实验 1.2　TCP 协议分析

09021227 金桥

2023 年 11 月 21 日

1　实验内容

TCP（Transmission Control Protocol 传输控制协议）是一种面向连接的、可靠的、基于字节流的传输层通信协议。本实验通过运用 Wireshark 对网络活动进行分析，观察 TCP 协议报文，分析通信时序，理解 TCP 的工作过程，掌握 TCP 工作原理与实现；学会运用 Wireshark 分析 TCP 连接管理、流量控制和拥塞控制的过程，发现 TCP 的性能问题。

- 观察 TCP 三次握手与四次挥手报文，注意报文收发过程中，双方 TCP 状态的变化。
- 以本次捕获的报文为依据，分别画出本次 TCP 连接三次握手与四次挥手的时序图，结合 TCP 状态机，在双方各阶段标出对应的 TCP 状态。
- 选择其中一个 TCP 报文，配合 Wireshark 截图，分析其 TCP 首部各字段定义、值及其含义。

两台实验机本地相互连接，在实验机中仿真不同的网络条件，以便观察 TCP 的各种控制现象。

- 观察 TCP 三次握手与四次挥手报文，注意报文收发过程中，双方 TCP 状态的变化。
- 以本次捕获的报文为依据，分别画出本次 TCP 连接三次握手与四次挥手的时序图，结合 TCP 状态机，在双方各阶段标出对应的 TCP 状态。
- 选择其中一个 TCP 报文，配合 Wireshark 截图，分析其 TCP 首部各字段定义、值及其含义。

2　实验过程

2.1　观察 TCP 连接三次握手与四次挥手

- 打开 Wireshark，在终端中输入 `curl www.bing.com`，使用 curl 访问必应.
- 在 Wireshark 中输入 `http`，过滤出对应分组，右键追踪流选择 TCP. 如图 1.
- 可以看到，前三条记录为三次握手，最后四条记录为四次挥手。

No.	Time	Source	Destination	Protocol	Length	Info
17	1.016529	192.168.1.218	202.89.233.100	TCP	66	59444 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
18	1.042560	202.89.233.100	192.168.1.218	TCP	66	80 → 59444 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1440 WS=256 SACK_PERM
19	1.042660	192.168.1.218	202.89.233.100	TCP	54	59444 → 80 [ACK] Seq=1 Ack=1 Win=132352 Len=0
20	1.042822	192.168.1.218	202.89.233.100	HTTP	129	GET / HTTP/1.1
21	1.069144	202.89.233.100	192.168.1.218	TCP	60	80 → 59444 [ACK] Seq=1 Ack=76 Win=4194304 Len=0
22	1.071960	202.89.233.100	192.168.1.218	HTTP	277	HTTP/1.1 301 Moved Permanently
23	1.072482	192.168.1.218	202.89.233.100	TCP	54	59444 → 80 [FIN, ACK] Seq=76 Ack=224 Win=132096 Len=0
24	1.101827	202.89.233.100	192.168.1.218	TCP	54	80 → 59444 [ACK] Seq=224 Ack=77 Win=4194304 Len=0
25	1.101827	202.89.233.100	192.168.1.218	TCP	54	80 → 59444 [FIN, ACK] Seq=224 Ack=77 Win=4194304 Len=0
26	1.101907	192.168.1.218	202.89.233.100	TCP	54	59444 → 80 [ACK] Seq=77 Ack=225 Win=132096 Len=0

图 1: TCP 分组。图中前三条记录为 TCP 三次握手，最后四条记录为 TCP 四次挥手.

2.2 TCP 连接三次握手与四次挥手时序图

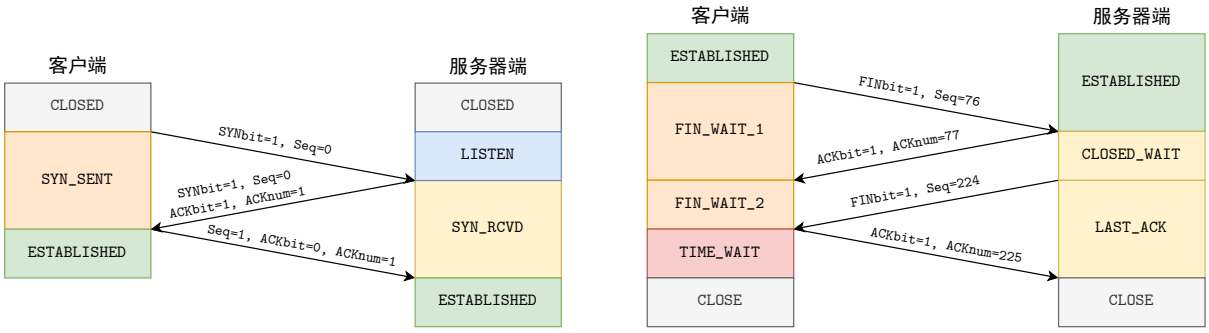

图 2: 左侧为三次握手时序图，右侧为四次挥手时序图

2.3 分析 TCP 首部

选择其中一个 TCP 报文，其 Wireshark 截图如下：

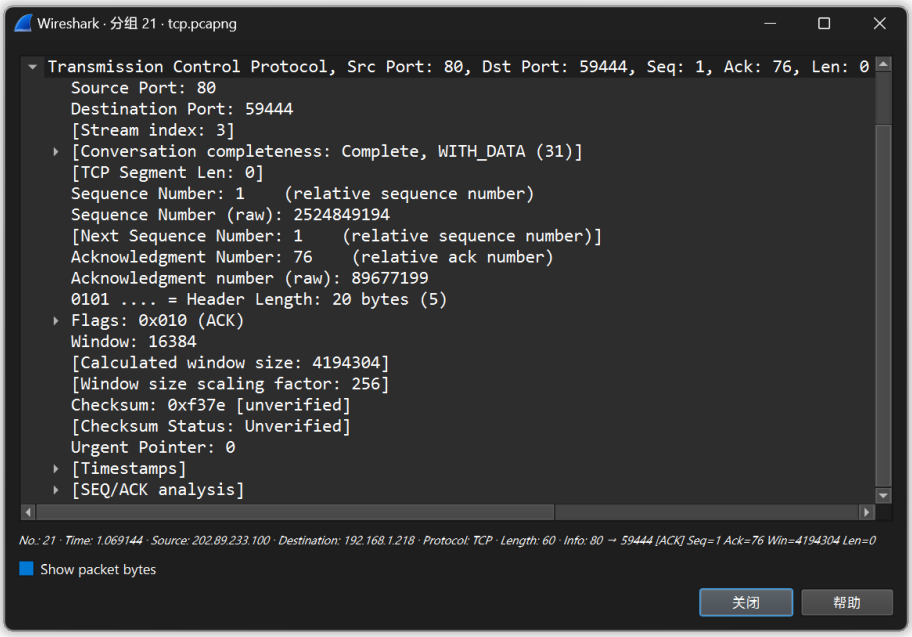

图 3: 其中一个 TCP 报文首部截图。

各字段定义、值及其含义如下：

- `Source Port` 源端口地址，值为 80
- `Destination Port` 目的端口地址，值为 59444
- `Sequence Number` 序号，实际序号为 2524849194，Wireshark 简化为 1
- `Acknowledgment Number` 应答序号，实际应答序号为 89677199，Wireshark 简化为 76
- `Header Length` 首部长度，值为 0101，即为 $5 \times 4 = 20$ 字节
- `Flags` 标志位，只有 ACK 对应位被置 1
- `Window` 接收端还可以接收的字节数，值为 16384
- `Checksum` 校验和，值为 `0xf37e`
- `Urgent Pointer` 紧急指针，为 0 无效

2.4 扩展实验

- 使用 VMware 在局域网内启动一个虚拟机，设置丢包率为 50%
- 通过以下指令：`python -m https.server 80` 使用 Python 启动一个 HTTP 服务器。
- 启动 Wireshark 进行记录，同时访问服务器，Wireshark 截图如图 4

No.	Time	Source	Destination	Protocol	Length	Info
3163	93.102520	192.168.1.218	192.168.1.73	TCP	66	53708 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
3171	94.107826	192.168.1.218	192.168.1.73	TCP	66	[TCP Retransmission] 53708 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
3172	94.109417	192.168.1.73	192.168.1.218	TCP	66	80 → 53708 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM WS=128
3173	94.109520	192.168.1.218	192.168.1.73	TCP	54	53708 → 80 [ACK] Seq=1 Ack=1 Win=65536 Len=0
3223	100.038114	192.168.1.218	192.168.1.73	HTTP	584	GET /%E8%80%83%E7%BC%96%E7%A7%98%E7%B1%8D HTTP/1.1
3224	100.039942	192.168.1.73	192.168.1.218	HTTP	253	HTTP/1.0 200 OK
3225	100.045436	192.168.1.218	192.168.1.73	TCP	54	53708 → 80 [FIN, ACK] Seq=531 Ack=200 Win=65280 Len=0
3226	100.046549	192.168.1.73	192.168.1.218	TCP	60	[TCP Previous segment not captured] 80 → 53708 [ACK] Seq=201 Ack=532 Win=64128 Len=0
3229	100.466636	192.168.1.73	192.168.1.218	TCP	60	[TCP Retransmission] 80 → 53708 [FIN, ACK] Seq=200 Ack=532 Win=64128 Len=0
3230	100.466698	192.168.1.218	192.168.1.73	TCP	54	53708 → 80 [ACK] Seq=532 Ack=201 Win=65280 Len=0

图 4: 设置丢包率为 50% 的 TCP 分组。图中第 2-4 条记录为 TCP 三次握手，最后四条记录为 TCP 四次挥手. 可以看到与图 1 不同，握手与挥手存在丢包重传的记录（黑色）

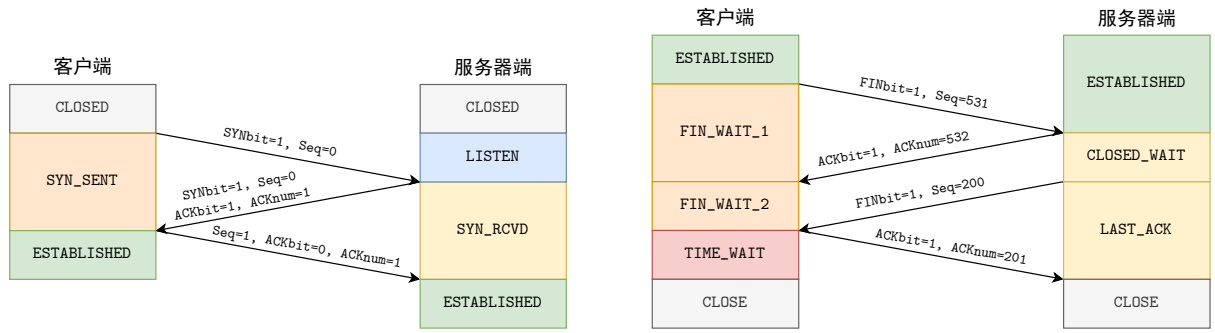

图 5: 设置丢包率为 50% 的时序图。左侧为三次握手时序图，右侧为四次挥手时序图（重传未画出）

选择其中一个 TCP 报文，其 Wireshark 截图如图 6

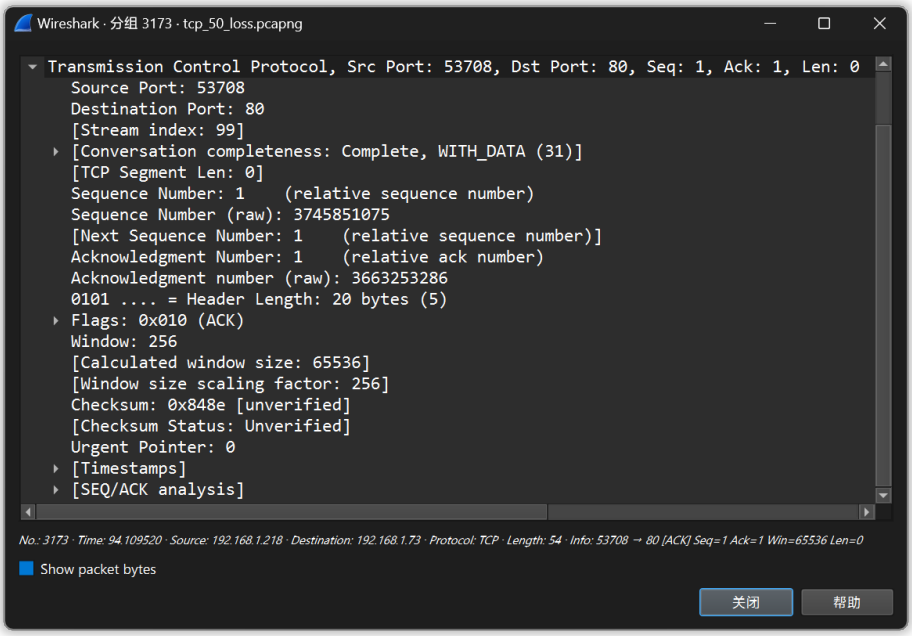

图 6: 其中一个 TCP 报文首部截图。

各字段定义、值及其含义如下：

- `Source Port` 源端口地址，值为 53708
- `Destination Port` 目的端口地址，值为 80
- `Sequence Number` 序号，实际序号为 3745851075，Wireshark 简化为 1
- `Acknowledgment Number` 应答序号，实际应答序号为 3663253286，Wireshark 简化为 1
- `Header Length` 首部长度，值为 0101，即为 $5 \times 4 = 20$ 字节
- `Flags` 标志位，只有 ACK 对应位被置 1
- `Window` 接收端还可以接收的字节数，值为 256
- `Checksum` 校验和，值为 `0x848e`
- `Urgent Pointer` 紧急指针，为 0 无效

3 实验体会

通过这次实验，我学习了如何使用 Wireshark 观察 TCP 连接。对于 TCP 三次握手与四次挥手过程有了更深刻的认识。