

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 1 班

姓 名 周宇

学 号 24320182203335

实验时间 2020 年 3 月 31 日

2020 年 3 月 31 日

1 实验目的

侦听并观察 TCP 数据段和 FTP 数据

2 实验环境

Windows10 wireshark winpcap

3 实验结果

用 WireShark 侦听 TCP 数据段

TCP 的三次握手建立连接

1	0.000000	192.168.0.107	123.151.26.103	TCP	66 57802 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
2	0.042156	123.151.26.103	192.168.0.107	TCP	66 80 → 57802 [SYN, ACK] Seq=0 Ack=1 Win=13600 Len=0 MSS=1360 SACK_PERM=1
3	0.042225	192.168.0.107	123.151.26.103	TCP	54 57802 → 80 [ACK] Seq=1 Ack=1 Win=131840 Len=0

第一次握手：客户端发送一个 TCP，标志位为 SYN，序列号为 0

```

Transmission Control Protocol, Src Port: 57802, Dst Port: 80
  Source Port: 57802
  Destination Port: 80
  [Stream index: 0]
  [TCP Segment Len: 0]
  Sequence number: 0      (relative sequence number)
  Sequence number (raw): 3232399490
  [Next sequence number: 1      (relative sequence number)]
  Acknowledgment number: 0
  Acknowledgment number (raw): 0
  1000 .... = Header Length: 32 bytes (8)
  > Flags: 0x002 (SYN)
  
```

第二次握手：服务器发回确认包，标志位为 SYN,ACK. 将确认序号设置为客户端的 ISN 加 1 (0+1=1)

```

v Transmission Control Protocol, Src Port: 80
  Source Port: 80
  Destination Port: 57802
  [Stream index: 0]
  [TCP Segment Len: 0]
  Sequence number: 0    (relative sequence
  Sequence number (raw): 971680748
  [Next sequence number: 1    (relative seq
  Acknowledgment number: 1    (relative ack
  Acknowledgment number (raw): 3232399491
  1000 .... = Header Length: 32 bytes (8)
  > Flags: 0x012 (SYN, ACK)
  Window size value: 13600
  [Calculated window size: 13600]

```

第三次握手：客户端再次发送确认包 SYN 标志位为 0,ACK 标志位为 1。

```

v Transmission Control Protocol, Src Port: 57802
  Source Port: 57802
  Destination Port: 80
  [Stream index: 0]
  [TCP Segment Len: 0]
  Sequence number: 1    (relative sequence
  Sequence number (raw): 3232399491
  [Next sequence number: 1    (relative seq
  Acknowledgment number: 1    (relative ack
  Acknowledgment number (raw): 971680749
  0101 .... = Header Length: 20 bytes (5)
  > Flags: 0x010 (ACK)
  Window size value: 515
  [Calculated window size: 131840]

```

TCP 理论上四次挥手断开连接（只抓到 3 个，百度：因为服务器端在给客户端传回的过程中，将两个连续发送的包进行了合并）

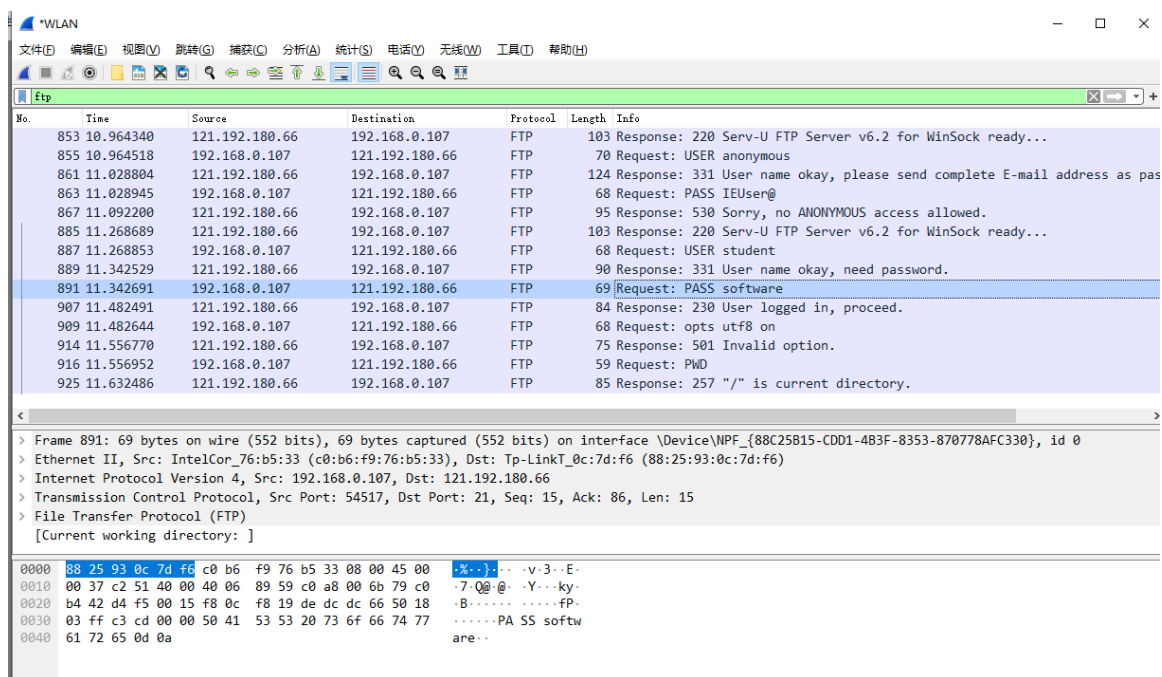
9.175420	192.168.0.107	123.151.26.103	TCP	54 57802 → 80 [FIN, ACK] Seq=744 Ack=2390 Win=1
9.216692	123.151.26.103	192.168.0.107	TCP	60 80 → 57802 [FIN, ACK] Seq=2390 Ack=745 Win=1
9.216745	192.168.0.107	123.151.26.103	TCP	54 57802 → 80 [ACK] Seq=745 Ack=2391 Win=131840

TCP 的窗口机制和拥塞控制机制：

168.0.107	113.24.195.10	TCP	54 49599 → 443 [ACK] Seq=33 Ack=33 Win=515 Len=0
151.26.103	192.168.0.107	TCP	60 80 → 49976 [ACK] Seq=1 Ack=731 Win=15360 Len=0
24.195.10	192.168.0.107	TCP	60 443 → 49599 [RST] Seq=32 Win=0 Len=0
24.195.10	192.168.0.107	TCP	60 443 → 49599 [RST] Seq=33 Win=0 Len=0
24.195.10	192.168.0.107	TCP	60 443 → 49599 [RST] Seq=32 Win=0 Len=0

零窗口暂停数据流，直到收到服务端的窗口更新，告知大小已经增加了才继续接收数据。

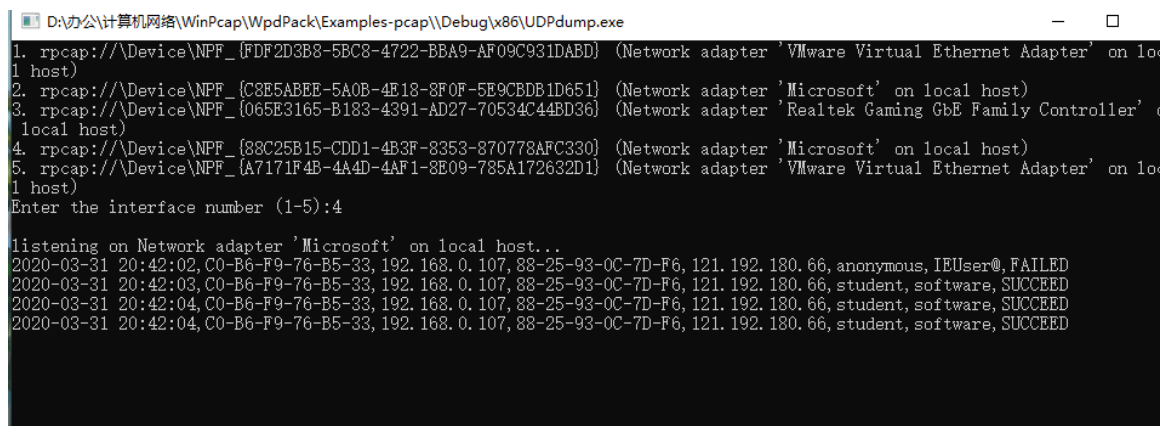
用 Wireshark 侦听 FTP 数据



用户名前为 USER，中间为用户名，后面为 0d0a（回车换行）

密码前为 PASS，中间为密码，后面为 0d0a

提取：匹配 USER/PASS 和 0d0a 之间的



4 实验总结

通过实验对 TCP 报文的握手协议以及滑动窗口机制和拥塞控制有了理解，了解到服务器以稳定速率传输的原理；对 FTP（学院）的侦听和分析，获取到了内容，观察得到了用户名和密码的获取方法及其他信息。搭建了本地的 FTP 但在设置用户名和密码的时候由于系统问题（不支持本地用户和角色的更改）导致无法侦听到信息。