

网络安全实验报告

题 目  捕包软件的使用与实现

专 业 信息安全

学 号 1170301027

学 生 冯帅

指 导 教 师 王彦

**一、实验目的**

理解捕包程序捕包过程，可以自己编程捕包并从数据包中解析出需要的信息。

# 二、实验内容

# 1.winpcapt或libpcap的安装

# 2.winpcapt或libpcap编程过程，编写程序捕获实验一的数据包。还原四元组（源目的IP源目的端口），和文件内容，写到一个文件里（默认文件），如果需要也可输出到特定文件（通过配置文件或命令行参数决定，屏幕也是特定文件之一）

# 三、实验过程

**（一）使用 wireshark 软件对协议进行还原**

## 实验基本信息：

实验环境：Windows10 x64

Ubuntu 18.04

## 1. 需求分析

本程序需要运用 libpcap 来捕获本机数据包，并获取数据包中的四元组，将其展示给

用户。

程序功能：

(1)捕获本机数据包（可以自定义过滤条件）；

(2)逐层解析数据包，获得 IPv4 数据包的源 ip、目的 ip、源端口、目的端口；以及解析链路层mac地址

(3)将上述四元组写入文件，文件名由程序运行时屏幕（命令行）录入。

注：实验一要求将特定格式的数据报发送出去，我定义的格式是数据中包含四元组以及要发送的数据，拆包时不是将解析出来的ip头的数据信息写入文件，而是从数据报中提取出来的。详见测试结果

## 2. 环境配置

(1)从官网下载安装包：tcpdump-4.1.1.tar.gz

(2)对以上安装包逐个使用以下命令来解压安装包：

tar -zxvf xxx.tar.gz

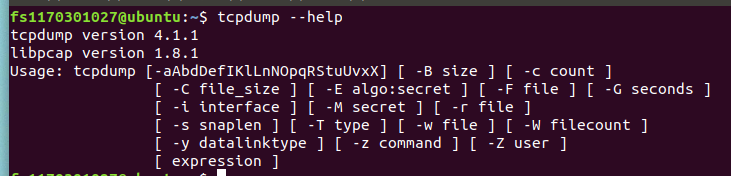
(3)进入解压后的文件夹后执行以下命令来安装库文件：

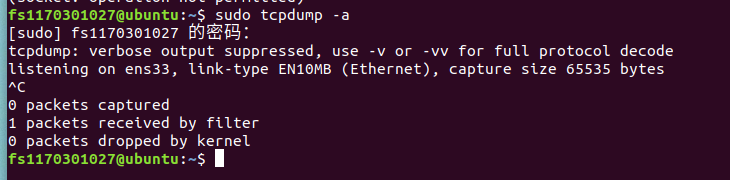
./configure

make

make install

(4)配置好环境后，可正常编译，运行



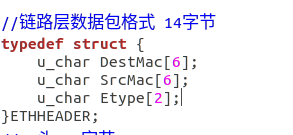


## 3. 数据结构设计

由于是逐层解析以太网数据帧，所以需要准备至少三种数据结构：以太网数据帧头、

IPv4 数据报头、传输层报文头。具体如下：

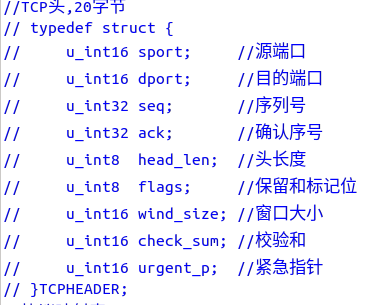
1. 以太网数据帧头



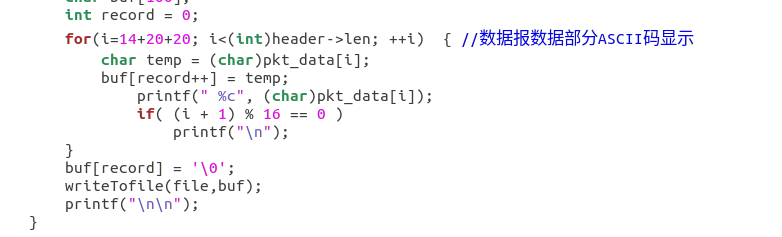
1. IPv4 数据报头



1. TCP 报文头



## 4. 回调函数设计

回调函数先构造链路帧头和ip头，提取出链路层mac地址和网络层IP地址并显示，然后根据数据帧头部、ip数据报头部、tcp头部大小的索引（14+20+20），输出数据部分的十六进制表示以及对应的ASCII码值，最终将数据写入文件，文件名由屏幕录入，默认为record.txt

## 5. 主函数设计

第一步：使用 pcap\_lookupdev()函数来获取网络设备。 否则为默认网卡，我的虚拟机下查看的默认网卡是ens33

第二步：使用 pcap\_open\_live()函数来获得捕包描述字，由于只需要捕获本机数据包所以设置为非混杂模式。

第三步：使用pcap\_lookupnet()获取网络号和子网掩码，屏幕输出显示

第四步：如果有过滤条件的话（作为程序运行参数读入），pcap\_setfilter()设置网络过滤器。

第五步： 使用pcap\_datalink()获取链路层适配器

第六步：使用 pcap\_loop()函数和回调函数 callback()来循环捕包。

## 6. 编译运行

使用下列命令编译：

gcc capture.c -o capture -lpcap

-lpcap 用来链接 pcap 库。

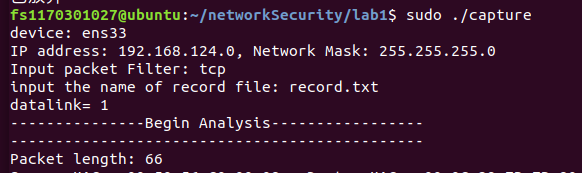
使用下列命令运行：

**sudo** ./capture

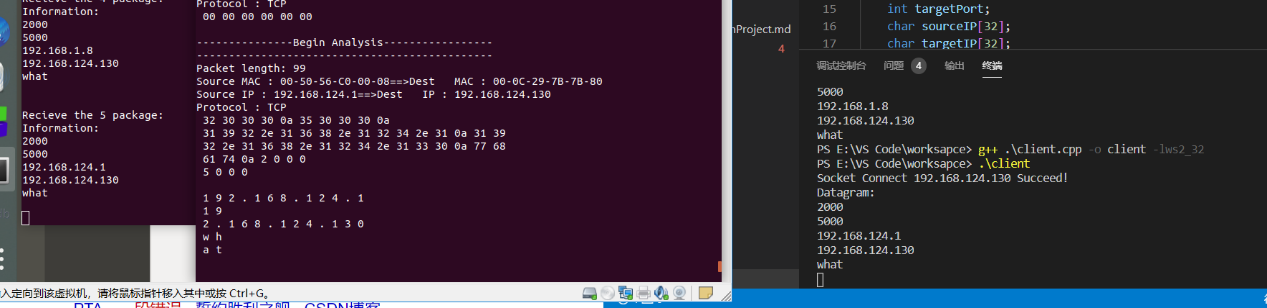
捕包程序在运行时需要 root 权限，否则无法正常打开。

# 四、实验结果

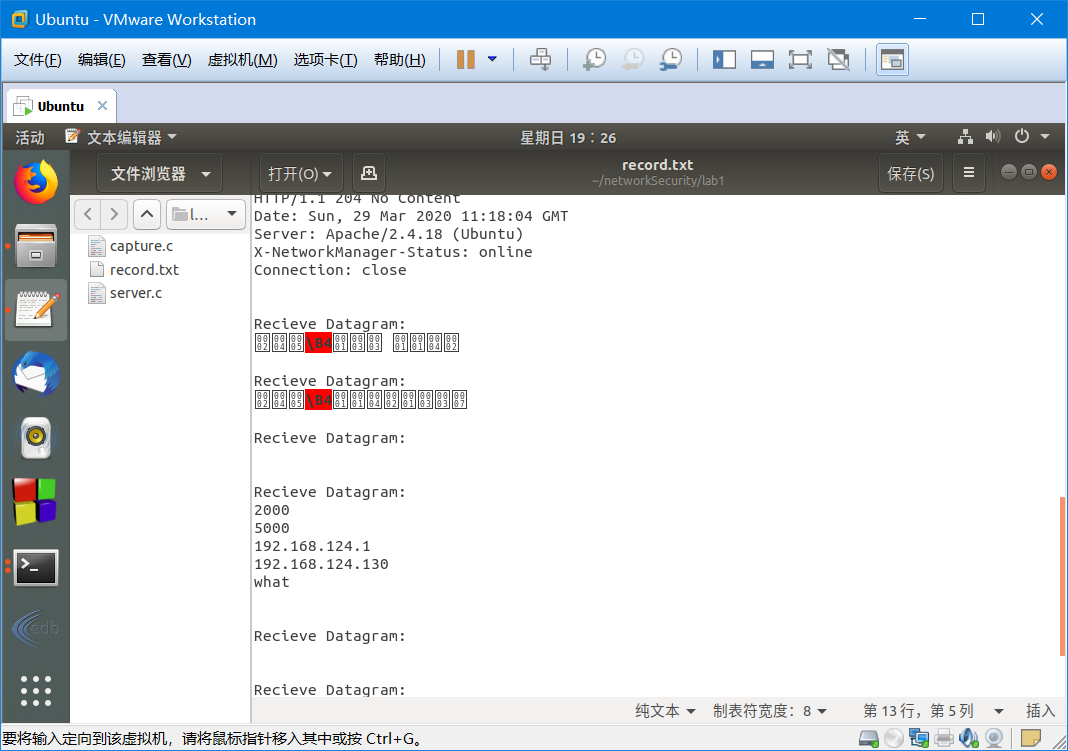
**利用 libpcap 编写捕包软件**



程序运行：需要录入过滤条件和输出文件



部Lab1中的数据报捕获



捕包时文件中部分输出

# 心得体会

体验了一下自己编写的捕包程序，虽然多是参考，但是毕竟动手走了一遍这个流程，感觉收获颇丰，比之直接从wireshark上面看现成的分析结果要好的多，理解也深刻一些。

# 附件

# 附件capture.c为捕包程序，record.txt为输出文件，client.c为实验一win下客户端文件，server.c为实验一linux下服务器端文件