

**网络技术与应用课程实验报告**

**实验三：编程获取IP地址和Mac地址对应关系**

****

专 业 信息安全

学 号 2113662

姓 名 张丛

班 级 信息安全一班

1. **实验目的**

（1）在IP数据报捕获与分析编程实验的基础上，学习NPcap的数据包发送方法。

（2）通过NPcap编程，获取IP地址与MAC地址的映射关系。

（3）程序要具有输入IP地址，显示输入IP地址与获取的MAC地址对应关系界面。界面可以是命令行界面，也可以是图形界面，但应以简单明了的方式在屏幕上显示。

（4）编写的程序应结构清晰，具有较好的可读性。

1. **实验过程**

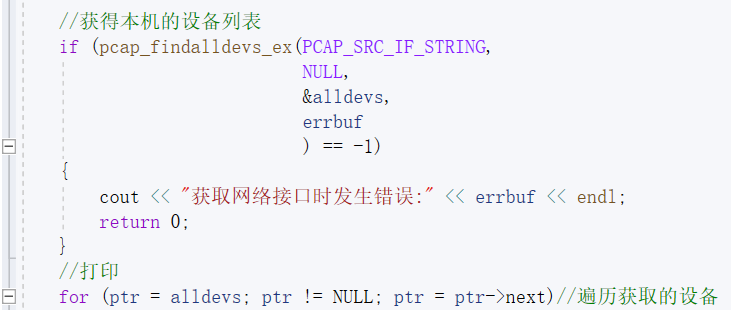
获取IP地址和Mac地址对应关系的流程：

* 获取设备列表
* 打开网卡
* 组装ARP报文
* 获取本机网卡的Mac地址和IP地址关系
* 获取远程主机的Mac地址和IP地址关系

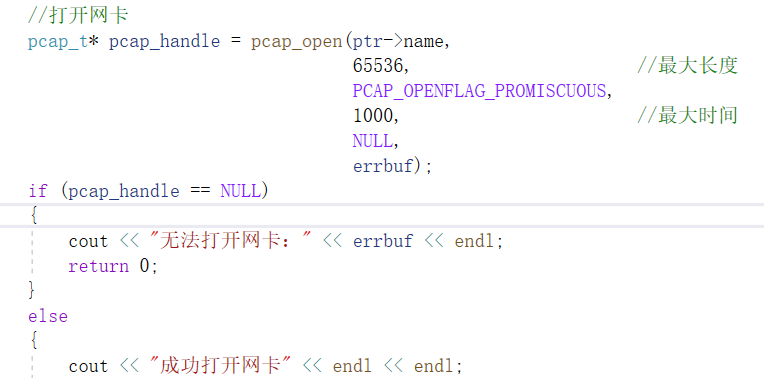
#### 获取及打开设备

对于获取和打开设备的部分，本次实验和上一次捕获数据包的使用是一样的，只是多了一点打印信息，故不再赘述。

主要还是使用pacp\_findalldevs\_ex()函数来获取设备：



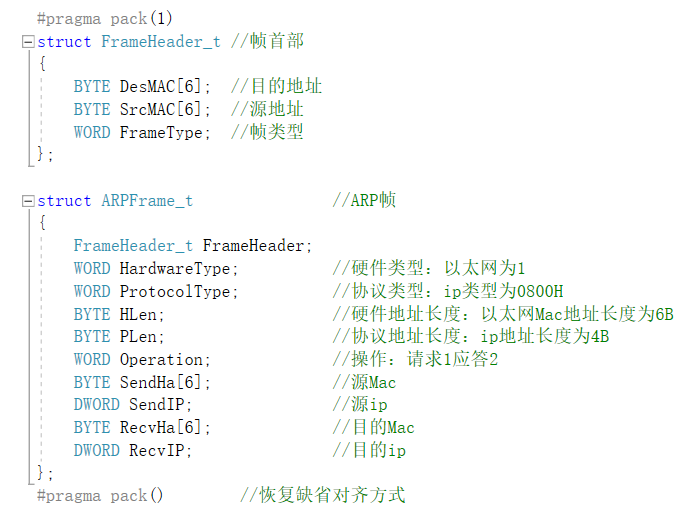
使用pacp\_open()函数来打开网卡：



#### 组装ARP报文

在上一次实验中，我们打开网卡后，利用pacp\_loop()函数捕获数据包，用回调函数 packet\_callback() 来处理数据包。

在这次实验中，我们要获取IP地址和Mac地址的对应关系，就需要捕获ARP响应报文；要获得ARP响应，我们需要先发送ARP请求。

ARP报文结构定义如下：  


在获取本机ip和mac对应关系以及远程主机ip和mac对应关系前，都需要组装相应的ARP报文。

#### 获取本机网卡的Mac地址和IP地址关系

获取本机网卡的IP地址和MAC地址关系，其实是将本地主机模拟成一个远端主机，发送一个ARP请求报文。

即本地主机IP地址作为ARP请求报文的目的IP地址（ARPFrame.RecvIP)，而请求报文的源ip（66-66-66-66-66-66）和源mac（112-112-112-112）则是虚假的。

本地主机ip可以由第一步获取设备列表时得到。

ARP请求报文如下：

1. ARPFrame\_t ARPFrame;
2. DWORD SendIP;
3. DWORD RevIP;
4. for (int i = 0; i < 6; i++)     *//Mac长度为6B*
5. {
6. ARPFrame.FrameHeader.DesMAC[i] = 0xFF;  *//广播：255.255.255.255.255.255*
7. ARPFrame.FrameHeader.SrcMAC[i] = 0x66;  *//虚假的源Mac：66-66-66-66-66-66-66*
8. ARPFrame.RecvHa[i] = 0;     *//目的Mac，先初始为为0*
9. ARPFrame.SendHa[i] = 0x66;    *//源Mac*
10. }
11. ARPFrame.FrameHeader.FrameType = htons(0x0806); *//帧类型为ARP*
12. ARPFrame.HardwareType = htons(0x0001);   *//硬件：以太网*
13. ARPFrame.ProtocolType = htons(0x0800);   *//协议：为IP*
14. ARPFrame.HLen = 6;
15. ARPFrame.PLen = 4;
16. ARPFrame.Operation = htons(0x0001);    *//操作：ARP请求*
17. SendIP = ARPFrame.SendIP = htonl(0x70707070); *//虚假的源ip地址：112.112.112.112.112.112*
18. *//所选网卡IP设为目的ip地址*
19. *//ptr已经迭代到所选网卡*
20. for (a = ptr->addresses; a != NULL; a = a->next)
21. {
22. if (a->addr->sa\_family == AF\_INET)*//判断该地址是否为ip地址*
23. {
24. RevIP = ARPFrame.RecvIP = inet\_addr(inet\_ntoa(((struct sockaddr\_in\*)(a->addr))->sin\_addr));
25. }
26. }

接下来发送ARP报文，再捕获所选网卡的数据包获得ARP响应报文即可。

响应报文的源ip是请求报文目的ip，响应报文的源mac是请求报文的目的mac，可以依次判断是否为响应报文。

发送以及捕获响应：

1. pcap\_sendpacket(pcap\_handle, (u\_char\*)&ARPFrame, sizeof(ARPFrame\_t));
2. *//捕获所选网卡数据包*
3. ARPFrame\_t\* IPPacket;
4. struct pcap\_pkthdr\* pkt\_header;
5. const u\_char\* pkt\_data;
6. while (true)
7. {
8. int rtn = pcap\_next\_ex(pcap\_handle, &pkt\_header, &pkt\_data);
9. if (rtn == -1)
10. {
11. cout << "  捕获数据包时发生错误：" << errbuf << endl;
12. return 0;
13. }
14. else if (rtn == 0)
15. {
16. cout << "  没有捕获到数据报" << endl;
17. }
18. else
19. {
20. IPPacket = (ARPFrame\_t\*)pkt\_data;
21. *//响应报文的源ip应为所选网卡ip，目的ip为虚假的112.112.112.112.112.112*
22. if (IPPacket->RecvIP == SendIP && IPPacket->SendIP == RevIP)*//判断是否为发送报文的响应*
23. {
24. printIP2MAC(IPPacket);
25. break;
26. }
27. }
28. }

最后打印IP和MAC的对应关系（printIP2MAC函数）即可。

#### 获取远程主机的Mac地址和IP地址关系

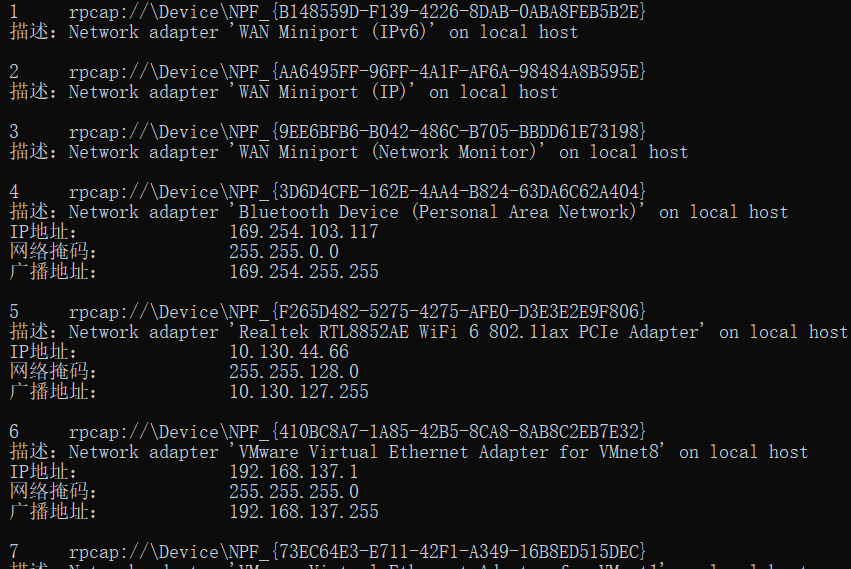
与上述过程类似，只是远程主机的ip由我们自主输入，第4步发送的请求报文的源ip和源mac是我们在第3步已经获得的本机ip和mac。

如下：

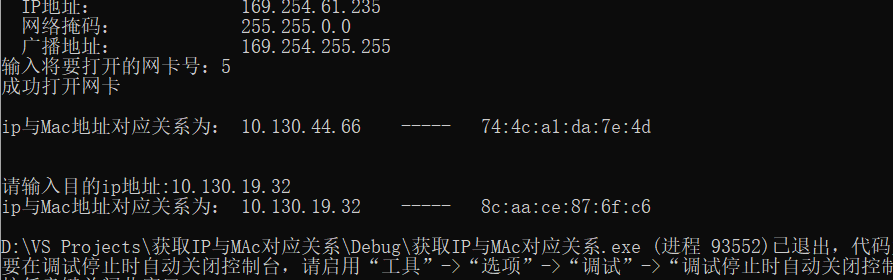
1. *//向网络发送数据包*
2. cout << "\n" << endl;
3. cout << "请输入目的ip地址:";
4. char str[15];
5. cin >> str;
6. RevIP = ARPFrame.RecvIP = inet\_addr(str);  *//目的ip*
7. SendIP = ARPFrame.SendIP = IPPacket->SendIP; *//源ip，即响应报文的源ip，即所选网卡ip*
8. for (int i = 0; i < 6; i++)
9. {
10. ARPFrame.SendHa[i] = ARPFrame.FrameHeader.SrcMAC[i] = IPPacket->SendHa[i]; *//网卡Mac给到新报文的源Mac*
11. }
12. if (pcap\_sendpacket(pcap\_handle, (u\_char\*)&ARPFrame, sizeof(ARPFrame\_t)) != 0)
13. {
14. cout << "ARP请求发送失败" << endl;
15. exit(1);
16. }
17. *//再次捕获*
18. while (true)
19. {
20. int rtn = pcap\_next\_ex(pcap\_handle, &pkt\_header, &pkt\_data);
21. if (rtn == -1)
22. {
23. cout << "  捕获数据包时发生错误：" << errbuf << endl;
24. return 0;
25. }
26. else if (rtn == 0)
27. {
28. cout << "  没有捕获到数据报" << endl;
29. }
30. else
31. {
32. IPPacket = (ARPFrame\_t\*)pkt\_data;
33. if (IPPacket->RecvIP == SendIP && IPPacket->SendIP == RevIP)*//响应报文*
34. {
35. printIP2MAC(IPPacket);
36. break;
37. }
38. }
39. }

#### 5.实验效果

设备列表：



获取对应关系：



可以对获取的两个MAC地址进行验证：

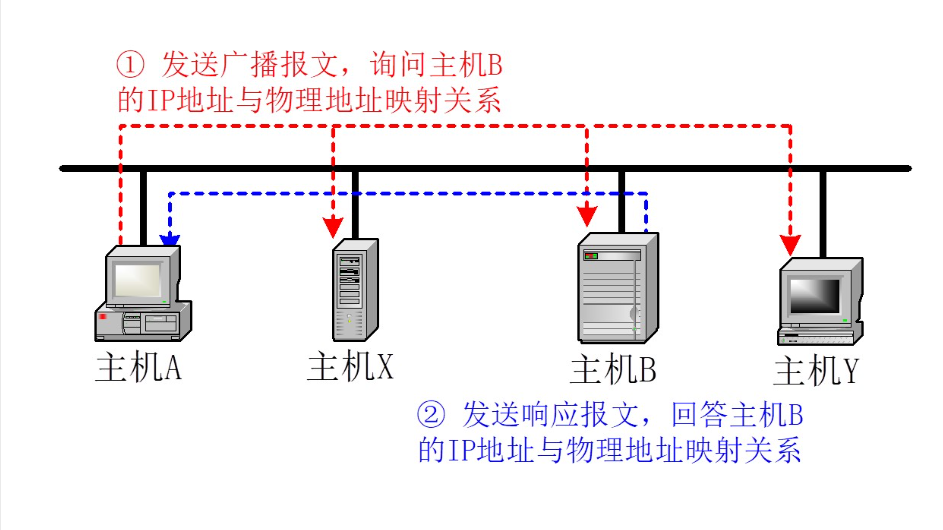




可见确实获取了正确的ip和mac的对应关系。

1. **总结与思考**

##### 完整的ARP工作过程：



如上图，若主机A想要获得主机B的ip和mac的对应关系：

1. A先检查自己高速缓存区的ARP表，查看是否有B的ip和mac地址映射关系。若无，下一步。
2. A广播ARP请求报文，报文请求解析B的ip和mac映射关系，报文含有A的ip地址和mac地址。
3. X、B、Y都接收到上述请求报文，均将A的ip和mac的映射关系存入各自的ARP表中。
4. 主机B发送ARP响应报文，含有映射关系。
5. 主机A接收响应报文，将B的映射关系存入ARP表，此后A可以顺利与B通信。

在2步中，因为A的请求报文只请求了B的映射关系，故X和Y不会发送响应报文。

通过ARP广播报文攻击：

如果A在ARP请求报文中将自己的ip地址改成以太网网关地址，广播出去后，其他非B的主机虽然不会发送响应，但会将A伪造的映射关系存入ARP表；那他们以后发送给网关的请求都有可能发送到伪造的MAC地址上。

##### 实验总结：

除了Npcap的知识，还复习到了课上学习的ARP知识，特别是ARP报文结构和ARP工作过程。