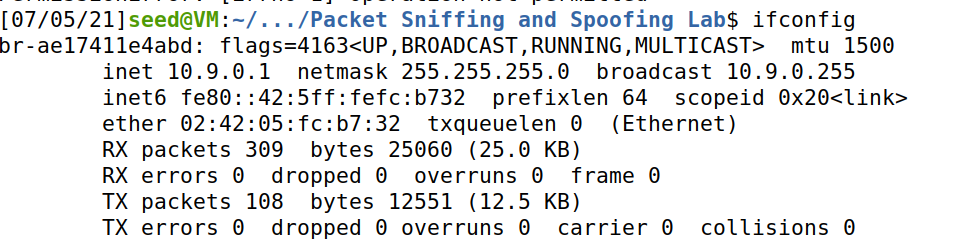
**Lab 1**

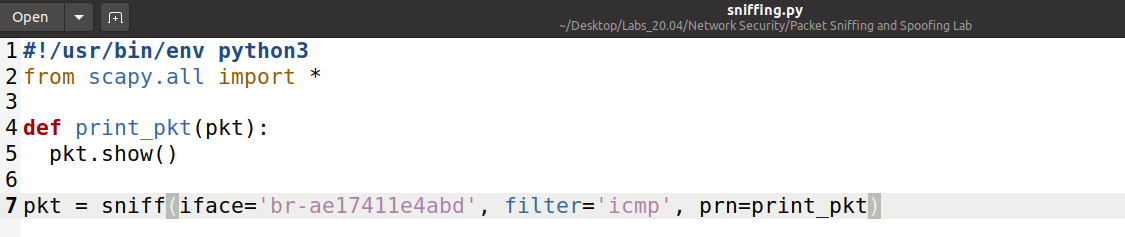
57118201邓彤

**1.1A**

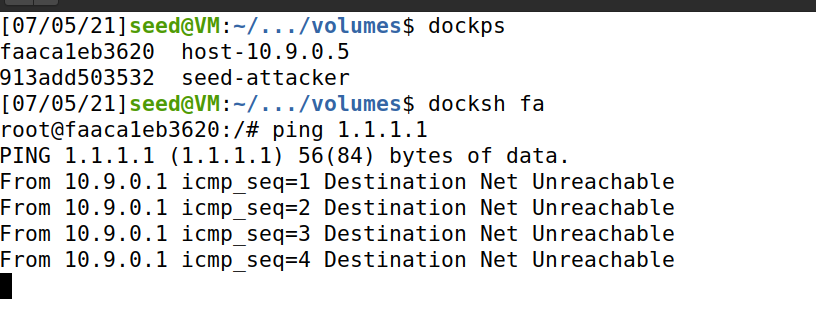
**ifcondig命令下显示的网卡描述为br-****ae17411e4abd**

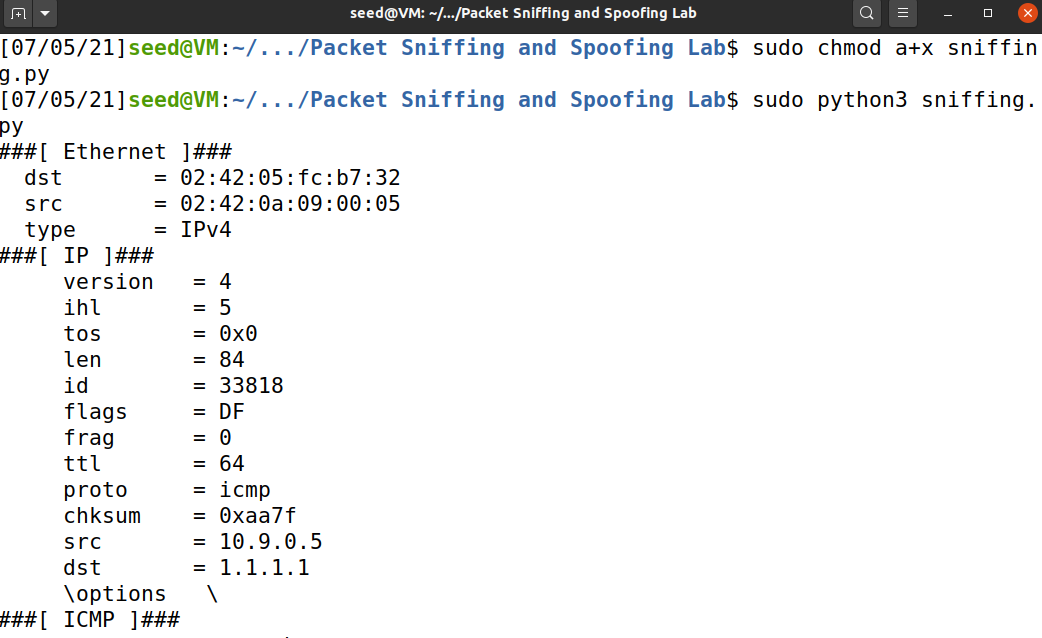


**进入 Lab目录下， 新建 sniffing.py 文件，代码如下**

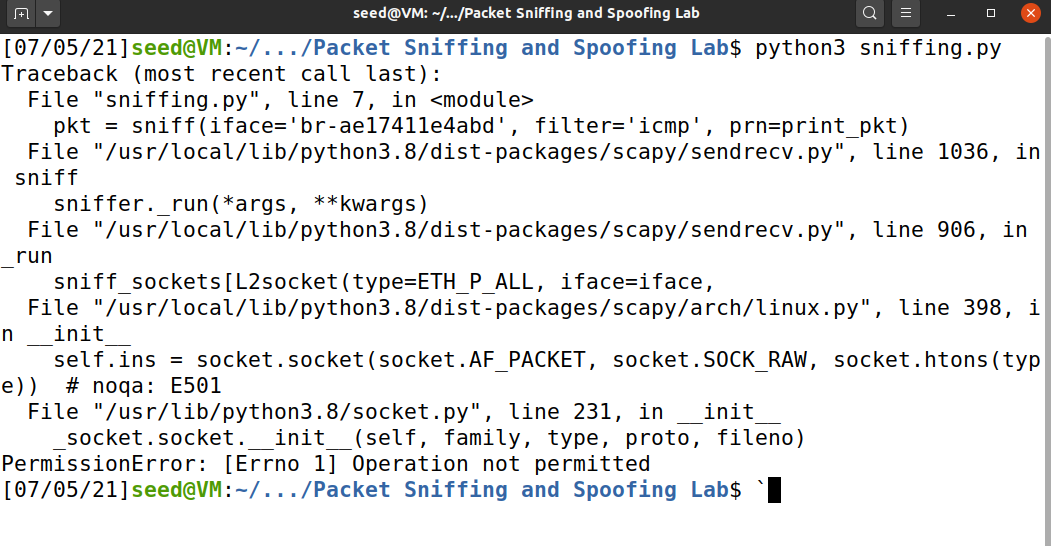


**提权之后，使用root权限，显示如下**





**未使用root权限时，显示如下：**



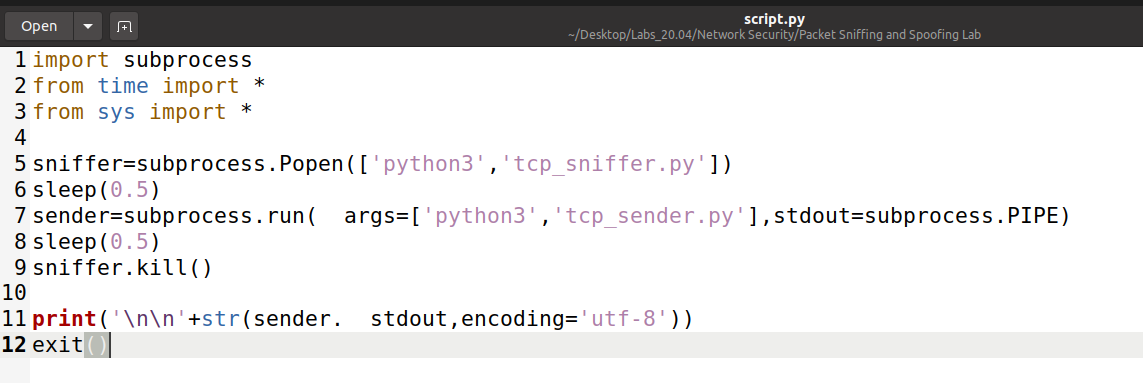
**1.1B**

* **第一问 捕获 ICMP 报文**

过程与1.1A相同

* **第二问 捕获来自特定 IP、目的端口 23 的数据包**

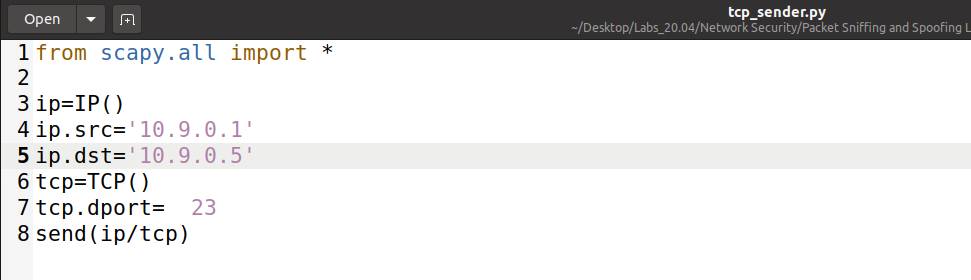
**script.py**代码如下：



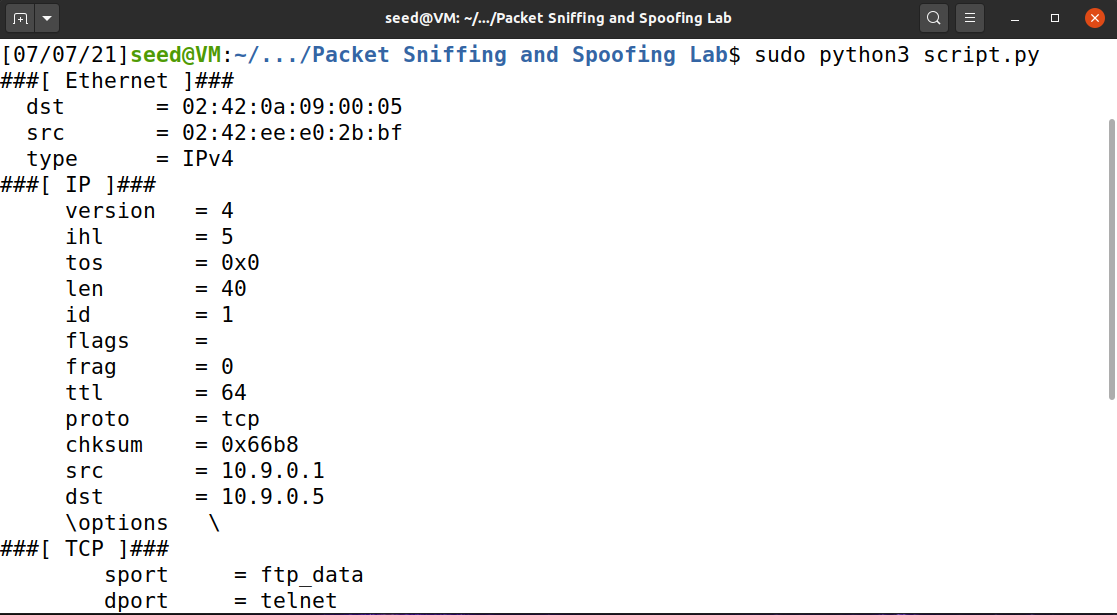
**tcp\_sniffer.py**代码如下：



**tcp\_sender.py**代码如下：

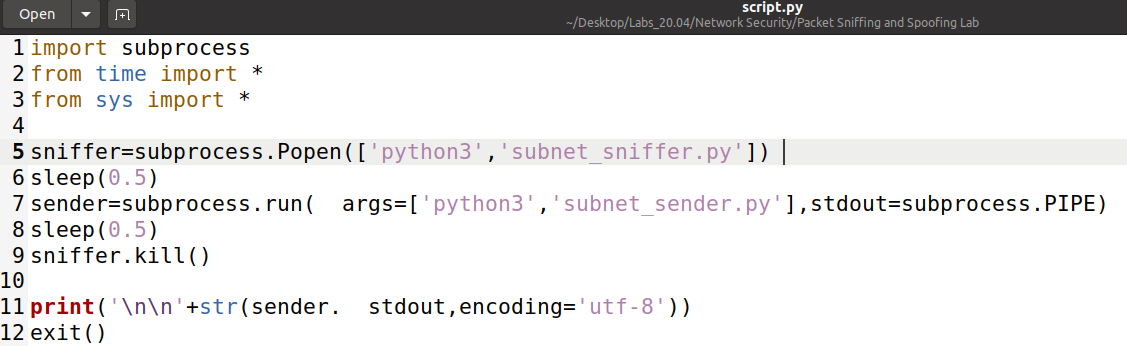


**程序监听到的报文如下：**

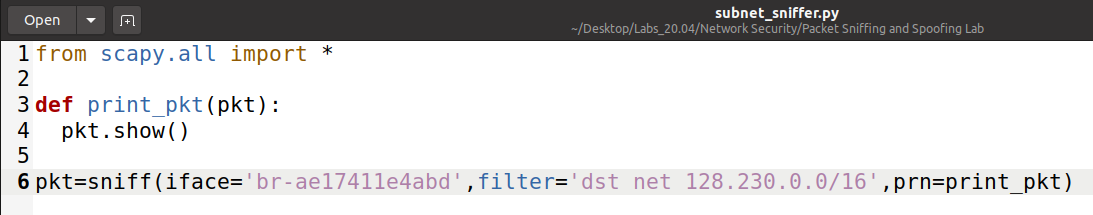


* **第三问 来自特定子网的报文**

**script.py**代码如下：



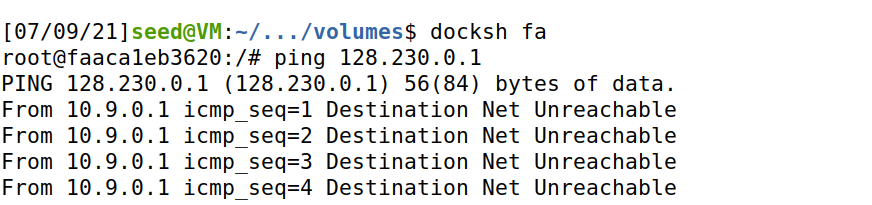
**subnet\_sniffer.py**代码如下：



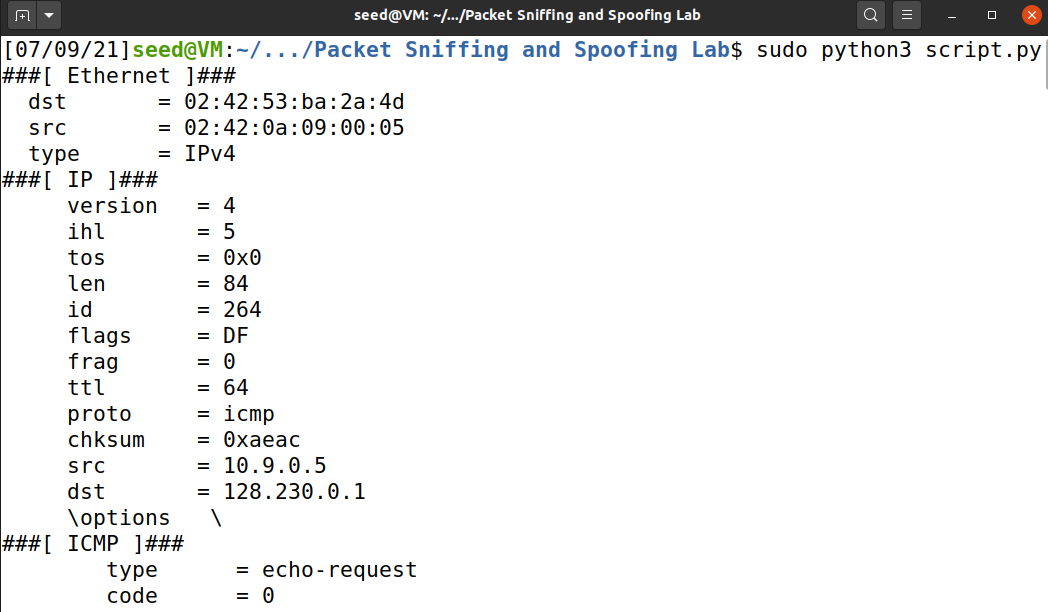
**subnet\_sender.py**代码如下：



**登录10.9.0.5 ping 128.230.0.1**



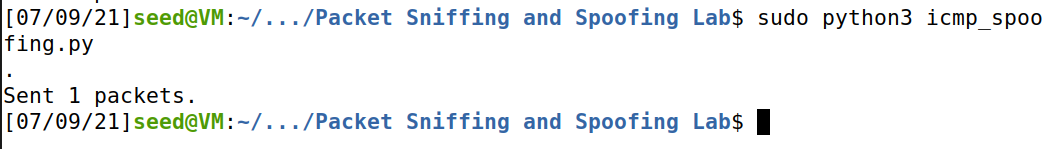
**程序监听到的报文如下：**



**1.2 构造ICMP欺骗数据包**



**执行程序发送一个数据包**

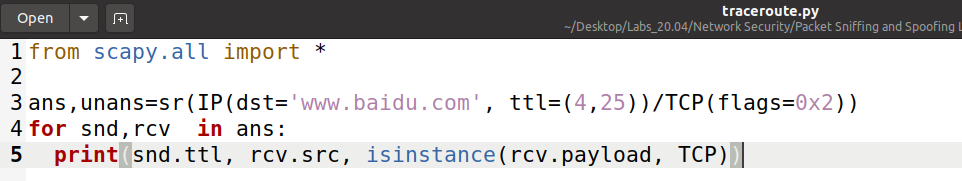


**Wireshark 捕获到的来回的数据包显示如下：**

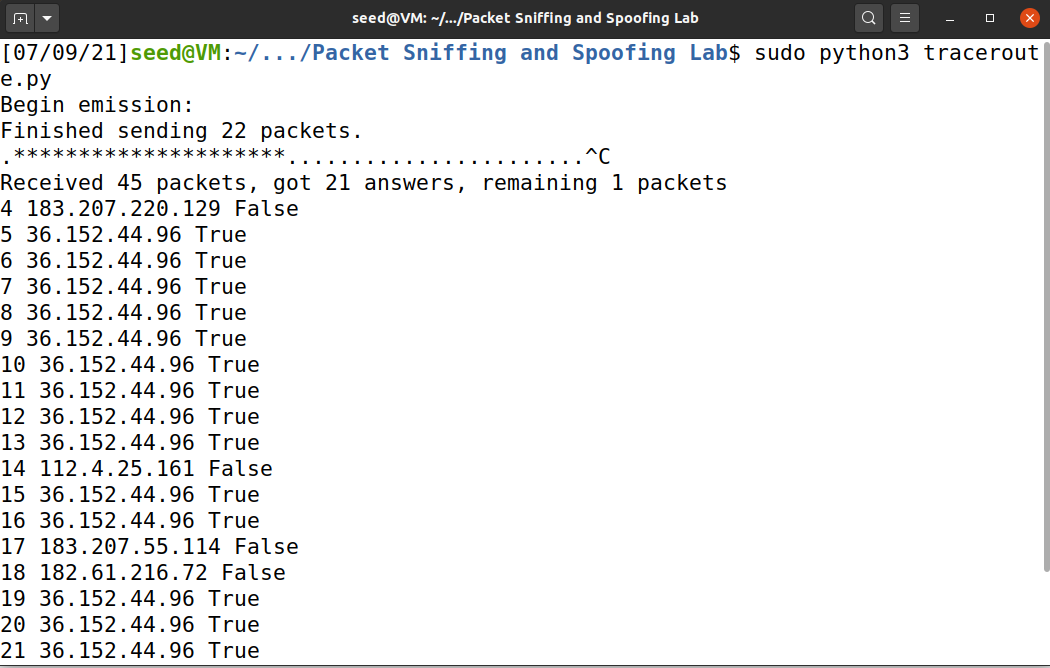
C:\Users\dengtong\AppData\Roaming\Tencent\Users\1977988055\QQ\WinTemp\RichOle\}IDU~1YJ7{]O6XRAHT4BW%R.png

**1.3 跟踪路由**

**编写traceroute.py**



**其运行结果如下**



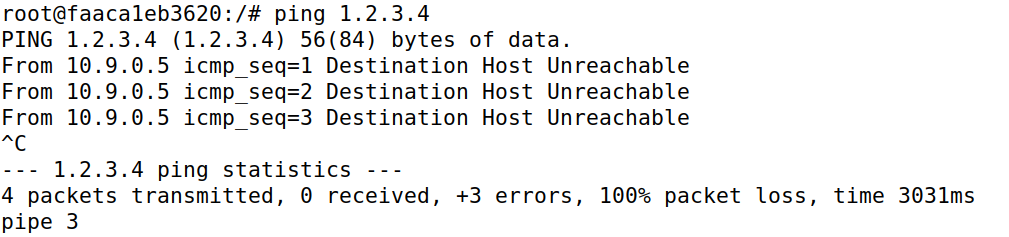
**1.4 嗅探然后欺骗**

**编写的sniff\_spoof.py如下**

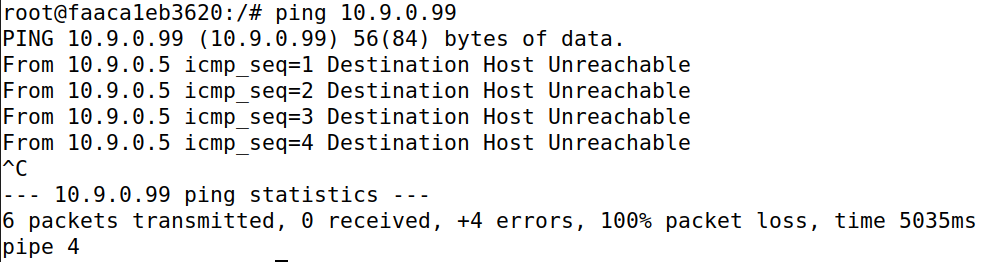


**运行程序前，分别ping三个地址**

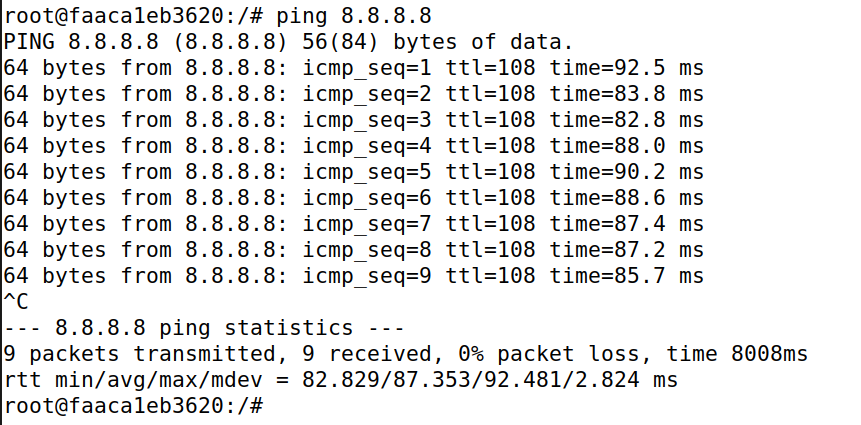
**ping 1.2.3.4：**



**ping 10.9.0.99：**



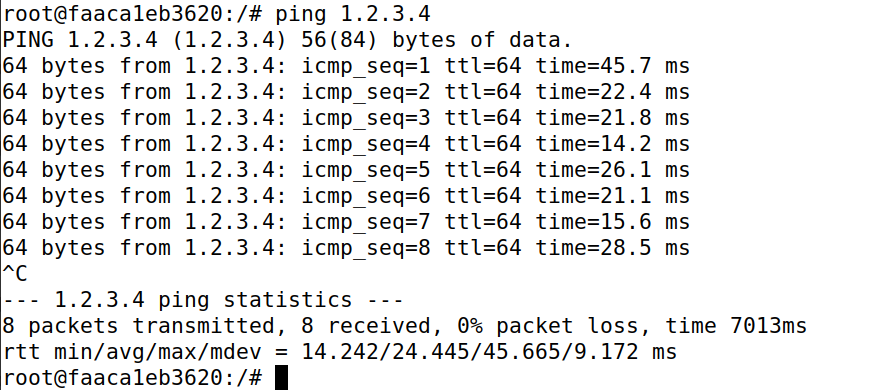
**ping 8.8.8.8：**



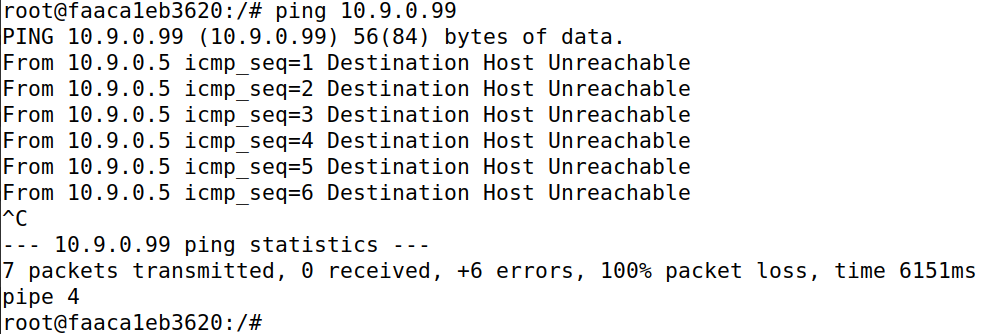
可以发现，在运行 spiff\_spoof.py 前，无法 ping 通 1.2.3.4 和 10.9.0.99，但能 ping 通 8.8.8.8

**运行程序后，再ping**

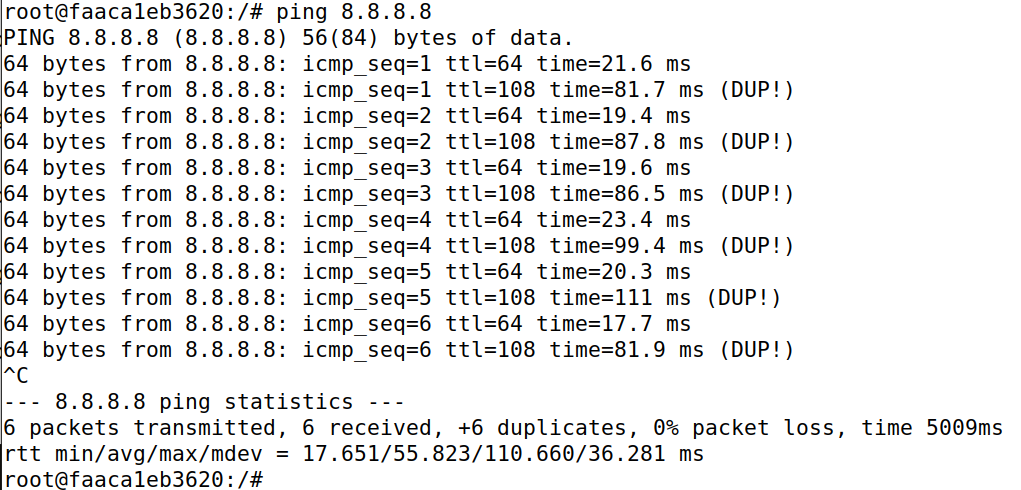
**Ping 1.2.3.4：**



**Ping 10.9.0.99：**



**Ping 8.8.8.8:**



**结论：**可以发现，在运行 sniff\_spoof.py 之后， 10.9.0.99 仍然无法 ping 通，但1.2.3.4 能 ping 通。因为在运行程序之前，网关 10.9.0.5 告知主机无法通过 ARP 协议找到1.2.3.4 和 10.0.9.99 对应的 MAC 地址，因此无法 ping 通；而 8.8.8.8 在互联网上存在，因此可以 ping 通。

在运行程序之后， ping 1.2.3.4 需要经过网关 10.9.0.5，网关拦截 ICMP 报文并欺骗主机可以 ping 通 1.2.3.4。而 10.9.0.99 和主机在同一个局域网内，通过广播 ARP 寻找相应的 MAC 地址，不需要经过网关，因此网关无法欺骗主机，10.9.0.99 仍然 ping 不通。