

廈門大學



信息学院软件工程系

《计算机网络》实验报告

题 目 实验三 基于 PCAP 库侦听并分析网络流量

班 级 软件工程 2023 级 1 班

姓 名 潘騰凱

学 号 37220232203786

实验时间 2025 年 2 月 21 日

2025 年 2 月 15 日

填写说明

- 1、本文件为 Word 模板文件，建议使用 Microsoft Word 2024 打开，在可填写的区域中如实填写；
- 2、填表时勿改变字体字号，保持排版工整，打印为 PDF 文件提交；
- 3、文件总大小尽量控制在 1MB 以下，最大勿超过 5MB；
- 4、在实验课结束 14 天内，按实验报告提交到我校课程网站的指定位置，源代码等主要材料上传在公开的代码托管平台上。
- 5、鼓励同学之间探讨，鼓励合理使用人工智能平台，提升效率，但不应滥用相关资源，如抄袭代码和代写作业。

1 实验目的

通过完成实验，理解数据链路层、网络层、传输层和应用层的基本原理。掌握用 Wireshark 观察网络流量并辅助网络侦听相关的编程；掌握用 Libpcap 或 WinPcap 库侦听并处理以太网帧和 IP 报文的方法；熟悉以太网帧、IP 报文、TCP 段和 FTP 命令的格式概念，掌握 TCP 协议的基本机制；熟悉帧头部或 IP 报文头部各字段的含义。熟悉 TCP 段和 FTP 数据协议的概念，熟悉段头部各字段和 FTP 控制命令的指令和数据的含义。

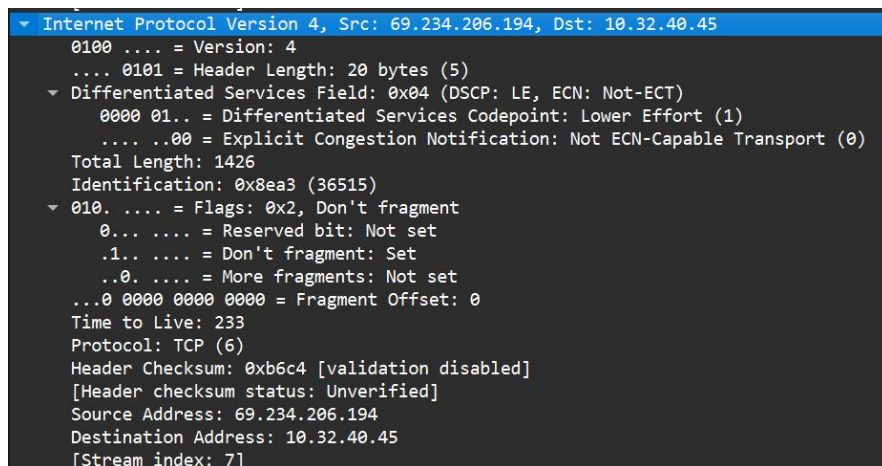
2 实验环境

系统: Windows 11, 软件: WireShark, 开发环境: VS2022 C++, Npcap 库 (Winpcap 库已经停止维护)

3 实验结果

1、用侦听解析软件观察数据格式

用 Wireshark 侦听网络上的数据流，验证理论课讲授的网络协议层次嵌套，验证帧格式、IP 报文格式、TCP 段格式和 FTP 协议命令和响应的格式，验证 MAC 地址、IP 地址、TCP 端口等协议地址格式。



```
▼ Internet Protocol Version 4, Src: 69.234.206.194, Dst: 10.32.40.45
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  ▼ Differentiated Services Field: 0x04 (DSCP: LE, ECN: Not-ECT)
    0000 01.. = Differentiated Services Codepoint: Lower Effort (1)
    .... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
  Total Length: 1426
  Identification: 0x8ea3 (36515)
  ▼ 010. .... = Flags: 0x2, Don't fragment
    0... .... = Reserved bit: Not set
    .1.. .... = Don't fragment: Set
    ..0. .... = More fragments: Not set
    ...0 0000 0000 0000 = Fragment Offset: 0
  Time to Live: 233
  Protocol: TCP (6)
  Header Checksum: 0xb6c4 [validation disabled]
  [Header checksum status: Unverified]
  Source Address: 69.234.206.194
  Destination Address: 10.32.40.45
  [Stream index: 7]
```

如该图所示，这是 wireshark 对一个 IPv4 数据包的详细解析，具体信息如下：

Part1. Internet Protocol Version 4 (IPv4) Header

Version: 4

表示这是一个 IPv4 数据包。

Header Length: 20 bytes (5)

IPv4 头部长度的 20 字节（5 个 32 位字）。

Differentiated Services Field: 0x04 (DSCP: LE, ECN: Not-ECT)

DSCP (Differentiated Services Codepoint): Lower Effort (LE)

值为 0x04，表示该数据包属于 “Lower Effort” 类别。

ECN (Explicit Congestion Notification): Not-ECT

表示该数据包不支持显式拥塞通知。

Total Length: 1426

数据包总长度为 1426 字节，包括头部和数据部分。

Identification: 0x8ea3 (36515)

分段标识符为 0x8ea3，用于唯一标识数据包。

Flags: 0x2, Don't fragment

Don't Fragment (DF) 标志设置为 1，表示该数据包不能被分片。

More Fragments (MF) 标志设置为 0，表示这是最后一个分片。

Fragment Offset: 0

表示该数据包没有偏移量，即它是原始数据包的一部分。

Time to Live: 233

TTL（生存时间）为 233，表示该数据包在网络中可以跳过的最大路由器数。

Protocol: TCP (6)

上层协议是 TCP，值为 6。

Header Checksum: 0xb6c4 [validation disabled]

头部校验和为 0xb6c4，但验证已禁用。

Source Address: 69.234.206.194

源 IP 地址为 69.234.206.194。

Destination Address: 10.32.40.45

目标 IP 地址为 10.32.40.45。

Part2. Stream Index

[Stream index: 7]

这是一个 Wireshark 内部使用的索引，用来标识数据流。在这个例子中，它被设置为 7。

总结

该图展示了一个 IPv4 数据包的详细信息，包括版本、头部长度的、服务类型字段、总长度、标识符、标志位、TTL、协议类型、头部校验和以及源/目标 IP 地址。这些信息帮我进一步理解了数据包在网络中的传输方式和处理过程。

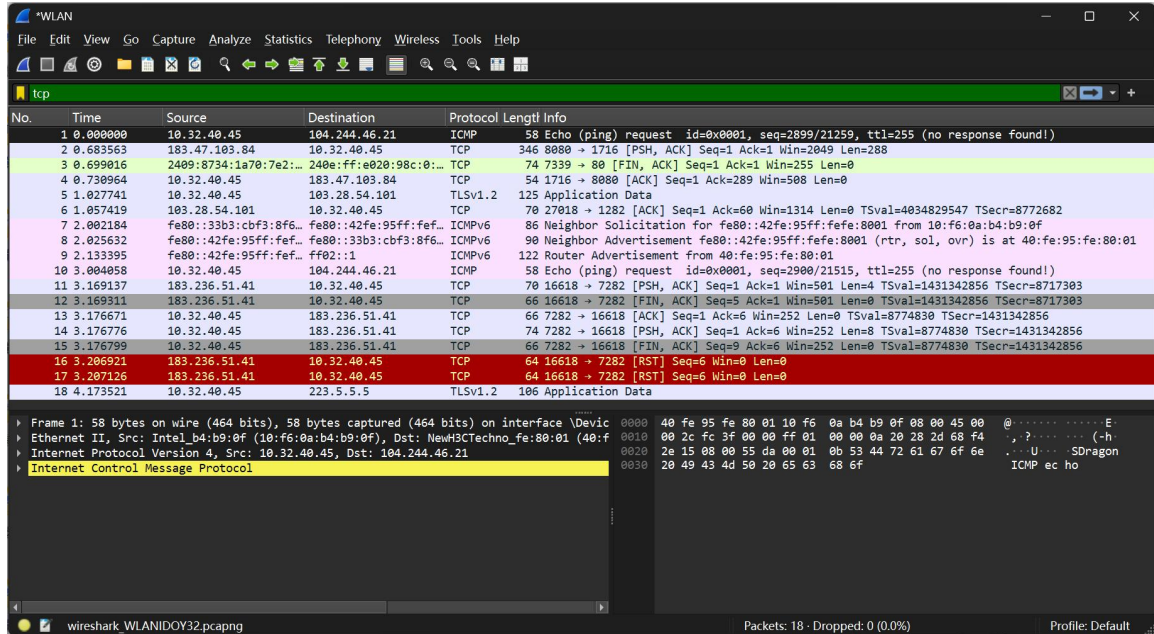
2、用侦听解析软件观察 TCP 机制

用 Wireshark 侦听并观察 TCP 数据段。观察其建立和撤除连接的过程，观察段 ID、窗口机制和拥塞控制机制等。将该过程截图在报告中。

Step1: 过滤并观察 TCP 流量

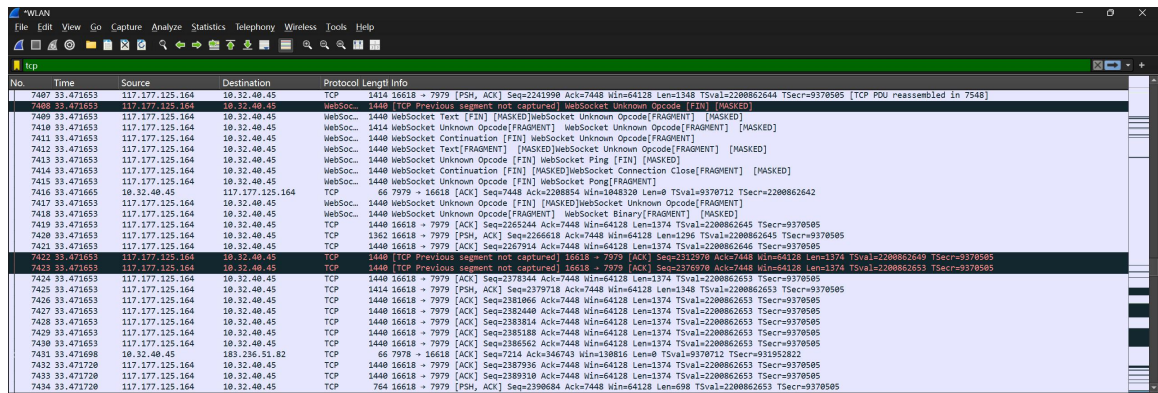
在捕获过滤器中输入 tcp，或者在捕获过程中或之后使用显示过滤器来筛选出 TCP 数据包。

执行一些会产生 TCP 流量的动作，例如浏览网页、发送电子邮件等。



Step2: 分析 TCP 三次握手和四次挥手过程

打开并关闭“人民网”，tcp 流监听结果如下



三次握手：这是 TCP 建立连接的过程，由客户端发起 SYN 请求，服务器回复 SYN-ACK 响应，最后客户端发送 ACK 确认消息。

四次挥手：这是 TCP 关闭连接的过程，由一方首先发送 FIN 消息，对方回应 ACK，接着另一方也发送 FIN 消息，原发送方回应 ACK 完成关闭。

步骤五：观察段 ID、窗口机制和拥塞控制

段 ID（序列号和确认号）：每个 TCP 数据包都包含一个序列号和确认号，用于确保数据按顺序正确到达。

窗口机制：窗口大小字段表示发送方愿意接收的数据量，这有助于控制数据流，避免网络过载。

拥塞控制机制：TCP 使用多种算法如慢启动、拥塞避免、快速重传和快速恢复等来管理网络拥塞。

3、用 Libpcap 或 WinPcap 库侦听网络数据

用 Libpcap 或 WinPcap 库侦听网络上的数据流，解析发送方与接收方的 MAC 和 IP 地址，并作记录与统计。程序在文件上输出形如下列 CSV 格式的日志：

时间、源 MAC、源 IP、目标 MAC、目标 IP、帧长度（以逗号间隔）

2015-03-14 13:05:16,60-36-DD-7D-D5-21,192.168.33.1,60-36-DD-7D-D5-72,192.168.33.2,1536

主要代码如下：

```

17
18 // 获取当前时间字符串
19 std::string current_time() {
20     auto now = std::chrono::system_clock::now();
21     std::time_t now_time = std::chrono::system_clock::to_time_t(now);
22     std::tm local_tm;
23     localtime_s(&local_tm, &now_time);
24     std::ostringstream oss;
25     oss << std::put_time(&local_tm, "%Y-%m-%d %H:%M:%S");
26     return oss.str();
27 }
28
29 // 解析数据包并记录日志
30 void packet_handler(u_char* user, const struct pcap_pkthdr* header, const u_char* packet) {
31     std::lock_guard<std::mutex> lock(log_mutex);
32     std::ofstream log_file("network_log.csv", std::ios::app);
33     if (!log_file) return;
34
35     struct ether_header {
36         u_char dest_mac[6];
37         u_char src_mac[6];
38         u_short eth_type;
39     };
40
41     struct ip_header {
42         u_char ihl_version;
43         u_char tos;
44         u_short total_length;
45         u_short id;
46         u_short flag_frag;
47         u_char ttl;
48         u_char protocol;
49         u_short checksum;
50         u_char src_ip[4];
51         u_char dest_ip[4];
52     };

```

结果如下：

选择监听的网卡：

```

C:\Users\86158\Desktop\ x + v
可用的网络接口：
1. WAN Miniport (Network Monitor) - \Device\NPF_{BD67F990-B75B-48E1-98D1-4F6C71617748}
2. WAN Miniport (IPv6) - \Device\NPF_{AEF03BA0-9E86-4E7D-92E8-B69464E71475}
3. WAN Miniport (IP) - \Device\NPF_{FE54C4BF-EFE3-430E-ADDC-DDBE85CA1CC9}
4. Bluetooth Device (Personal Area Network) - \Device\NPF_{A679C00E-48B8-4288-AEE4-94B656B78407}
5. Intel(R) Wi-Fi 6E AX211 160MHz - \Device\NPF_{333B194C-0644-4E01-8111-E3EA9BB0A7BB}
6. Microsoft Wi-Fi Direct Virtual Adapter #2 - \Device\NPF_{73278F46-8889-447D-B780-62F2D1A8D021}
7. Microsoft Wi-Fi Direct Virtual Adapter - \Device\NPF_{938E2E0F-3BCB-49EA-8BCA-F3E97E2AFCAF}
8. Adapter for loopback traffic capture - \Device\NPF_{Loopback}
9. Sangfor SSL VPN CS Support System VNIC - \Device\NPF_{4756CC10-9B8A-4BF0-9279-A9BABA68B357}
10. Realtek Gaming GbE Family Controller - \Device\NPF_{D866121D-DC8A-4883-96E1-CE440A1F5CCC}
11. TAP-Windows Adapter V9 - \Device\NPF_{F1807DBB-C2EB-4876-8B85-1DEE5F002F37}
选择接口编号：5
开始监听数据包...

```

内容都输出到 network_log.csv 里，如下：

列属性从左往右分别为时间、源 MAC 地址，源 IP，目的 MAC 地址，目的 IP，帧长度。

	A	B	C	D	E	F
3900	4/3/2025 18:08	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	210.34.0.18	82
3901	4/3/2025 18:08	40-FE-95-FE-80-01	210.34.0.18	10-F6-0A-B4-B9-0F	10.32.40.45	210
3902	4/3/2025 18:08	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	210.34.0.18	82
3903	4/3/2025 18:08	40-FE-95-FE-80-01	210.34.0.18	10-F6-0A-B4-B9-0F	10.32.40.45	120
3904	4/3/2025 18:08	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	110.249.194.71	74
3905	4/3/2025 18:08	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	110.249.194.71	74
3906	4/3/2025 18:08	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	110.249.194.71	66
3907	4/3/2025 18:08	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	110.249.194.71	210
3908	4/3/2025 18:08	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	110.249.194.71	955
3909	4/3/2025 18:08	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	110.249.194.71	66
3910	4/3/2025 18:08	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	110.249.194.71	211
3911	4/3/2025 18:08	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	110.249.194.71	1386
3912	4/3/2025 18:08	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	110.249.194.71	782
3913	4/3/2025 18:08	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	110.249.194.71	66
3914	4/3/2025 18:08	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	110.249.194.71	66
3915	4/3/2025 18:08	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	110.249.194.71	210
3916	4/3/2025 18:08	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	110.249.194.71	890
3917	4/3/2025 18:08	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	210.34.0.18	76
3918	4/3/2025 18:08	40-FE-95-FE-80-01	210.34.0.18	10-F6-0A-B4-B9-0F	10.32.40.45	92
3919	4/3/2025 18:08	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	210.34.0.18	76
3920	4/3/2025 18:08	40-FE-95-FE-80-01	210.34.0.18	10-F6-0A-B4-B9-0F	10.32.40.45	76
3921	4/3/2025 18:08	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	210.34.0.18	70
3922	4/3/2025 18:08	40-FE-95-FE-80-01	210.34.0.18	10-F6-0A-B4-B9-0F	10.32.40.45	86
3923	4/3/2025 18:08	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	210.34.0.18	70
3924	4/3/2025 18:08	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	110.43.81.79	74
3925	4/3/2025 18:08	40-FE-95-FE-80-01	210.34.0.18	10-F6-0A-B4-B9-0F	10.32.40.45	135
3926	4/3/2025 18:08	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	120.52.183.190	74
3927	4/3/2025 18:08	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	110.249.194.71	66
3928	4/3/2025 18:08	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	110.43.81.79	54
3929	4/3/2025 18:08	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	110.43.81.79	309
3930	4/3/2025 18:08	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	120.52.183.190	54
3931	4/3/2025 18:08	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	120.52.183.190	261
3932	4/3/2025 18:08	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	110.43.81.79	329
3933	4/3/2025 18:08	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	120.52.183.190	54
3934	4/3/2025 18:08	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	110.249.194.71	211

每分钟捕获一次数据：

	A	B	C	D	E	F
4469	4/3/2025 18:08	40-FE-95-F36.156.184	10-F6-0A-F10.32.40.4			66
4470	4/3/2025 18:08	40-FE-95-F36.156.184	10-F6-0A-F10.32.40.4			124
4471	4/3/2025 18:08	10-F6-0A-F10.32.40.4	40-FE-95-F36.156.184			66
4472	4/3/2025 18:08	40-FE-95-F36.156.184	10-F6-0A-F10.32.40.4			877
4473	4/3/2025 18:08	40-FE-95-F36.156.184	10-F6-0A-F10.32.40.4			117
4474	4/3/2025 18:08	10-F6-0A-F10.32.40.4	40-FE-95-F36.156.184			66
4475	4/3/2025 18:08	10-F6-0A-F10.32.40.4	40-FE-95-F36.156.184			124
4476	4/3/2025 18:08	40-FE-95-F36.156.184	10-F6-0A-F10.32.40.4			66
4477	4/3/2025 18:13	10-F6-0A-F10.32.40.4	40-FE-95-F183.47.110			54
4478	4/3/2025 18:13	10-F6-0A-F10.32.40.4	40-FE-95-F124.70.33.			134
4479	4/3/2025 18:13	10-F6-0A-F10.32.40.4	40-FE-95-F110.249.19			66
4480	4/3/2025 18:13	10-F6-0A-F10.32.40.4	40-FE-95-F110.249.19			66
4481	4/3/2025 18:13	10-F6-0A-F10.32.40.4	40-FE-95-F110.249.19			66
4482	4/3/2025 18:13	10-F6-0A-F10.32.40.4	40-FE-95-F110.249.19			66
4483	4/3/2025 18:13	10-F6-0A-F10.32.40.4	40-FE-95-F110.249.19			74
4484	4/3/2025 18:13	10-F6-0A-F10.32.40.4	40-FE-95-F110.249.19			74
4485	4/3/2025 18:13	10-F6-0A-F10.32.40.4	40-FE-95-F210.34.0.7			81
4486	4/3/2025 18:13	40-FE-95-F210.34.0.7	10-F6-0A-F10.32.40.4			144
4487	4/3/2025 18:13	10-F6-0A-F10.32.40.4	40-FE-95-F210.34.0.7			81

4、解析侦听到的网络数据

用 Wireshark 侦听并观察 FTP 数据，分析其用户名密码所在报文的上下文特征，再总结出提取用户名密码的有效方法。解析协议内容，并作记录与统计。对用户登录行为进行记录。程序在文件上输出形如下列 CSV 格式的日志：

时间、源 MAC、源 IP、目标 MAC、目标 IP、登录名、口令、成功与否

2015-03-14 13:05:16,60-36-DD-7D-D5-21,192.168.33.1,60-36-DD-7D-D5-72,192.168.33.2,student,software,SUCCEED

2015-03-14 13:05:16,60-36-DD-7D-D5-21,192.168.33.1,60-36-DD-7D-D5-72,192.168.33.2,student,software1,FAILED

通过监听 FTP 通信时的网络分组，了解 FTP 的通信协议的过程。在此基础上，重点掌握 FTP 登陆环节的通信过程。

通过实验 3 熟悉网络分组的格式，并对“数据区”进行分析。通过对分组中“数据区”的分析，提取用户名、密码、登陆是否成功的反馈信息。一般登录名以“USER”开头，口令以“PASS”开头，登录成功以“230”开头，失败以“530”开头。

```

Microsoft Visual Studio 调 × + v
输入要读取的 pcapng 文件名: C:\Users\86158\Desktop\计网\capture.pcapng
开始解析数据包...
解析完成, 日志已写入 network_log.csv

C:\Users\86158\Desktop\计网\Exp3_00\x64\Release\Exp3_00.exe (进程 31232)已退出, 代码为 0。
按任意键关闭此窗口. . . |

```

	A	B	C	D	E	F
1	4/3/2025 17:57	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	142.250.198.65	74
2	4/3/2025 17:57	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	142.250.198.74	74
3	4/3/2025 17:57	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	103.28.54.100	144
4	4/3/2025 17:57	40-FE-95-FE-80-01	103.28.54.100	10-F6-0A-B4-B9-0F	10.32.40.45	66
5	4/3/2025 17:57	40-FE-95-FE-80-01	103.28.54.100	10-F6-0A-B4-B9-0F	10.32.40.45	1404
6	4/3/2025 17:57	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	103.28.54.100	66
7	4/3/2025 17:57	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	103.28.54.100	156
8	4/3/2025 17:57	40-FE-95-FE-80-01	103.28.54.100	10-F6-0A-B4-B9-0F	10.32.40.45	66
9	4/3/2025 17:57	40-FE-95-FE-80-01	103.28.54.100	10-F6-0A-B4-B9-0F	10.32.40.45	271
10	4/3/2025 17:57	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	103.28.54.100	66
11	4/3/2025 17:57	40-FE-95-FE-80-01	69.234.206.194	10-F6-0A-B4-B9-0F	10.32.40.45	398
12	4/3/2025 17:57	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	69.234.206.194	66
13	4/3/2025 17:57	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	69.234.206.194	74
14	4/3/2025 17:57	40-FE-95-FE-80-01	69.234.206.194	10-F6-0A-B4-B9-0F	10.32.40.45	97
15	4/3/2025 17:57	40-FE-95-FE-80-01	69.234.206.194	10-F6-0A-B4-B9-0F	10.32.40.45	66
16	4/3/2025 17:57	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	69.234.206.194	54
17	4/3/2025 17:57	40-FE-95-FE-80-01	69.234.206.194	10-F6-0A-B4-B9-0F	10.32.40.45	74
18	4/3/2025 17:57	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	69.234.206.194	66
19	4/3/2025 17:57	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	69.234.206.194	268
20	4/3/2025 17:57	40-FE-95-FE-80-01	69.234.206.194	10-F6-0A-B4-B9-0F	10.32.40.45	66
21	4/3/2025 17:57	40-FE-95-FE-80-01	69.234.206.194	10-F6-0A-B4-B9-0F	10.32.40.45	1440
22	4/3/2025 17:57	40-FE-95-FE-80-01	69.234.206.194	10-F6-0A-B4-B9-0F	10.32.40.45	1440
23	4/3/2025 17:57	40-FE-95-FE-80-01	69.234.206.194	10-F6-0A-B4-B9-0F	10.32.40.45	1440
24	4/3/2025 17:57	40-FE-95-FE-80-01	69.234.206.194	10-F6-0A-B4-B9-0F	10.32.40.45	390
25	4/3/2025 17:57	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	69.234.206.194	66
26	4/3/2025 17:57	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	69.234.206.194	66
27	4/3/2025 17:57	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	69.234.206.194	192
28	4/3/2025 17:57	40-FE-95-FE-80-01	69.234.206.194	10-F6-0A-B4-B9-0F	10.32.40.45	66
29	4/3/2025 17:57	40-FE-95-FE-80-01	69.234.206.194	10-F6-0A-B4-B9-0F	10.32.40.45	270
30	4/3/2025 17:57	40-FE-95-FE-80-01	69.234.206.194	10-F6-0A-B4-B9-0F	10.32.40.45	135
31	4/3/2025 17:57	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	69.234.206.194	66
32	4/3/2025 17:57	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	69.234.206.194	153
33	4/3/2025 17:57	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	69.234.206.194	249
34	4/3/2025 17:57	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	69.234.206.194	104
35	4/3/2025 17:57	10-F6-0A-B4-B9-0F	10.32.40.45	40-FE-95-FE-80-01	69.234.206.194	655

4 实验代码

本次实验的代码已上传于以下代码仓库：

<https://gitee.com/larks-csdsc/exp-03-npcap/tree/master/>。

5 课后思考题

无

6 实验总结

在本次实验聚焦网络流量侦听与分析，借助 Wireshark、Npcap 库和 VS2022 开发环境，实现理论与实践的结合。

实验中，Wireshark 助力我直观认识网络协议层次嵌套，深入理解 IPv4 数据包、TCP 段、FTP 协议格式及各层原理。观察 TCP 机制，清晰看到连接建立、拆除过程，以及段 ID、窗口和拥塞控制机制，巩固了对 TCP 协议的认知。利用 Npcap 库侦听网络数据、解析地址，提升了编程和数据处理能力，解析 FTP 数据也加深了对应用层协议的理解。

但实验并非毫无阻碍，配置环境和库时遭遇文件缺失、路径错误等问题，经不断尝试才解决，这也锻炼了我的问题解决能力。同时，处理大量数据时意识到优化代码性能的重要性。

此次实验将理论知识与实际操作紧密相连，提升了我的动手能力、问题解决能力，加深了对网络协议的理解。