

廈門大學



信息学院软件工程系

《计算机网络》实验报告

题 目 实验四 观察 TCP 报文段并侦听分析 FTP 协议

姓 名 杨浩然

学 号 24320182203309

班 级 软件工程 2018 级 2 班

实验时间 2020 年 3 月 26 日

2020 年 3 月 26 日

1 实验目的

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。

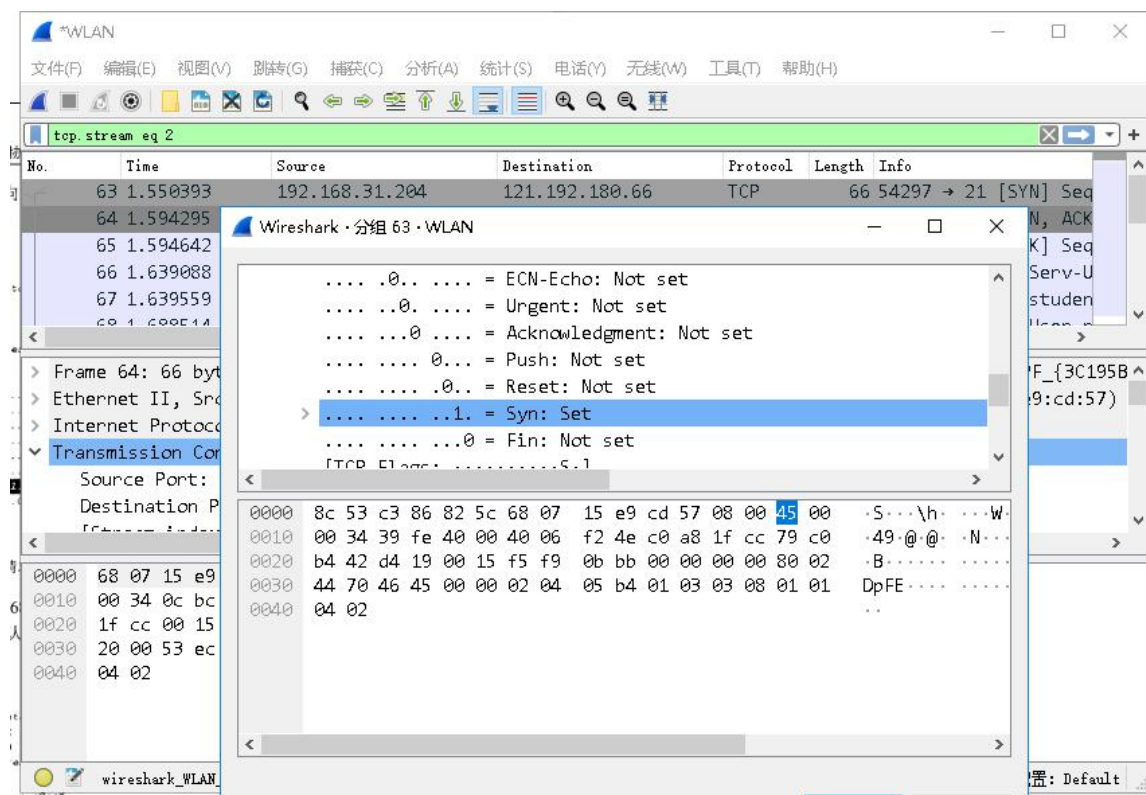
用 Wireshark 侦听并观察 FTP 数据，分析其用户名密码所在报文的上下文特征，再总结出提取用户名密码的有效方法。基于 WinPCAP 工具包制作程序，实现监听网络上的 FTP 数据流，解析协议内容，并作记录与统计。对用户登录行为进行记录。

2 实验环境

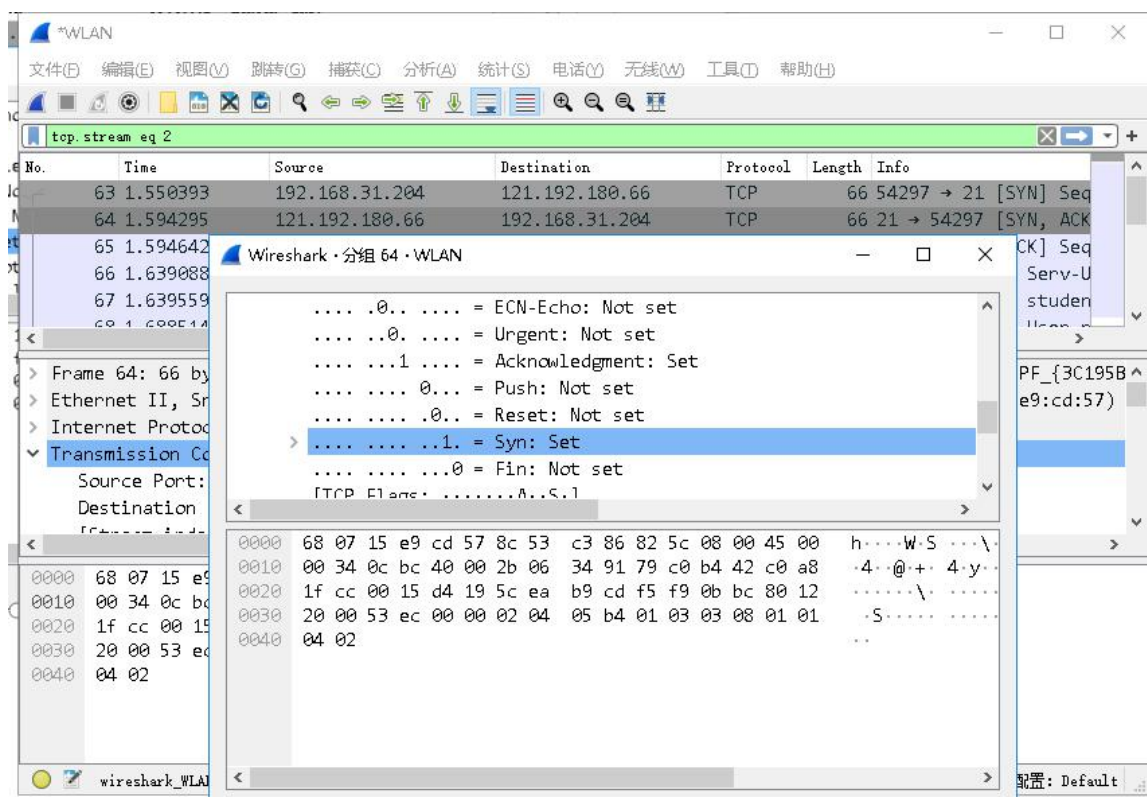
Win10, C++。

3 实验结果

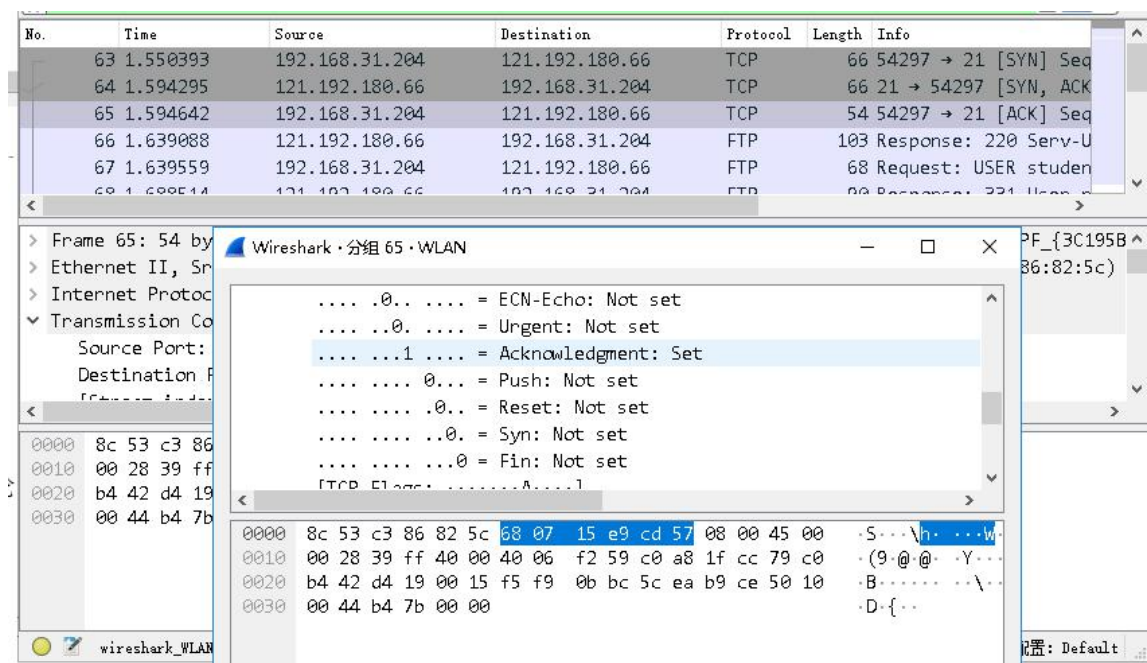
第一个数据包是本机 192.168.31.204 向 FTP 发送连接请求，标志位中只有 Syn



第二个数据包是 FTP 服务器发来的握手信号，里面有同步位和确认位



第三个数据包是本机发出的，里面只有一个确认位，也就是做再次确认



第四个数据包是 FTP 服务器发来的欢迎语句

No.	Time	Source	Destination	Protocol	Length	Info
63	1.550393	192.168.31.204	121.192.180.66	TCP	66	54297 → 21 [SYN] Seq
64	1.594295	121.192.180.66	192.168.31.204	TCP	66	21 → 54297 [SYN, ACK
65	1.594642	192.168.31.204	121.192.180.66	TCP	54	54297 → 21 [ACK] Seq
66	1.639088	121.192.180.66	192.168.31.204	FTP	103	Response: 220 Serv-U
67	1.639559	192.168.31.204	121.192.180.66	FTP	68	Request: USER studen
68	1.689514	121.192.180.66	192.168.31.204	FTP	80	Response: 331 User n

> Frame 66: 103 bytes on wire (824 bits), 103 bytes captured (824 bits) on interface \Device\NPF_{3C19...}

> Ethernet II, Src: BeijingX_86:82:5c (8c:53:c3:86:82:5c), Dst: IntelCor_e9:cd:57 (68:07:15:e9:cd:57)

> Internet Protocol Version 4, Src: 121.192.180.66, Dst: 192.168.31.204

▼ Transmission Control Protocol, Src Port: 21, Dst Port: 54297, Seq: 1, Ack: 1, Len: 49

Source Port: 21

Destination Port: 54297

[Frame is loaded...]

0000	68 07 15 e9 cd 57 8c 53 c3 86 82 5c 08 00 45 00	h...W.S...E.
0010	00 59 0c bd 40 00 2b 06 34 6b 79 c0 b4 42 c0 a8	Y...@...4ky...B.
0020	1f cc 00 15 d4 19 5c ea b9 ce f5 f9 0b bc 50 18\...P.
0030	01 00 0b b7 00 00 32 32 30 20 53 65 72 76 2d 5522 0 Serv-U
0040	20 46 54 50 20 53 65 72 76 65 72 20 76 36 2e 32	FTP Ser ver v6.2
0050	20 66 6f 72 20 57 69 6e 53 6f 63 6b 20 72 65 61	for Win Sock rea
0060	64 79 2e 2e 2e 0d 0a	dy....

wireshark_WLAN_20200331143220_al1604.pcapng | 分组: 211 · 已显示: 21 (10.0%) · 已丢弃: 0 (0.0%) | 配置: Default

第五个数据包中含用户名

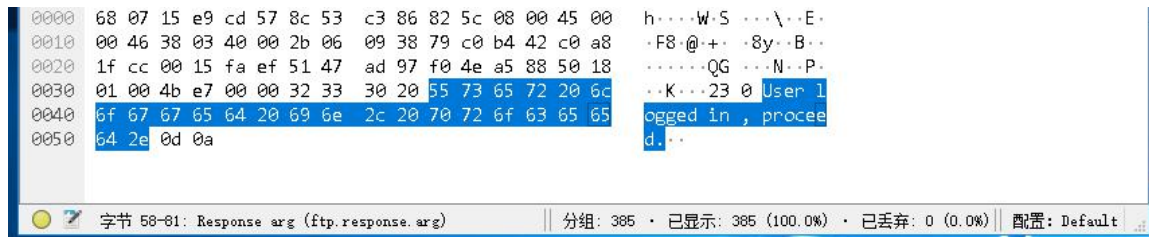
0000	8c 53 c3 86 82 5c 68 07 15 e9 cd 57 08 00 45 00	.S...h...W...E.
0010	00 36 3a 00 40 00 40 06 f2 4a c0 a8 1f cc 79 c0	.6:@@...J...y.
0020	b4 42 d4 19 00 15 f5 f9 0b bc 5c ea b9 ff 50 18	.B.....\...P.
0030	00 44 a4 c2 00 00 55 53 45 52 20 73 74 75 64 65	.D...US ER stude
0040	6e 74 0d 0a	nt..

wireshark_WLAN_20200331143220_al1604.pcapng | 分组: 211 · 已显示: 21 (10.0%) · 已丢弃: 0 (0.0%) | 配置: Default

第七个数据包中含密码

0000	8c 53 c3 86 82 5c 68 07 15 e9 cd 57 08 00 45 00	.S...h...W...E.
0010	00 37 3a 01 40 00 40 06 f2 48 c0 a8 1f cc 79 c0	.7:@@...H...y.
0020	b4 42 d4 19 00 15 f5 f9 0b ca 5c ea ba 23 50 18	.B.....\...#P.
0030	00 44 3b 9c 00 00 50 41 53 53 20 73 6f 66 74 77	.D;...PA SS softw
0040	61 72 65 0d 0a	are..

如果登陆成功则以 230 开头



找到用户名及密码前分别有 USER 和 PASS，通过遍历每个包的内容进行筛选

```
//形同特征则输出文件
for (int i = 0; i < length; i++) {
    if (pkt_data[i] == 85 && pkt_data[i + 1] == 83 && pkt_data[i + 2] == 69 && pkt_data[i + 3] == 82) {
        fprintf(fp, "%s", " ", timestr);
        for (int i = 0; i < 6; i++) {
            fprintf(fp, "%02X ", mh->src_addr[i]);
        }
        fprintf(fp, " ");
        for (int i = 0; i < 4; i++) {
            fprintf(fp, "%d.", ih->saddr[i]);
        }
        fprintf(fp, " ");
        for (int i = 0; i < 6; i++) {
            fprintf(fp, "%02X ", mh->dest_addr[i]);
        }
        fprintf(fp, " ");
        for (int i = 0; i < 4; i++) {
            fprintf(fp, "%d.", ih->daddr[i]);
        }
        fprintf(fp, " ");
        for (int j = i + 4; pkt_data[j] != 13; j++) fputc(pkt_data[j], fp);
        fprintf(fp, " ");
        fclose(fp);
    }
    else if (pkt_data[i] == 80 && pkt_data[i + 1] == 65 && pkt_data[i + 2] == 83 && pkt_data[i + 3] == 83) {
        FILE *fp1 = fopen("user.csv", "a");
        for (int j = i + 4; pkt_data[j] != 13; j++) fputc(pkt_data[j], fp1);
        fprintf(fp1, " ");
        fclose(fp1);
    }
    else if (pkt_data[i] == 50 && pkt_data[i + 1] == 51 && pkt_data[i + 2] == 48) {
```

出现 230 则成功登入，530 则失败登入。以数据 ASCLL 码值对应

```
    else if (pkt_data[i] == 50 && pkt_data[i + 1] == 51 && pkt_data[i + 2] == 48) {
        FILE *fp2 = fopen("user.csv", "a");
        fprintf(fp2, "SUCCEED ");
        fclose(fp2);
    }
    else if (pkt_data[i] == 53 && pkt_data[i + 1] == 51 && pkt_data[i + 2] == 48) {
        FILE *fp2 = fopen("user.csv", "a");
        fprintf(fp2, "FAILED ");
        fclose(fp2);
    }
```


得到结果

	A	B	C	D	E	F	G	H	
1	19:33:16	8C53C386	192.168.	680715e9	121.192.	student	software	SUCCEED	
2	19:36:52	8C53C386	192.168.	680715e9	121.192.	student	software	FAILED	
3									

4 实验总结

TCP 建立过程分为三次握手，第一次是客户端发起连接，客户端 TCP 发送一个 Syn 同步记号，申请和服务器连接。

第二次是服务器确认了客户端的 Syn，发送 Acknowledgement 记号和一个 Syn 同步记号。

第三次是客户端确认服务器的 Ack 和 Syn 记号，向服务器发送 Ack，成功建立连接。

每次传输的 TCP 数据包中格式都如版本号 4 位，数据包头部长度 4 位，服务类型 8 位，总长 16 位，重组标识 16 位等。最后才是用户数据。