

## 华东师范大学数据科学与工程学院实验报告

课程名称：计算机网络与编程

年级：2021 级

上机实践成绩：

指导教师：张召

姓名：彭一琄

学号：10215501412

上机实践名称：TCP 协议分析

上机实践日期：2023.5.19

上机实践编号：11

组号：

上机实践时间：9:50

### 一、实验目的

了解 TCP 协议的工作原理

学习 TCP 建立连接三次握手的过程

学习 TCP 断开连接四次挥手的过程

### 二、实验任务

使用 Wireshark 快速了解 TCP 协议

### 三、使用环境

Wireshark

Windows11

### 四、实验过程

task1: 利用 Wireshark 抓取一个 TCP 数据包，查看其具体数据结构和实际的数据（要求根据报文结构正确标识每个部分），请将实验结果附在实验报告中。

查看如下 tcp 协议包：

```
31 10.989257 172.30.240.107 223.119.158.171 TCP 66 61459 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256...
Transmission Control Protocol, Src Port: 61459, Dst Port: 80, Seq: 0, Len: 0
  Source Port: 61459
  Destination Port: 80
  [Stream index: 3]
  [Conversation completeness: Incomplete, DATA (15)]
  [TCP Segment Len: 0]
  Sequence Number: 0 (relative sequence number)
  Sequence Number (raw): 819942906
  [Next Sequence Number: 1 (relative sequence number)]
  Acknowledgment Number: 0
  Acknowledgment number (raw): 0
  1000 .... = Header Length: 32 bytes (8)
  > Flags: 0x002 (SYN)
    Window: 64240
    [Calculated window size: 64240]
    Checksum: 0x1ad4 [unverified]
    [Checksum Status: Unverified]
    Urgent Pointer: 0
  > Options: (12 bytes), Maximum segment size, No-Operation (NOP), Window scale, No-Operation (NOP), No-Operation (NOP), SACK permitted
  > [Timestamps]
```

Tcp 协议包的源端口号是 61459，目标端口号是 80.

相对序号是 0，序号是 819942906 (30 df 55 fa)

确认号是 0

首部长度值为 8，表示 Header 长度是 32 比特

```

v Flags: 0x002 (SYN)
  000. .... = Reserved: Not set
  ...0 .... = Accurate ECN: Not set
  .... 0... = Congestion Window Reduced: Not set
  .... .0.. = ECN-Echo: Not set
  .... ..0. = Urgent: Not set
  .... ...0 = Acknowledgment: Not set
  .... .... 0... = Push: Not set
  .... .... .0.. = Reset: Not set
> .... .... ..1. = Syn: Set
  .... .... ...0 = Fin: Not set
[TCP Flags: .....S.]
    
```

保留字有 4 位，包含 reserved、拥塞控制 Accurate ECN

CWR-拥塞窗口减小，与 ECE 标志都用于 IP 首部的 ECN 字段，ECE 标志为 1 时，则通知对方已将拥塞窗口缩小。此处设为 0

ECN-Echo-显式拥塞提醒回应。置为 1 会通知通信对方，从对方到这边的网络有拥塞。在收到数据包的 IP 首部中 ECN 为 1 时，将 TCP 首部中的 ECE 设置为 1。此处设为 0

URG-紧急指针，表明发送端向另一端使用紧急方式发送数据。包中有需要紧急处理的数据。此处设为 0 表示没有这种数据。

ACK-应答响应，表示确认序号有效。确认应答的字段有效。TCP 规定除了最初建立连接时的 SYN 包之外该位必须设置为 1。此时为第一次握手最初连接的包，设为 0

PUSH-推送，数据包立即发送。表示接收方应该尽快将这个报文交给应用层。表示需要将收到的数据立刻传给上层应用协议。PSH 为 0，也就是普通情况下，则不需要立即传，而是先进行缓存

RST-复位，中断一个连接，表示连接重置。表示 TCP 连接中出现异常必须强制断开连接。此时没有连接重置的需求。

SYN-同步，表示开始会话请求，用来发起一个连接，建立连接。SYN 为 1 表示希望建立连接，并在其序列号的字段进行序列号初始值的设定。

FIN-结束，结束会话，关闭连接。表示发送方完成任务，今后不会有数据发送，希望断开连接。当通信结束希望断开连接，通信双方的主机之间就可以相互交换 FIN 位置为 1 的 TCP 段。

接收窗口为 64240，校验和为 0x1ad4，由于没有紧急数据，紧急数据指针为 0

```

v Options: (12 bytes), Maximum segment size, No-Operation
  > TCP Option - Maximum segment size: 1460 bytes
  > TCP Option - No-Operation (NOP)
  > TCP Option - Window scale: 8 (multiply by 256)
  > TCP Option - No-Operation (NOP)
  > TCP Option - No-Operation (NOP)
  > TCP Option - SACK permitted
    
```

选项的内容包括：

最大报文段长度为 1460 字节，窗口扩大因子为 8，允许 SACK

task2: 根据 TCP 三次握手的交互图和抓到的 TCP 报文详细分析三次握手过程，请将实

验结果附在实验报告中。

第一次握手：

```

v Transmission Control Protocol, Src Port: 61459, Dst Port: 80, Seq: 0, Len: 0
  Source Port: 61459
  Destination Port: 80
  [Stream index: 3]
  [Conversation completeness: Incomplete, DATA (15)]
  [TCP Segment Len: 0]      第一次握手序列号为0
  Sequence Number: 0 (relative sequence number)
  Sequence Number (raw): 819942906
  [Next Sequence Number: 1 (relative sequence number)]
  Acknowledgment Number: 0
  Acknowledgment number (raw): 0
  1000 .... = Header Length: 32 bytes (8)
v Flags: 0x002 (SYN)
  000. .... = Reserved: Not set
  ...0 .... = Accurate ECN: Not set
  .... 0... = Congestion Window Reduced: Not set
  .... .0.. = ECN-Echo: Not set
  .... ..0. = Urgent: Not set
  .... ...0 .... = Acknowledgment: Not set
  .... .... 0... = Push: Not set 发起同步请求
  .... .... .0.. = Reset: Not set
> .... .... .1. = Syn: Set
  .... .... ...0 = Fin: Not set
  [TCP Flags: .....S.]

```

第二次握手：ACK=第一次握手的序列号+1，创建一个序列号。

```

v Transmission Control Protocol, Src Port: 80, Dst Port: 61459, Seq: 0, Ack: 1, Len: 0
  Source Port: 80
  Destination Port: 61459
  [Stream index: 3]
  [Conversation completeness: Incomplete, DATA (15)]
  [TCP Segment Len: 0] 初始序列号0
  Sequence Number: 0 (relative sequence number)
  Sequence Number (raw): 1323134704
  [Next Sequence Number: 1 (relative sequence number)] 确认号ACK=1
  Acknowledgment Number: 1 (relative ack number)
  Acknowledgment number (raw): 819942907
  1000 .... = Header Length: 32 bytes (8)
v Flags: 0x012 (SYN, ACK)
  000. .... = Reserved: Not set
  ...0 .... = Accurate ECN: Not set
  .... 0... = Congestion Window Reduced: Not set
  .... .0.. = ECN-Echo: Not set
  .... ..0. = Urgent: Not set
  .... .. 1 .... = Acknowledgment: Set
  .... .... 0... = Push: Not set
  .... .... .0.. = Reset: Not set
  > .... .... .1. = Syn: Set
  .... .... ...0 = Fin: Not set
  [TCP Flags: .....A..S.]
  
```

第三次握手: ACK=第二次握手的序列号+1, 序列号=第一次握手的序列号

```

v Transmission Control Protocol, Src Port: 61459, Dst Port: 80, Seq: 1, Ack: 1, Len: 0
  Source Port: 61459
  Destination Port: 80
  [Stream index: 3]
  [Conversation completeness: Incomplete, DATA (15)]
  [TCP Segment Len: 0]
  Sequence Number: 1 (relative sequence number)
  Sequence Number (raw): 819942907
  [Next Sequence Number: 1 (relative sequence number)]
  Acknowledgment Number: 1 (relative ack number)
  Acknowledgment number (raw): 1323134705
  0101 .... = Header Length: 20 bytes (5)
v Flags: 0x010 (ACK)
  000. .... = Reserved: Not set
  ...0 .... = Accurate ECN: Not set
  .... 0... = Congestion Window Reduced: Not set
  .... .0.. = ECN-Echo: Not set
  .... ..0. = Urgent: Not set
  .... .. 1 .... = Acknowledgment: Set
  .... .... 0... = Push: Not set
  .... .... .0.. = Reset: Not set
  .... .... ..0. = Syn: Not set
  .... .... ...0 = Fin: Not set
  [TCP Flags: .....A.....]
  
```

task3: 根据 TCP 四次挥手的交互图和抓到的 TCP 报文详细分析四次挥手过程, 请将实验结果附在实验报告中。

第一次挥手：发送 FIN=1, ACK=1, 序列号=3867, ACK 的序列号=374

```
Source Port: 80
Destination Port: 62289
[Stream index: 40]
[Conversation completeness: Complete, WITH_DATA (31)]
[TCP Segment Len: 0]
Sequence Number: 3867 (relative sequence number)
Sequence Number (raw): 4157661109
[Next Sequence Number: 3868 (relative sequence number)]
Acknowledgment Number: 374 (relative ack number)
Acknowledgment number (raw): 651791074
0101 .... = Header Length: 20 bytes (5)
Flags: 0x011 (FIN, ACK)
 000. .... = Reserved: Not set
...0 .... = Accurate ECN: Not set
.... 0... = Congestion Window Reduced: Not set
.... .0.. = ECN-Echo: Not set
.... ..0. = Urgent: Not set
.... ...1 = Acknowledgment: Set
.... .... 0... = Push: Not set
.... .... .0.. = Reset: Not set
.... .... ..0. = Syn: Not set
> .... .... ..1 = Fin: Set
> [TCP Flags: .....A....F]
```

第二次挥手：被动关闭方回复 ACK=1, 序列号=第一次挥手的 ACK 序列号, ACK 序列号=第一次挥手的序列号+1

```
Destination Port: 80
[Stream index: 40]
[Conversation completeness: Complete, WITH_DATA (31)]
[TCP Segment Len: 0]
Sequence Number: 374 (relative sequence number)
Sequence Number (raw): 651791074
[Next Sequence Number: 374 (relative sequence number)]
Acknowledgment Number: 3868 (relative ack number)
Acknowledgment number (raw): 4157661110
0101 .... = Header Length: 20 bytes (5)
Flags: 0x010 (ACK)
 000. .... = Reserved: Not set
...0 .... = Accurate ECN: Not set
.... 0... = Congestion Window Reduced: Not set
.... .0.. = ECN-Echo: Not set
.... ..0. = Urgent: Not set
.... ...1 = Acknowledgment: Set
.... .... 0... = Push: Not set
.... .... .0.. = Reset: Not set
.... .... ..0. = Syn: Not set
.... .... ...0 = Fin: Not set
[TCP Flags: .....A....]
```

第三次挥手：被动关闭方发送 FIN=1, ACK=1, 序列号=第二次挥手时的序列号, ACK 序

列号=第二次挥手时的 ACK 序列号。

```

Destination Port: 80
[Stream index: 40]
[Conversation completeness: Complete, WITH_DATA (31)]
[TCP Segment Len: 0]
Sequence Number: 374      (relative sequence number)
Sequence Number (raw): 651791074
[Next Sequence Number: 375      (relative sequence number)]
Acknowledgment Number: 3868      (relative ack number)
Acknowledgment number (raw): 4157661110
0101 .... = Header Length: 20 bytes (5)
Flags: 0x011 (FIN, ACK)
  000. .... = Reserved: Not set
  ...0 .... = Accurate ECN: Not set
  .... 0... = Congestion Window Reduced: Not set
  .... .0.. = ECN-Echo: Not set
  .... ..0. = Urgent: Not set
  .... ...1 = Acknowledgment: Set
  .... .... 0... = Push: Not set
  .... .... .0.. = Reset: Not set
  .... .... ..0. = Syn: Not set
> .... .... ...1 = Fin: Set
> [TCP Flags: .....A...F]

```

第四次挥手: ACK=1, 序列号=第三次挥手时的序列号+1, ACK 序列号=第三次挥手时的 ACK 序列号

```

Destination Port: 62289
[Stream index: 40]
[Conversation completeness: Complete, WITH_DATA (31)]
[TCP Segment Len: 0]
Sequence Number: 3868      (relative sequence number)
Sequence Number (raw): 4157661110
[Next Sequence Number: 3868      (relative sequence number)]
Acknowledgment Number: 375      (relative ack number)
Acknowledgment number (raw): 651791075
0101 .... = Header Length: 20 bytes (5)
Flags: 0x010 (ACK)
  000. .... = Reserved: Not set
  ...0 .... = Accurate ECN: Not set
  .... 0... = Congestion Window Reduced: Not set
  .... .0.. = ECN-Echo: Not set
  .... ..0. = Urgent: Not set
  .... ...1 = Acknowledgment: Set
  .... .... 0... = Push: Not set
  .... .... .0.. = Reset: Not set
  .... .... ..0. = Syn: Not set
  .... .... ...0 = Fin: Not set
[TCP Flags: .....A.....]

```

## 五、总结

本次实验的主要目的是了解 TCP 协议的工作原理，学习 TCP 建立连接三次握手的过程，学习 TCP 断开连接四次挥手的过程。通过使用 WireShark 抓包分析 TCP 报文的结构和每一部分的内容，我们可以更深入地了解 TCP 协议的工作原理和实现细节。在实验过程中，我们学习了 TCP 连接的建立和断开过程，包括三次握手和四次挥手的具体步骤和流程。通过实验，我们深入了解了 TCP 协议的工作原理和实现细节，为我们更好地理解和应用 TCP 协议提供了基础。