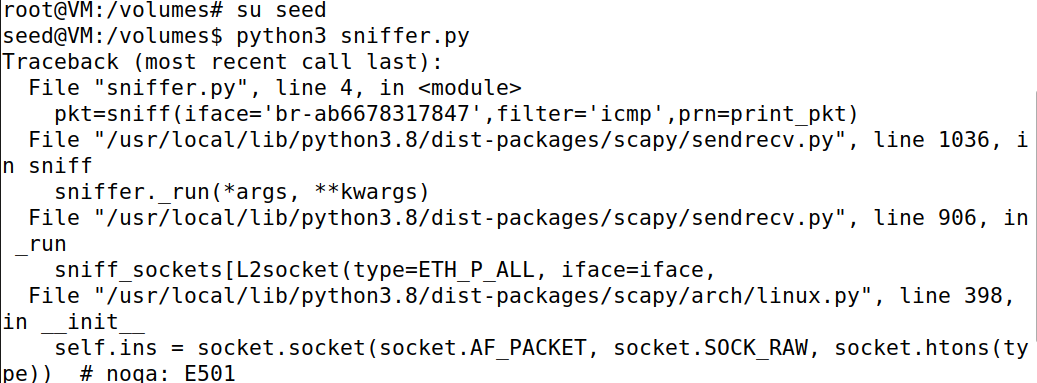
**姓名：齐天一 学号：57118223**

**Task 1.1**

**Task1.1A**

1. 非root权限下运行

输入以下指令，在seed用户下运行sniffer.py，发现程序报错



根据信息我们知道，非root情况下无法运行的原因在于无法创建socket

1. root权限下运行

切换到root权限并执行  


**Task1.1B**

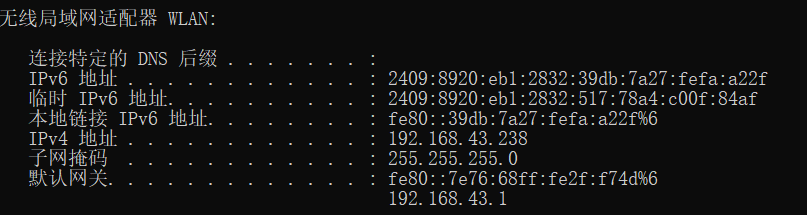
1. 只查看ICMP报文

和1.1A的程序一样，都是令filter等于'icmp'，运行结果与之前一致。

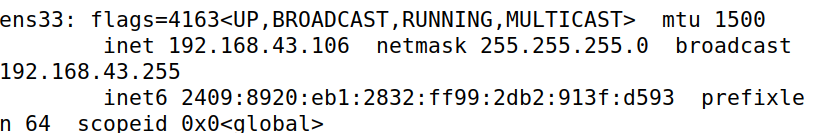
1. 查看特定ip发出，由23端口接受的报文

为了发送由23端口报文，选择在宿主机上向虚拟机发起telnet请求，即可产生特定ip到23端口的报文

首先查看自己电脑的ip地址（192.168.43.238）：



再查看虚拟机ip地址（192.168.43.106)，并记下端口名(ens33)：



修改sniffer程序:

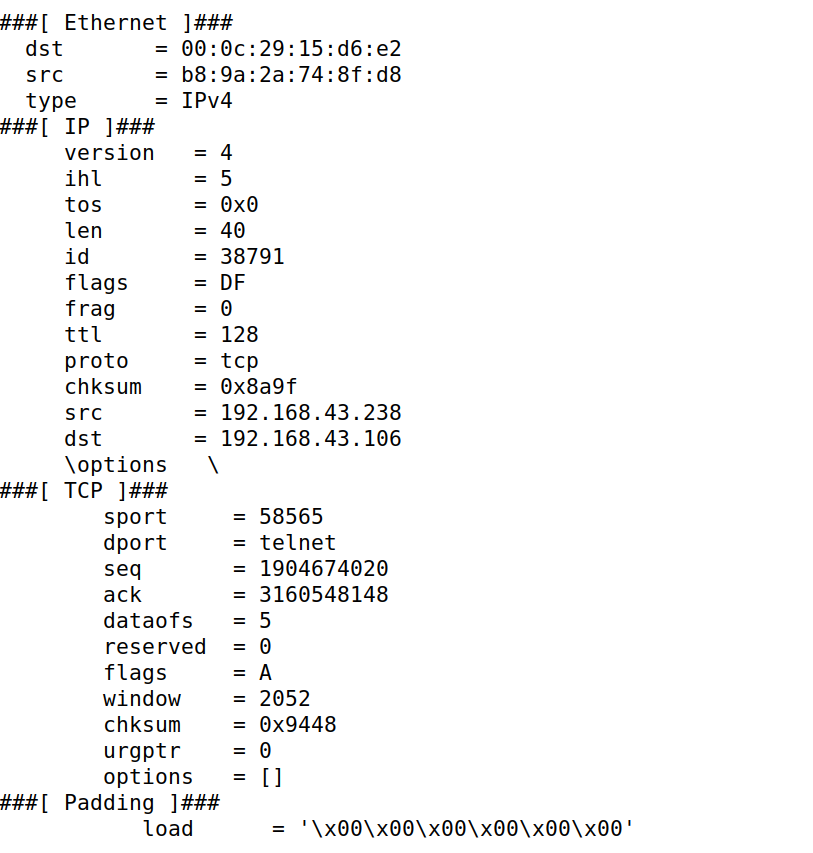
包括①修改ipace为‘ens33’

②修改filter为‘src host 192.168.43.106 and tcp dst port 23’

程序代码如下：



程序运行结果：

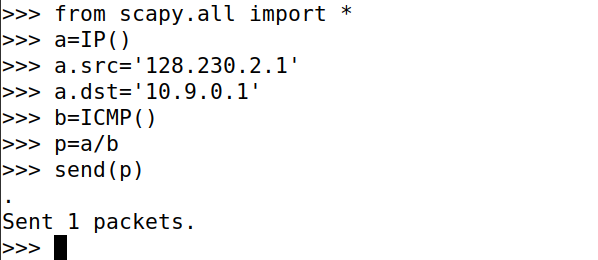


可以看到成功捕获了。

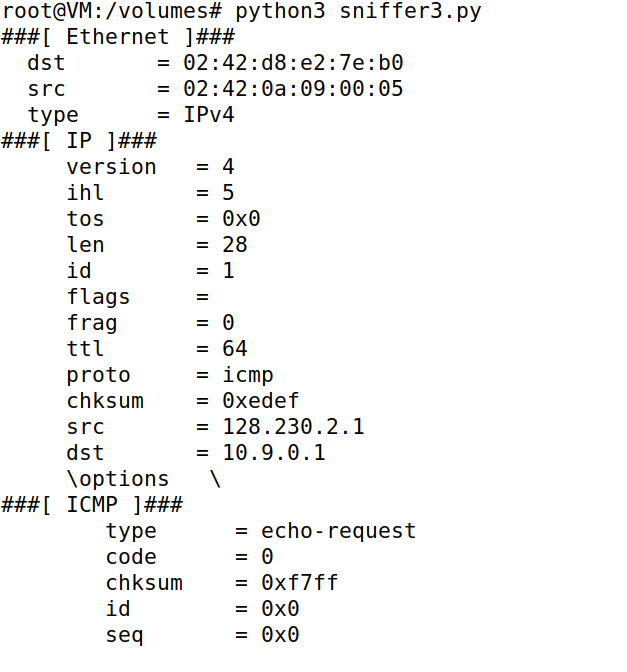
1. 查看某个子网的报文

为了产生子网的报文，我们可以使用Task1.2中的伪装技术，在用户容器中产生源地址在128.230.0.0/16网络中的报文，并发送给攻击者，攻击者运行程序，从而模拟嗅探过程。

代码如下（其中用src net来表示来源的网络，其值为128.230，表示子网掩码是255.255.0.0）：



攻击者嗅探程序的运行结果：



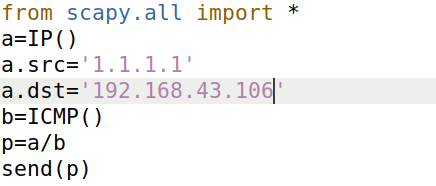
可以看到，成功收到了来自128.230.2.1的报文。

---------------------------------------------------------------------

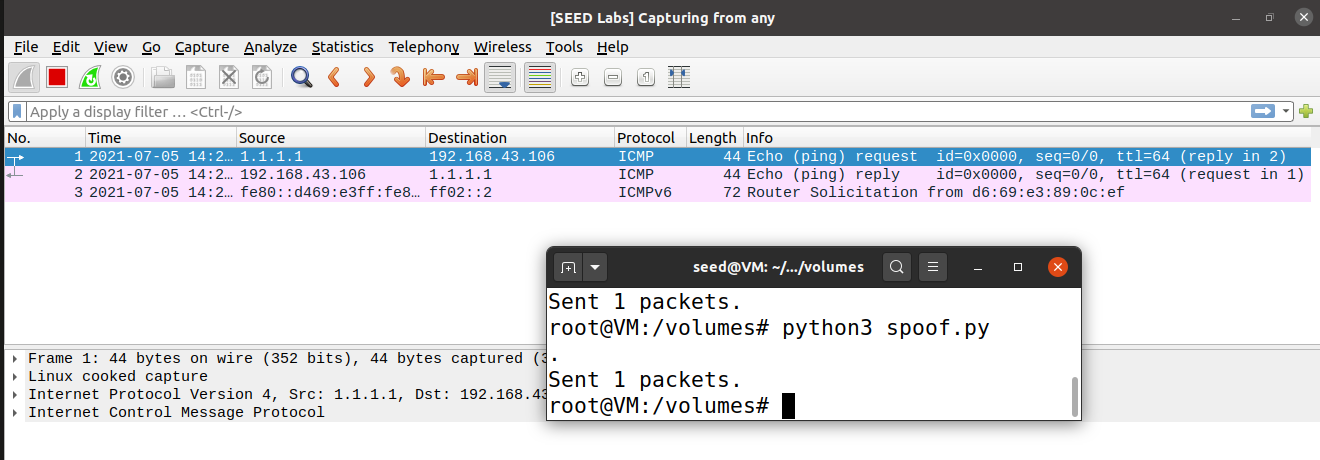
**Task1.2**

创建IP对象，将其源地址设为要伪造的ip地址，目标地址设为要发送的主机ip。这里将伪造ip设为1.1.1.1，目标ip地址为虚拟机地址。将其负载到icmp中对象中并发送。

程序如下：



Wireshark检测192.168.43.106端口，发现了来自1.1.1.1的报文，说明伪造成功。



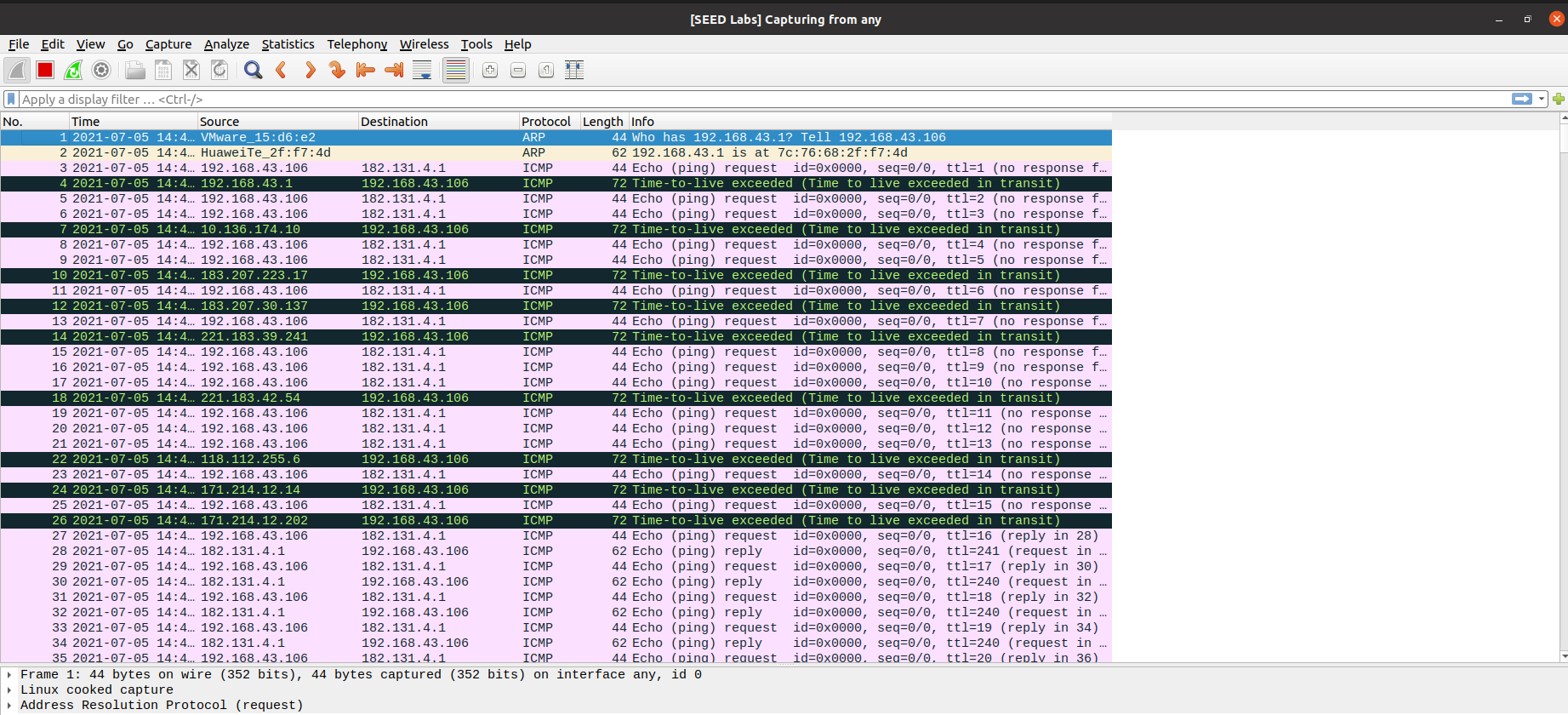
---------------------------------------------------------------------

**Task1.3**

编写程序如下，不断增加ttl值，构造报文并发送：



京东公司ip：



可以看到在ttl为16的时候收到了reply报文，说明路径上的跳数大约为16，其中有9个路由器发送了icmp错误报文。

---------------------------------------------------------------------

**Task1.4**

该任务的目的是：对于正常用户发送的ping，无论ip地址是否存在都返回一个回应的报文。

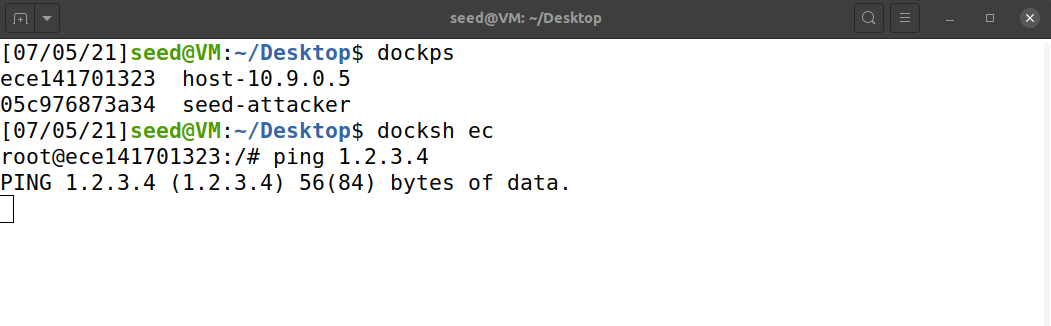
实现方法：对于IP段，交换源地址和宿地址作为回应报文的IP；对于ICMP类型，设为reply类型（0）；对于其它部分，复制用户报文的内容放到相应位置。

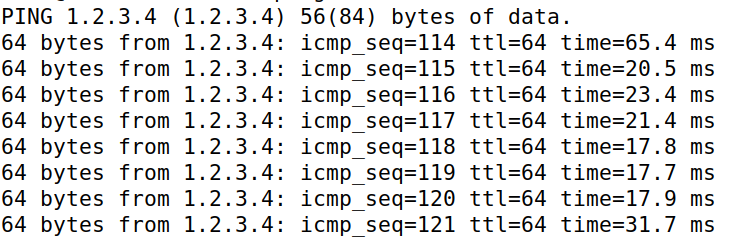
代码如下：



1. 尝试ping 1.2.3.4（网络中不存在的ip）:

攻击者未开启程序:

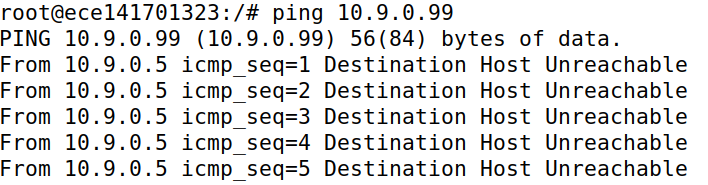
攻击者开启程序：



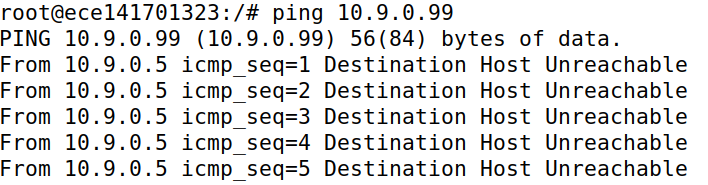
开启伪造程序后，用户成功收到了相应报文，说明伪造成功！

1. 尝试ping （局域网中不存在的ip）

攻击者未开启：



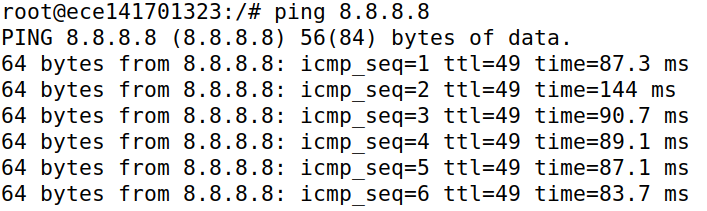
开启程序：



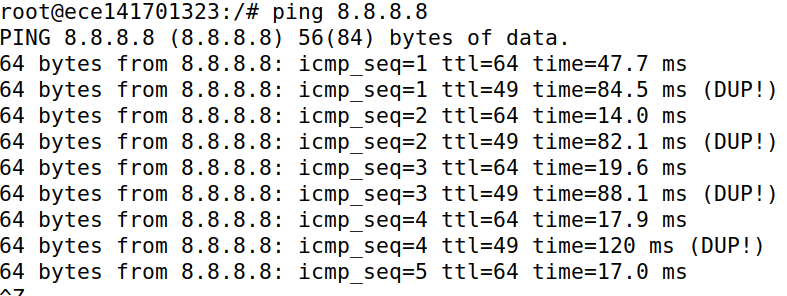
造成这种现象的原因是局域网中没有配置该地址的路由。

（3）尝试ping 8.8.8.8（网络中存在的ip）

攻击者未开启程序：



开启程序：



出现DUP的原因是因为真实的8.8.8.8主机会发送icmp回应报文，而攻击者也会发送一份，这就导致用户会重复受到相同报文，从而提示DUP信息。