

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

班 级 软件工程 2018 级 2 班

姓 名 林正男

学 号 20420172201787

实验时间 2020 年 3 月 25 日

2020 年 3 月 27 日

1 实验目的

1.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。


2. 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。

2 实验环境

Windows10 c++ wireshark

3 实验结果

添加 FTP 站点 ? ×

 **站点信息**

FTP 站点名称(I):

内容目录

物理路径(H):

创建自己的 ftp



授权一个名 user 的用户，口令为 123

但因为都是一台计算机的地址，wireshark 或者编写的 winpcap 程序只能检测到经由网卡的数据传输，无法检测到 localhost 的数据传输。

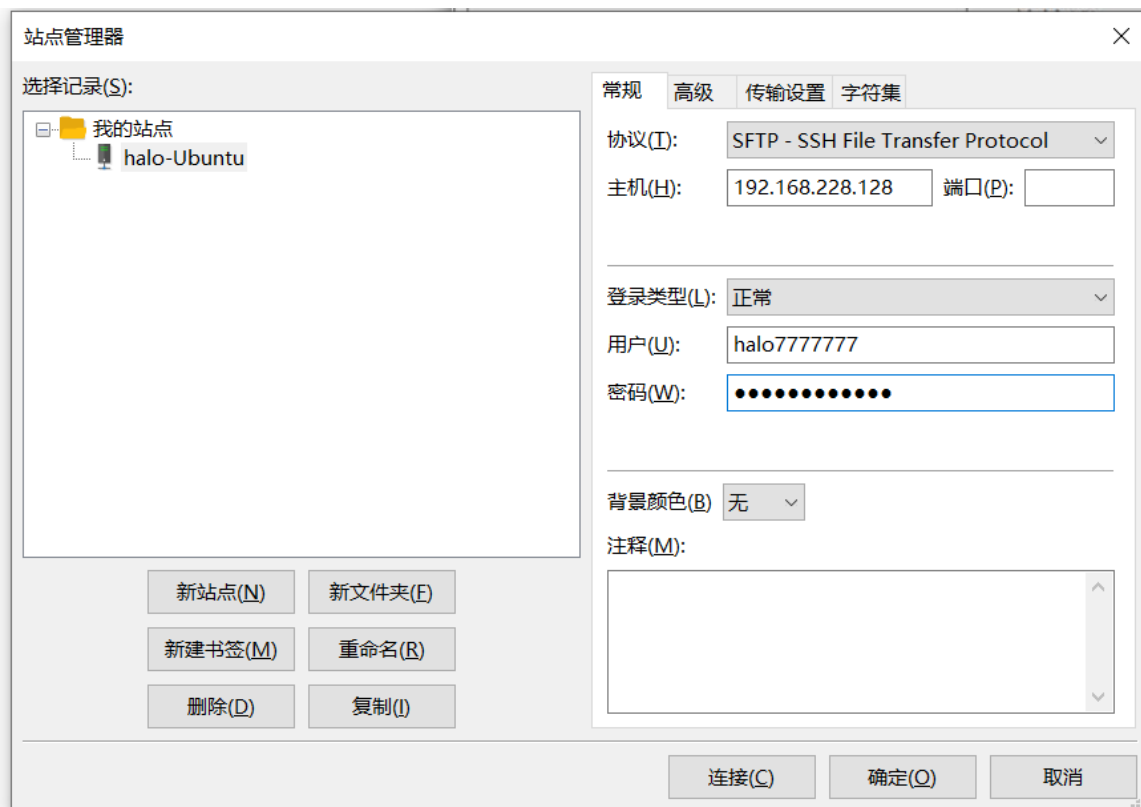
```
C:\WINDOWS\system32>route add [REDACTED] mask 255.255.255.255 [REDACTED]
操作完成!

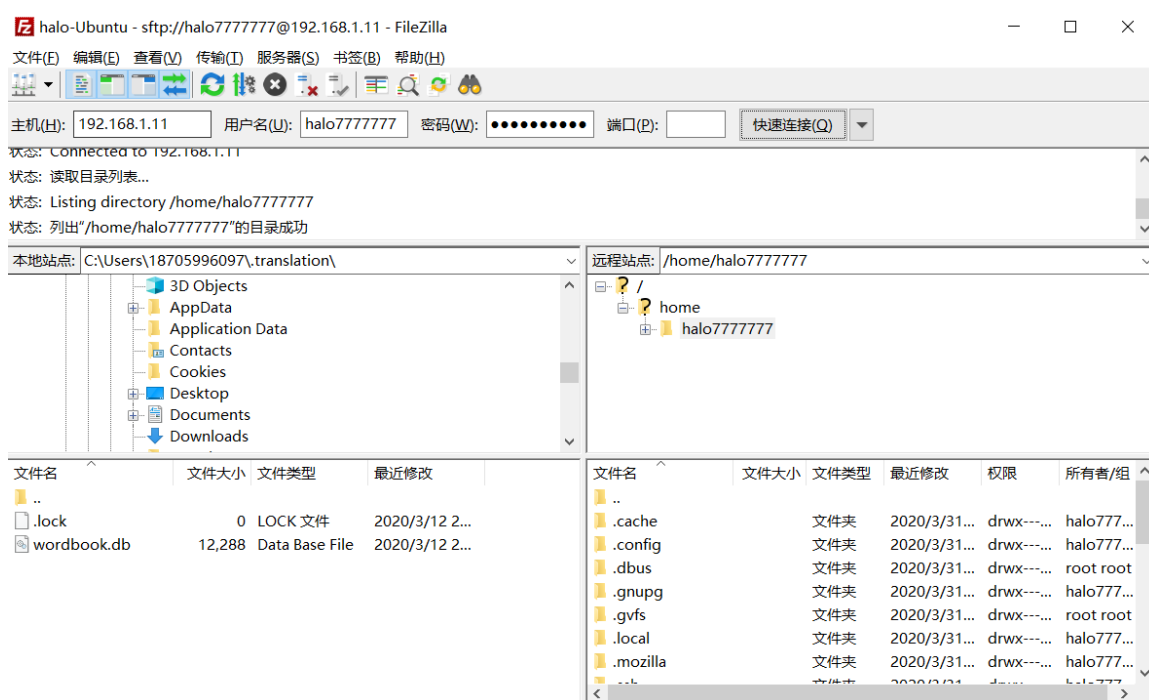
C:\WINDOWS\system32>
```

改变 IP 地址，使传输的数据经由网卡，但 ftp 也会因为其改变而无法打开，
使用虚拟机 Ubuntu 系统

```
halo777777@halo777777-virtual-machine: ~  
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)  
halo777777@halo777777-virtual-machine:~$ ifconfig  
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
    inet 192.168.1.11 netmask 255.255.255.0 broadcast 192.168.1.255  
    inet6 fe80::2746:646a:5051:8b8f prefixlen 64 scopeid 0x20<link>  
    inet6 2409:8a34:a019:bc40:90a0:5af9:627:99ba prefixlen 64 scopeid 0x0  
<global>  
    inet6 2409:8a34:a019:bc40:782a:755:efce:48a9 prefixlen 64 scopeid 0x0  
<global>  
    ether 00:0c:29:73:02:9c txqueuelen 1000 (以太网)  
    RX packets 38463 bytes 55439907 (55.4 MB)  
    RX errors 0 dropped 5 overruns 0 frame 0  
    TX packets 4679 bytes 354418 (354.4 KB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536  
    inet 127.0.0.1 netmask 255.0.0.0  
    inet6 ::1 prefixlen 128 scopeid 0x10<host>  
    loop txqueuelen 1000 (本地环回)  
    RX packets 267 bytes 21612 (21.6 KB)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 267 bytes 21612 (21.6 KB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

虚拟地址为 192.168.1.11





使用 FileZilla 创建与虚拟主机的 ftp 连接

43	3.575465	192.168.1.4	192.168.1.11	TCP	66 11619 → 22 [SYN] Seq=0 Win=65535 Len=0 MSS=1440 WS=128 SACK_PERM=1
44	3.575705	192.168.1.11	192.168.1.4	TCP	66 22 → 11619 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM=1 WS=128
45	3.575781	192.168.1.4	192.168.1.11	TCP	54 11619 → 22 [ACK] Seq=1 Ack=1 Win=4194304 Len=0
46	3.575887	192.168.1.4	192.168.1.11	SSHv2	82 Client: Protocol (SSH-2.0-FileZilla_3.47.2.1)
47	3.581690	192.168.1.11	192.168.1.4	SSHv2	95 Server: Protocol (SSH-2.0-OpenSSH_7.6p1 Ubuntu-4ubuntu0.3)
48	3.582007	192.168.1.11	192.168.1.4	TCP	60 22 → 11619 [ACK] Seq=42 Ack=29 Win=64256 Len=0
49	3.583629	192.168.1.4	192.168.1.11	SSHv2	1222 Client: Key Exchange Init
50	3.583748	192.168.1.11	192.168.1.4	TCP	60 22 → 11619 [ACK] Seq=42 Ack=1197 Win=64128 Len=0
51	3.583889	192.168.1.11	192.168.1.4	SSHv2	1134 Server: Key Exchange Init
52	3.586595	192.168.1.4	192.168.1.11	SSHv2	102 Client: Elliptic Curve Diffie-Hellman Key Exchange Init
53	3.586720	192.168.1.11	192.168.1.4	TCP	60 22 → 11619 [ACK] Seq=1122 Ack=1245 Win=64128 Len=0
54	3.591711	192.168.1.11	192.168.1.4	SSHv2	262 Server: Elliptic Curve Diffie-Hellman Key Exchange Reply, New Keys

可观察到三次握手等情况，因为连接时协议为 SFTP 所以为 SSHV2

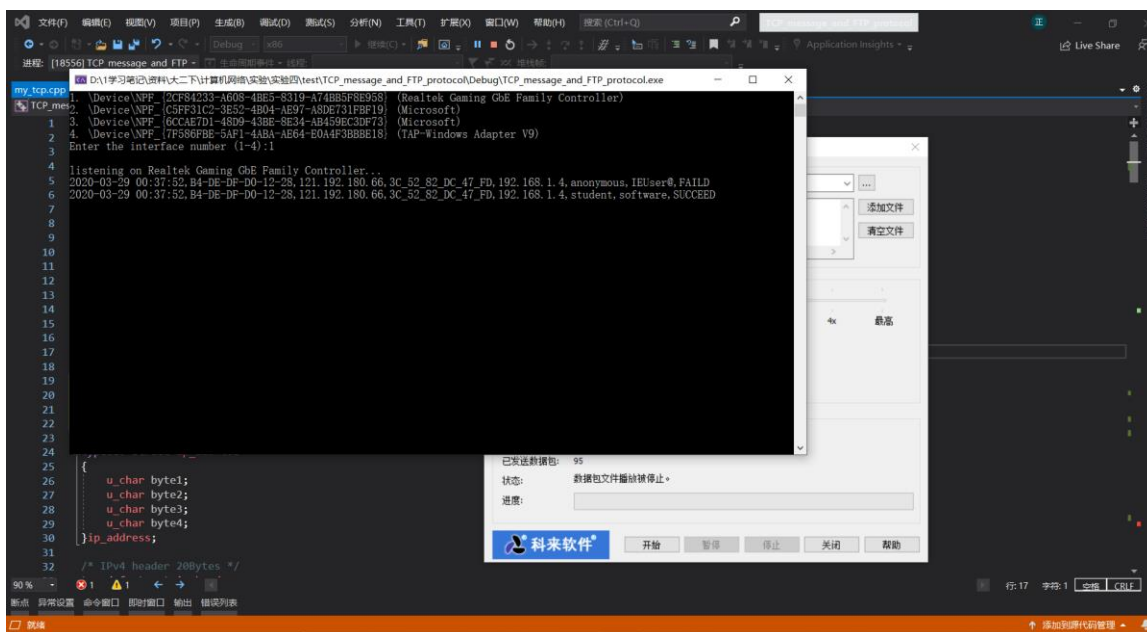
接下来使用学院 ftp

50	3.242636	192.168.1.4	121.192.180.66	TCP	66 7551 → 21 [SYN] Seq=0 Win=65535 Len=0 MSS=1440 WS=256 SACK_PERM=1
52	3.336177	121.192.180.66	192.168.1.4	TCP	66 21 → 7551 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1440 WS=256 SACK_PERM=1
53	3.336314	192.168.1.4	121.192.180.66	TCP	54 7551 → 21 [ACK] Seq=1 Ack=1 Win=262144 Len=0
54	3.430991	121.192.180.66	192.168.1.4	FTP	103 Response: 220 Serv-U FTP Server v6.2 for WinSock ready...

图示即为与 ftp 建立连接的三次握手过程，第 4 条即为 ftp 消息已建立连接

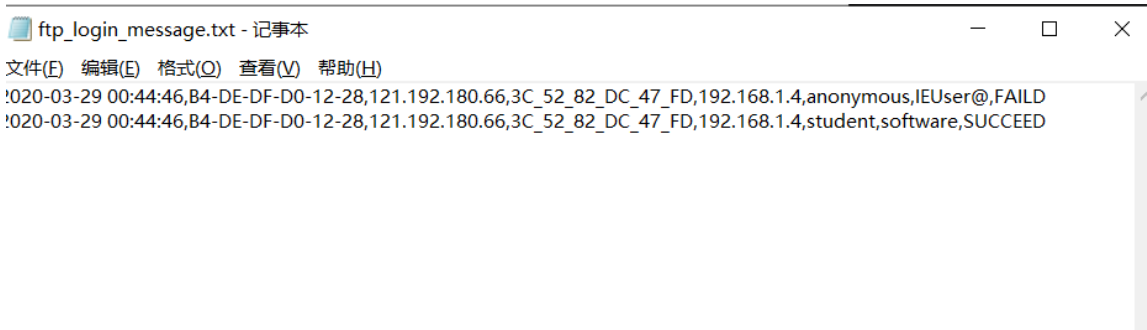
91	4.578902	192.168.1.4	121.192.180.66	TCP	54 7552 → 21 [FIN, ACK] Seq=49 Ack=168 Win=261888 Len=0
92	4.669576	121.192.180.66	192.168.1.4	TCP	60 21 → 7552 [ACK] Seq=168 Ack=50 Win=66048 Len=0
93	4.670291	121.192.180.66	192.168.1.4	TCP	60 21 → 7552 [FIN, ACK] Seq=168 Ack=50 Win=66048 Len=0
94	4.670342	192.168.1.4	121.192.180.66	TCP	54 7552 → 21 [ACK] Seq=50 Ack=169 Win=261888 Len=0

图示即为四次挥手与 ftp 服务器断开连接



在终端测试输出

第一次是匿名访问，由于学院 ftp 没有开启所以失败，第二次正确账户，口令成功连接



重定向将 CSV 格式日志输出到文件

4 实验总结

1. 学会了通过过滤器获得 TCP 信号的方法
2. 学习了 TCP 中的三次握手四次挥手的原理
3. 学习了 TCP 的窗口流量控制与拥塞控制方法

