中国科学技术大学计算机学院

计算机网络实验报告

实验三

利用 Wireshark 观察 http 报文

学　　号：PB17111568
姓　　名：郭雨轩
专　　业：计算机科学与技术
指导老师：张信明

中国科学技术大学计算机学院
2019 年 11 月 30 日

一、 实验目的

1、捕获从计算机到远程服务器的大量TCP传输；
2、根据获得的跟踪结果对TCP传输机制作一些必要的分析，加深对TCP协议的理解；

二、 实验原理

Wireshark（前称 Ethereal）是一个网络封包分析软件。网络封包分析软件的功能是抓取网络封包,并尽可能显示出最为详细的网络封包资料。Wireshark 使用 WinPCAP 作为接口，直接与网卡进行数据报文交换，监听共享网络上传送的数据包，并不能对其进行修改或者控制。

本实验使用 Wireshark 抓取 Chrome 浏览器的在访问网页时发送和接收的数据包，对其进行分析。

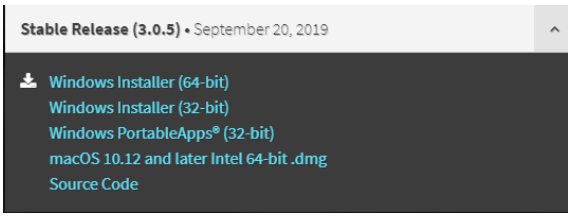
三、 实验环境

软件：Windows10-64bit，Wireshark，Chrome.
硬件：Intel Corei5-7300HQ，NVIDIA GTX1050Ti，16GiB RAM.

四、 实验过程

1、 Wireshark 安装

1） 访问 wireshark.org 得到了 wireshark 安装包



2） 双击打开 wireshark 安装包即可完成安装

2、 实验过程

1） 问题一：

客户端 IP：192.168.43.238

客户端端口号：Source Port: 56481

2） 问题二：

服务器 IP：128.119.245.12

服务器端口号：Destination Port: 80

3）问题三：

客户端 IP：192.168.1.102

客户端端口号：Source Port: 1161

4）问题四：

初始化连接的序列号：Sequence number: 0

哪部分决定是 SYN：

```
Flags: 0x002 (SYN)
    000. .... .... = Reserved: Not set
    ...0 .... .... = Nonce: Not set
    .... 0... .... = Congestion Window Reduced (CWR): 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
```

5）问题五：

序列号：Sequence number: 0

ACK：Acknowledgment number: 1 ，根据客户端的序列号加 1 确定

哪部分决定是 SYN：

```
Flags: 0x012 (SYN, ACK)
  000. .... .... = Reserved: Not set
  ...0 .... .... = Nonce: Not set
  .... 0... .... = Congestion Window Reduced (CWR): 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
```

6）问题六

POST 序列号：Sequence number: 1

7）问题七

NO.	SEQ	Send time	ACK time	RTT	E-RTT
1	1	0.026477	0.053937	0.02746	0.02746
2	566	0.041737	0.077294	0.035557	0.0285
3	2026	0.054026	0.124085	0.070059	0.0337
4	3486	0.054690	0.169118	0.11443	0.0438
5	4946	0.077405	0.217299	0.13989	0.0558
6	6406	0.078157	0.267802	0.18964	0.0725

8）问题八

NO	Length

1	565
2	1460
3	1460
4	1460
5	1460
6	1460

9） 问题九

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.1.102	128.119.245.12	TCP	62	1161 → 80 [SYN] Seq=0 Win=16384 Len=0 MSS=
2	0.023172	128.119.245.12	192.168.1.102	TCP	62	80 → 1161 [SYN, ACK] Seq=0 Ack=1 Win=5840

在整个发送过程中，只有在 No.2 报文时的 Win 最小为 5480

10） 问题十

没有，因为 ACK 单调递增

11） 问题十一

一个 ACK 通常确认 1460 的数据，

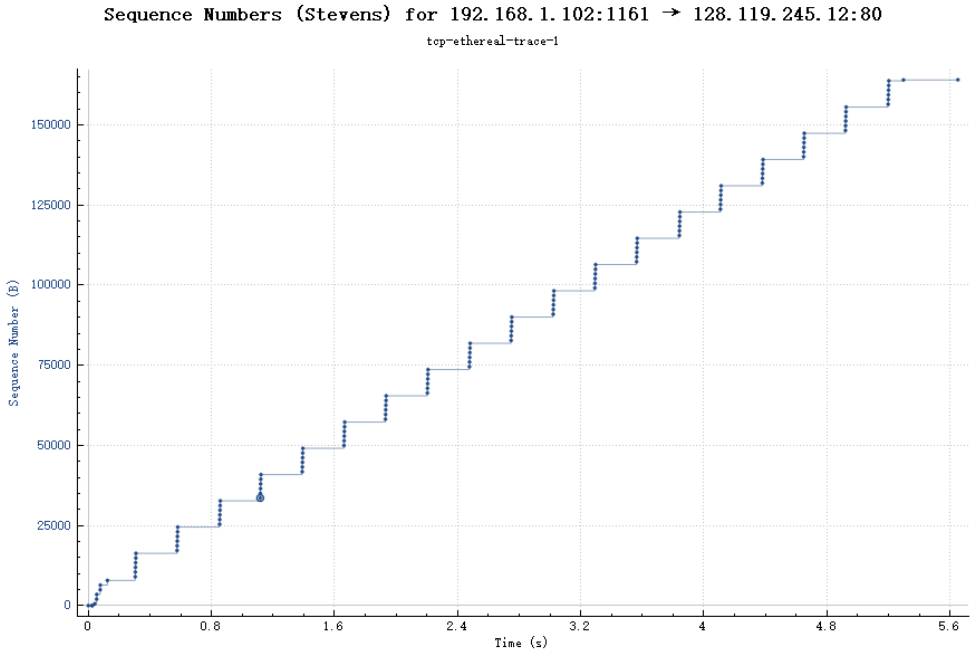
78	1.758227	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=52893 Win=62780 Len=0
79	1.860063	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=55813 Win=62780 Len=0

ACK 跨度为 2920，正在压缩其他段。

12） 问题十二

吞吐量 = (164091-1)/(5.45583- 0.026477) = 30.222 (kB/s)

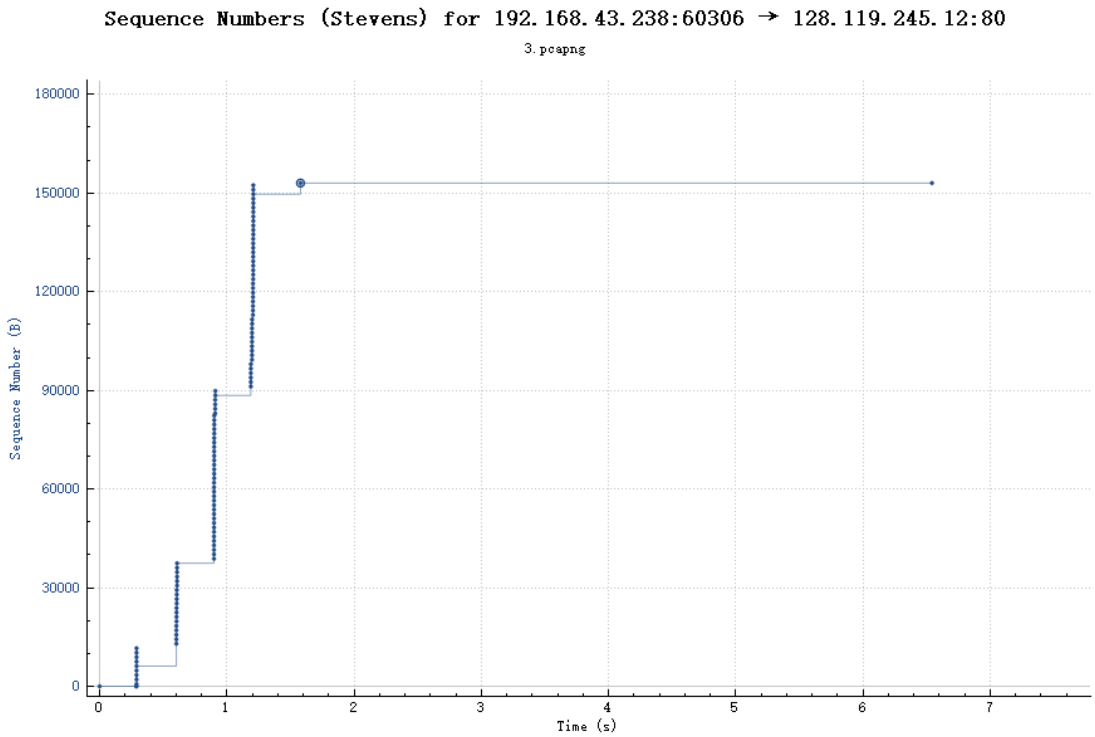
13） 问题十三



只在最开始的一小部分时慢启动，之后进入拥塞避免状态，发送速度一直平稳不变，与课本上讲的有较大出入。

这个图与课本的出入在于，当结束慢启动之后，就一直维持恒定的发送速率进行发送，不会有每过一个周期加 1 这个操作。

14） 问题十四



在我的发送过程中，一直进行慢启动直到文件发送完成也没有结束慢启动。至少这个过程与课本上讲得慢启动基本一致，至于拥塞避免状态的行为和快速重传的行为则不确定。

五、 实验总结

1） 实验收获：

a) 我熟悉了 Wireshark 的使用，学会了通过 Wireshark 的统计数据获得数据包发送的情况。

b) 通过阅读 TCP 报文，我对连接建立和拆除的行为有了更加直观的认识，同时根据得到的统计图，我对真实的 TCP 的行为也有了了解。