



南开大学
Nankai University

南 开 大 学

网 络 空 间 安 全 学 院

网络技术与应用实验报告

通过编程获取 IP 地址与 MAC 地址的对应关系

李潇逸 2111454

年级：2021 级

专业：信息安全、法学

指导教师：张建忠

2023 年 11 月 18 日

目录

一、 实验要求	1
二、 结果展示	1
三、 代码分析	1
四、 总结	3

一、 实验要求

通过编程获取 IP 地址与 MAC 地址的对应关系实验，要求如下：

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

二、 结果展示

运行代码，首先将直接显示电脑上的所有设备运行列表

```
C:\Users\19309\Desktop\文档 > +
网卡1 rpcap://Device\NPF_{3AB1ED82-365E-49B2-88A1-61A6FA1EC40E}
描述信息: Network adapter 'WAN Miniport (Network Monitor)' on local host
网卡2 rpcap://Device\NPF_{24915485-F862-4B57-B2B8-4B3A1BD35AF8}
描述信息: Network adapter 'WAN Miniport (IPv6)' on local host
网卡3 rpcap://Device\NPF_{C92B1299-7986-4FA5-B451-EDCF808629D}
描述信息: Network adapter 'WAN Miniport (IP)' on local host
网卡4 rpcap://Device\NPF_{D180F9BC-8AD7-44BA-811E-AB4E24CABC2F}
描述信息: Network adapter 'Bluetooth Device (Personal Area Network)' on local host
IP地址: 169.254.30.128
子网掩码: 255.255.0.0
网卡5 rpcap://Device\NPF_{77544071-EF36-49B8-98EB-EA0C8D6A61}
描述信息: Network adapter 'Intel(R) Wi-Fi 6 AX201 160MHz' on local host
IP地址: 10.128.187.98
子网掩码: 255.255.128.0
网卡6 rpcap://Device\NPF_{E9E5FEA6-C74A-428F-8138-8DF9BD178418}
描述信息: Network adapter 'VMware Virtual Ethernet Adapter for VMnet8' on local host
IP地址: 192.168.30.1
子网掩码: 255.255.255.0
网卡7 rpcap://Device\NPF_{524EFF59-FA85-4E22-8D32-528477E66FFE}
描述信息: Network adapter 'VMware Virtual Ethernet Adapter for VMnet1' on local host
IP地址: 192.168.31.1
子网掩码: 255.255.255.0
网卡8 rpcap://Device\NPF_{5E931949-E14F-4E7E-87BD-24B999999C97}
描述信息: Network adapter 'Microsoft Wi-Fi Direct Virtual Adapter #2' on local host
IP地址: 169.254.294.141
子网掩码: 255.255.0.0
网卡9 rpcap://Device\NPF_{DC78E18B-F16E-46B7-B7F5-1595A9E3C7E2}
描述信息: Network adapter 'Microsoft Wi-Fi Direct Virtual Adapter' on local host
IP地址: 169.254.215.165
子网掩码: 255.255.0.0
```

图 1: Caption

之后根据所显示的设备序号选择所要监听的设备并进行监听。此时将打开网卡，同时发送一个 ARP 数据包，得到网卡 IP 和 MAC

```
在网卡序号: 255.255.0.0
请连续打开网卡序号: 5
成功打开网卡
ARP请求发送成功
IP: 10.128.187.98
MAC: dc:46:28:13:b6:72:
```

图 2: Caption

在此之后我们输入想要请求的 IP，得到 MAC

```
向网卡发送一个数据包
请求IP地址: 192.168.43.38
ARP请求发送成功
IP: 192.168.43.38
MAC: dc:46:28:13:b6:72:
```

图 3: Caption

三、 代码分析

首先来看对于 ARP 数据包的构建。在这里我们首先定义了一个帧首部，在其中定义了目的地址、源地址和帧类型；同时也定义了 ARP 的结构体，在其中定义了 ARP 的相应属性。

ARP 数据包

```

1 struct FrameHeader_t //帧首部
2 {
3     BYTE DesMAC[6]; //目的地址
4     BYTE SrcMAC[6]; //源地址
5     WORD FrameType; //帧类型
6 };
7
8 struct ARPFrame_t //ARP帧
9 {
10     FrameHeader_t FrameHeader;
11     WORD HardwareType;
12     WORD ProtocolType;
13     BYTE HLen;
14     BYTE PLen;
15     WORD Operation;
16     BYTE SendHa[6];
17     DWORD SendIP;
18     BYTE RecvHa[6];
19     DWORD RecvIP;
20 };

```

打开网卡的相关部分使用的是第一次的代码，因此不再赘述。
报文需要进行组装，相关代码如下：

组装报文

```

1 //组装报文
2 for (int i = 0; i < 6; i++)
3 {
4     ARPFrame.FrameHeader.DesMAC[i] = 0xFF; //将APRFrame.FrameHeader.DesMAC
        设置为广播地址
5     ARPFrame.FrameHeader.SrcMAC[i] = 0x66; //将APRFrame.FrameHeader.SrcMAC
        设置为本机网卡的MAC地址
6     ARPFrame.RecvHa[i] = 0; //将ARPFrame.RecvHa设置为0表示目的地址未知
7     ARPFrame.SendHa[i] = 0x66; //将ARPFrame.SendHa设置为本机网卡的MAC地址
8 }
9 ARPFrame.FrameHeader.FrameType = htons(0x0806); //帧类型为ARP
10 ARPFrame.HardwareType = htons(0x0001); //硬件类型为以太网
11 ARPFrame.ProtocolType = htons(0x0800); //协议类型为IP
12 ARPFrame.HLen = 6; //硬件地址长度为6
13 ARPFrame.PLen = 4; // 协议地址长为4
14 ARPFrame.Operation = htons(0x0001); //操作为ARP请求
15 SendIP = ARPFrame.SendIP = htonl(0x70707070); //源IP地址设置为虚拟的IP地址
    112.112.112.112.112.112

```

四、 总结

本次实验了解了 IP 地址与 MAC 地址的对应关系，深入理解了相应原理，对网络技术有了更深的了解。

NIKE

参考文献

- [1] 李潇逸. github 作业链接. https://github.com/SesameZMT/NKU_NetworkTechnology.git.

NKU