

网络安全实验报告

题 目 基于 libnet 的程序设计

专 业 计算机科学与技术

学 号 7203610316

学 生 符兴

指 导 教 师 王彦

# 一、实验目的

掌握 libnet 数据包的构造原理。

1. **实验内容**

1.掌握 libnet 数据包的构造原理

2.编程实现基于 libnet 的数据包构造，结合前面实验给出验证过程。能够对源码进行解释。

# 三、实验过程

**基于 libnet 的数据包构造**

**实验基本信息：**

实验环境：Ubuntu 20.04 x64

编程语言：C 语言

## 1. 需求分析

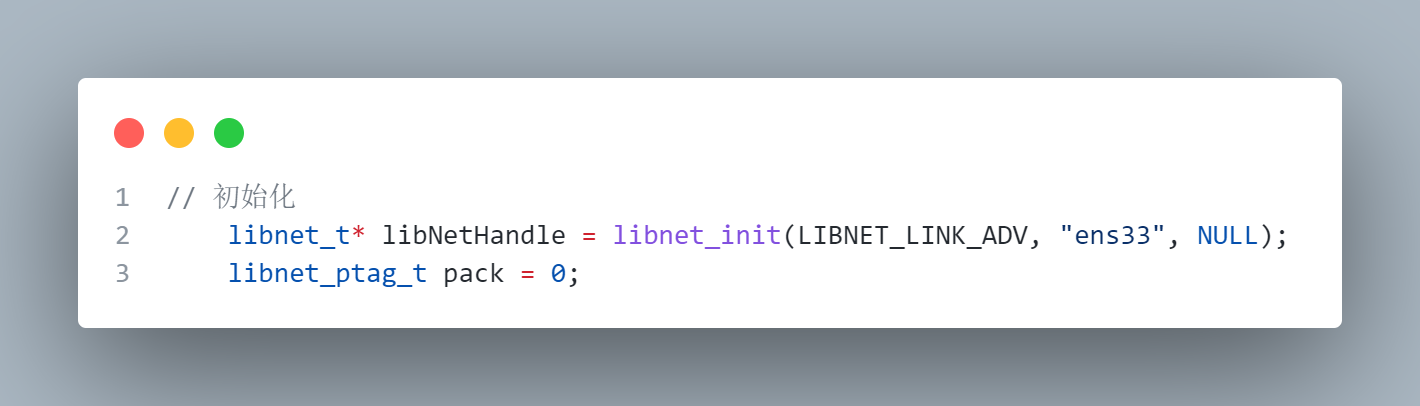
需要使用 libnet 构造并发送一个数据包，并验证这个数据包被成功发送了。验证这一过程需要用到实验二中的捕包程序 pcap，将生成的数据包从虚拟机 B 发送到虚拟机 A，虚拟机 A 中的捕包程序会自动将其捕获，通过检查各项信息，证明捕获的数据包就是从虚拟机 B 此程序 mackPack 中发送的数据包。

追加：在虚拟机 A 中编写接收来自相应端口 udp 数据包的程序 recvUDP，验证该数据包确实可以被正确接收。

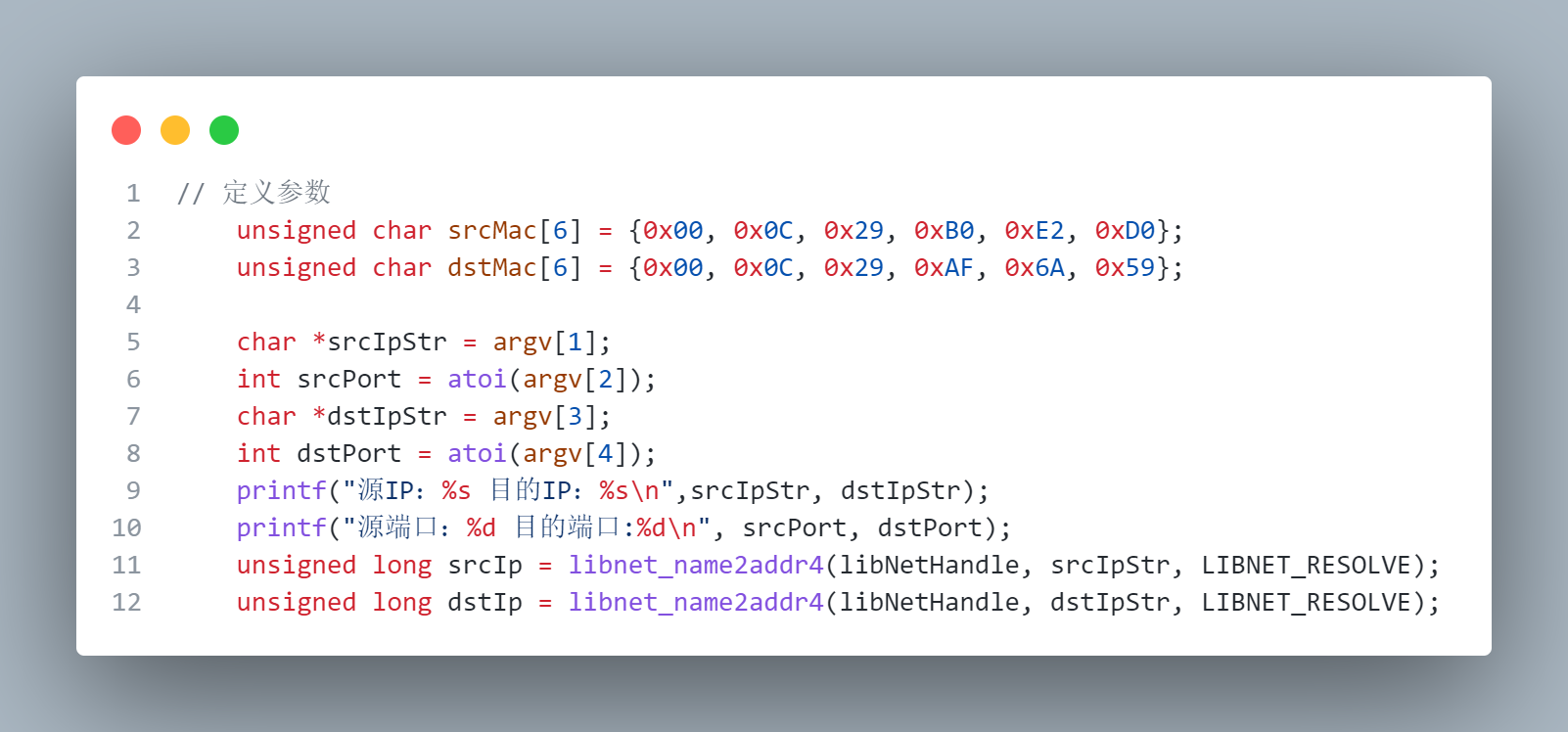
## 2. 程序结构

(1) makePack程序的大致结构为：

① 初始化libNet句柄，其中需要设置好网卡接口。



② 定义源、目的主机的MAC地址以及源、目的主机的IP地址和端口；其中从参数接收的IP地址字符串需要调用libnet中的函数将其转为网络字节序。



③ 设置UDP需要发送的数据。

④ 依照顺序依次构建数据包：UDP数据包->IP数据报->以太网帧。

|  |
| --- |
|  |
| 图 构建UDP数据包 |

|  |
| --- |
|  |
| 图 构建IP数据报 |

|  |
| --- |
|  |
| 图 构建以太网帧 |

⑤ 发送数据

(2) recvUDP程序结构：

① 创建socket句柄

② bind连接

③ recvfrom()监听相应端口接收的UDP数据包。

## 3. 进一步验证

(1) 设置makePack中UDP所发送的数据为:” Data From B.”；同时，在recvUDP中使用strcmp()函数对接收到的数据进行判断。

|  |
| --- |
|  |
| 图 recvUDP对数据进行判断 |

(2) 同时在目标主机上运行实验二所构建的pcap程序对UDP数据包进行捕获，查看其中的源、目的IP以及源、目的端口判断makePack程序所构建的数据包是否正确，并是否正确发送到目标主机。

# 四、实验结果

|  |
| --- |
|  |
| 图 recvUDP检验结果 |

在实验过程中，主机A运行makePack程序，其中源IP地址为192.168.142.128，源端口为6687；目的IP地址为192.168.142.129，目的端口为6689；

从上图的实验结果中可以看到，makePack程序已正确发送数据包，并且主机B上的recvUDP程序也正确接收到该数据。

|  |
| --- |
|  |
| 图 pcap捕获结果 |

从上图可以看到，主机B上的pcap程序捕获到该数据包，且正确解析该数据包的参数；

# 五、心得体会

(1) 掌握基于libnet开发流程，并正确依次构建了UDP数据包、IP数据报以及以太网帧。

(2) 更深入了解了网络中两台主机之间发送数据的具体过程，以及在各个网络层次结构中所进行的具体操作；