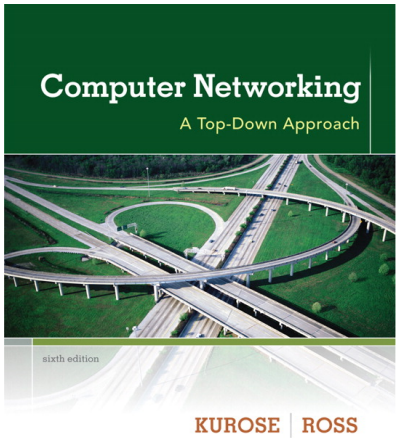
Wireshark 实验: TCP v6.0



Computer Networking: A Top-Down Approach, 6th ed.,
J.F. Kurose and K.W. Ross

"不闻不若闻之，闻之不若见之；见之不若知之，知之不若行之；学至于行而止矣。"

——《荀子•儒效篇》

翻译：马可, 宋涛
审校：秦大力

在这个实验中，我们将详细研究 TCP 协议。本实验将从你的计算机传输一个 150KB 的文件（包含了 Lewis Carrol 的《爱丽丝梦游仙境》的文本）到远程服务器，并研究这一过程中发送和接收到的 TCP 段。 我们将研究 TCP 对序列和确认号的使用，以提供可靠的数据传输；我们还将研究 TCP 拥塞控制算法（慢启动和拥塞避免算法）和 TCP 流量控制机制；最后，我们还将研究简化的 TCP 连接建立过程，并分析你的计算机与服务器之间 TCP 连接的性能（如吞吐量、RTT 等）。

在本实验开始之前，请复习一下教材的 3.5 和 3.7 节。

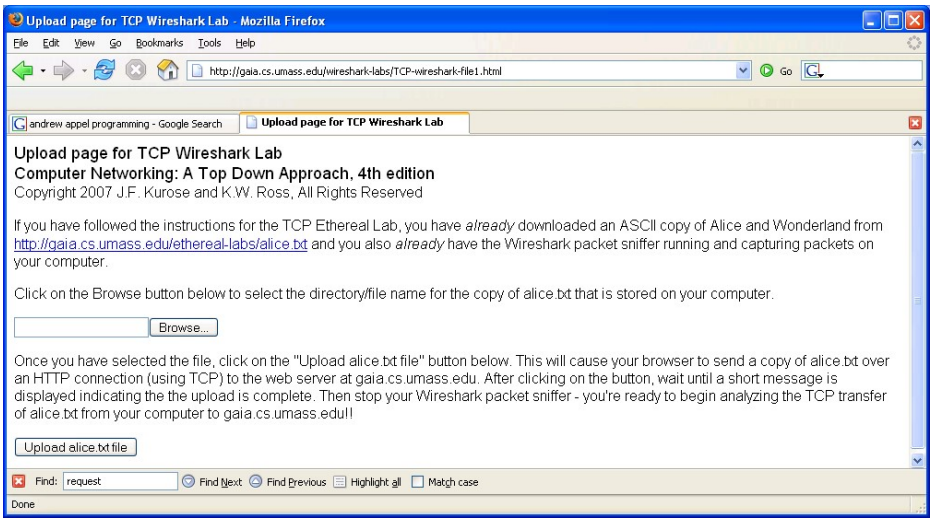
一、 捕获从本机到远端服务器之间的批量 TCP 传输

在开始探索 TCP 之前，我们需要使用 Wireshark 来捕获从本机到服务器的 TCP 传输数据包。 你可以通过访问一个网页来进行访问，该网页将允许你输入存储在计算机上的文件的名称（其中包含《爱丽丝梦游仙境》的 ASCII 文本），然后使用 HTTP POST 方法（见教材的 2.2.3 节）将文件传输到 Web 服务器。由于我们希望将大量数据从计算机传输到另一台计算机，因而使用了 POST 方法而不是 GET 方法。注意，在上传文件时我们才运行 Wireshark，从而获取从本机发送和接收的 TCP 报文段。

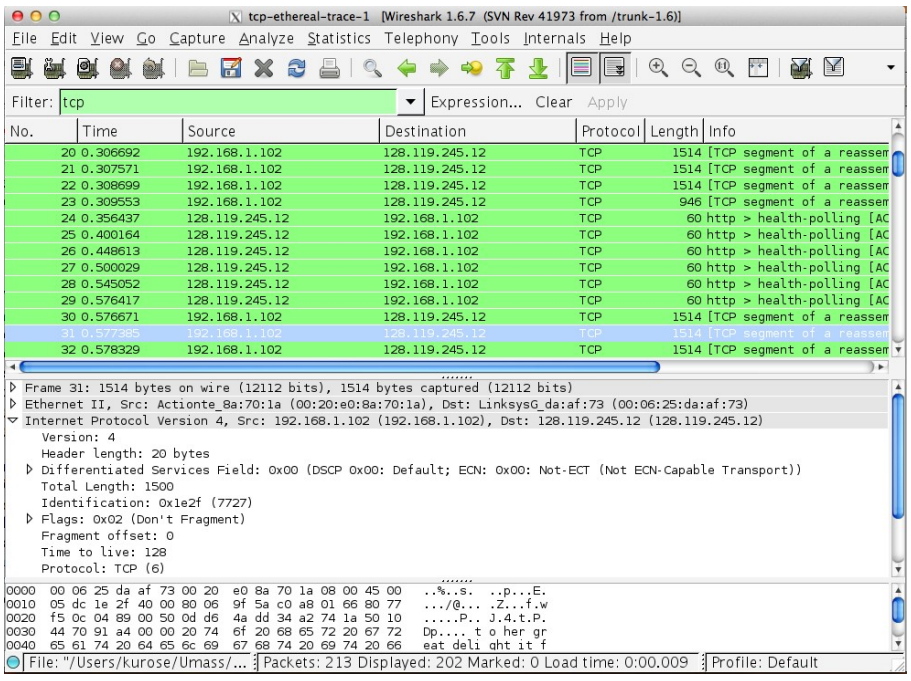
请执行以下操作：

- 启动 Web 浏览器。
- 转到 http://gaia.cs.umass.edu/wireshark-labs/alice.txt，并获得《爱丽丝梦游仙境》的 ASCII 副本，然后将此文件存储在计算机上的某个位置。.

- 接下来访问 http://gaia.cs.umass.edu/wireshark-labs/TCP-wireshark-file1.html，你将看到以下页面：



- 现在，你的计算机上应该存在一个包含《爱丽丝梦游仙境》的文件，请使用此表单中的“Browser”按钮给出该文件的完整路径。此时先不要点击“*Upload alice.txt file*”按钮。
- 启动 Wireshark 并开始捕获（*Capture* → *Start*），然后在 Wireshark 抓包选项对话框上点击 OK（我们不需要在此处选择任何选项）。
- 返回到浏览器，点击“*Upload alice.txt file*”按钮将文件上传到服务器 gaia.cs.umass.edu。文件上传后，浏览器窗口将显示一条简短的祝贺信息。
- 停止捕获。Wireshark 窗口应类似于下面所示：



如果无法在实时网络连接上运行 Wireshark，则可以下载数据包跟踪文件（该文件是在作者的计算机上按照上述步骤捕获的）。即使你使用自己的跟踪文件时，也可能会发现下载上述跟踪文件很有参考价值。

二、 查看捕获的跟踪文件

在具体分析 TCP 连接的行为之前，我们先来仔细研究一下跟踪文件。

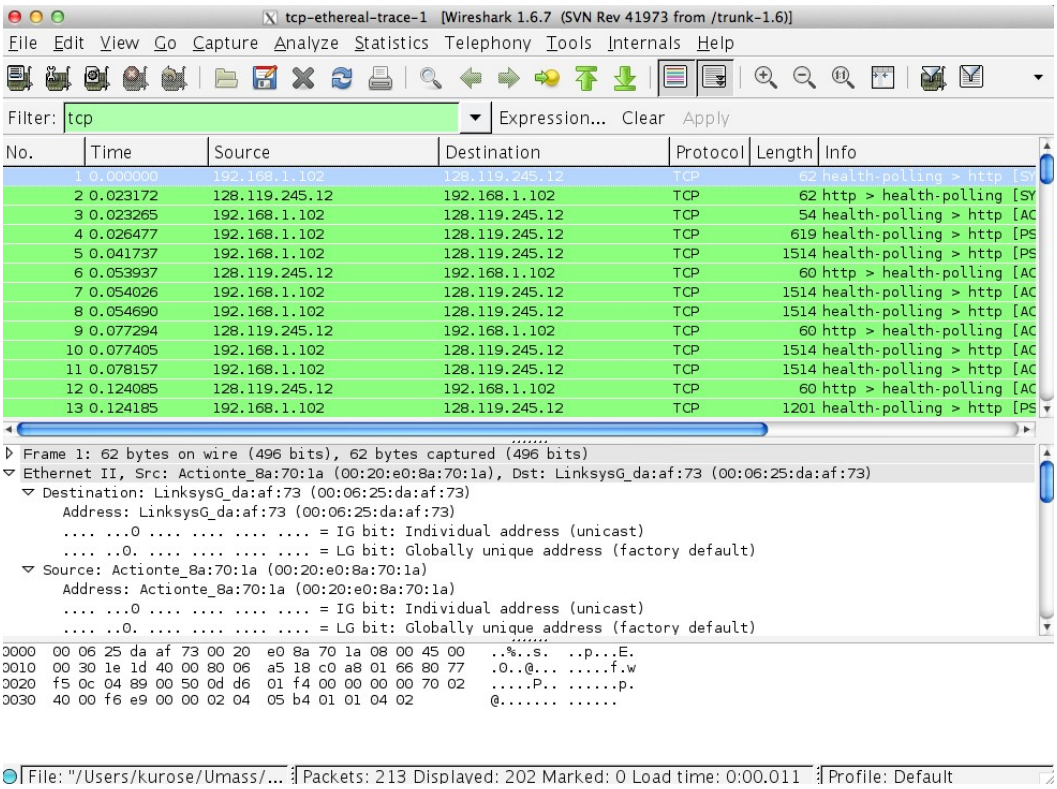
- 首先，在 Wireshark 过滤器中输入“tcp”（小写且无引号，不要忘记在输入后点击“Apply”或回车！）过滤 Wireshark 窗口中显示的数据包。

你应该看到的是本机与服务器 gaia.cs.umass.edu 之间的一系列 TCP 和 HTTP 报文，包括包含 SYN 报文的 TCP 三次握手初始化过程以及 HTTP POST 报文。你可能会看到一系列从本机发送到 gaia.cs.umass.edu 的“HTTP Continuation”报文（Wireshark 的不同版本会略有差异）。回想一下，在前面的 HTTP 实验中，我们已经知道这并不是一个所谓 “连续”的报文，而是 Wireshark 的一种表达方式，用来指示有多个 TCP 段参与了单个 HTTP 报文的传输。在最新版本的 Wireshark 中，你会在 Wireshark 的 Info 列中看到“[TCP segment of a reassembled PDU]”，以表明此 TCP 段包含属于上层协议消息的数据（在本例中为 HTTP 报文）。你还应该看到从 gaia.cs.umass.edu 返回到你的计算机的 TCP ACK 报文段。

使用 http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip 中的跟踪文件 tcp-ethereal-trace-1 来回答问题 1 和问题 2。只要有可能，在回答问题时，可以将跟踪文件中的数据包信息打印输出，使用彩色笔进行标记，并注释说明你的答案。要打印数据包信息，请使用 Wireshark 菜单项“*File->Print*”，并选择 *Selected packet only* 和 *Packet summary line*，打印出回答下列问题所需的信息。如果你使用自己的跟踪文件，请回答问题 3，否则，请回答问题 1。

1. 将文件传输到 gaia.cs.umass.edu 的客户端计算机（源）使用的 IP 地址和 TCP 端口号是多少？为了回答这个问题，可以使用“details of the selected packet header window”来选择 HTTP 报文，并分析携带此 HTTP 报文的 TCP 段的详细信息。
2. gaia.cs.umass.edu 的 IP 地址是什么？在此连接上发送和接收 TCP 段的端口号是什么？
3. 你的客户端计算机（源）将文件传输到 gaia.cs.umass.edu 时使用的 IP 地址和 TCP 端口号是多少？

由于本实验室是关于 TCP 而不是 HTTP 协议，因此我们先修改一下 Wireshark 的“listing of captured packets”窗口，以便显示包含 HTTP 报文的 TCP 段信息，而不是显示 HTTP 报文。要使 Wireshark 执行此操作，请选择 *Analyze* → E*nabled Protocols*，取消选中 HTTP 复选框，然后选择确定。你将看到的 Wireshark 窗口如下所示：



这正是我们希望看到的：在你的计算机和 gaia.cs.umass.edu 之间发送的一系列 TCP 段。我们将使用你捕获的数据包跟踪文件来研究 TCP 的行为（也可以使用教材作者提供的压缩文件 gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip 中的跟踪文件 tcp-ethereal-trace-1，如何提取该文件请参阅 HTTP 实验文档中的脚注）。

三、 TCP 基础

回答以下关于 TCP 段的问题：

4. 用于启动客户端计算机和 gaia.cs.umass.edu 之间 TCP 连接的 TCP SYN 报文段的序号是什么？在 TCP 段中，是什么标志将该段标识为 SYN 报文段？

5. gaia.cs.umass.edu 发送给客户端计算机的 SYNACK 报文段序号是什么？SYNACK 段中的确认字段的值是多少？gaia.cs.umass.edu 如何确定这个值？在该 TCP 段中，是什么标志将该段标识为 SYNACK 报文段？

6. 包含 HTTP POST 命令的 TCP 段序号是什么？请注意，为了找到 POST 命令，需要在 Wireshark 窗口底部的数据包内容字段中进行查找，以便找到 DATA 字段中包含“POST”的 TCP 报文段。

7. 考虑包含 HTTP POST 的 TCP 段，这里我们将其看作 TCP 连接中的第一个 TCP 段。TCP 连接中前六个报文段（包括那个包含 HTTP POST 的报文段）的序号是什么？每个报文段发送的时刻是多少？每个报文段在什么时刻收到了 ACK？对比每个 TCP 段的发送时刻和收到确认的时刻，这六个 TCP 段的 RTT 值分别是多少？收到 ACK 之后，每个 TCP 段的 `EstimatedRTT` 值是多少（参见教材第 3.5.3 节，p239）？假设开始时 `EstimatedRTT` 值等于第一个 TCP 段的 RTT 测量值，其后所有 TCP 段的 `EstimatedRTT` 值使用第 239 页的 `EstimatedRTT` 方程来计算。

   注意 Wireshark 有一个很不错的功能，允许为本机发送的每个 TCP 段绘制出 RTT。在已捕获数据包中选择一个从客户端发送到 gaia.cs.umass.edu 服务器的 TCP 段，然后选择*统计 → TCP 流图形 → 往返时间*。

8. 前六个 TCP 段的长度是多少？[1]

9. .在整个跟踪过程中，接收端给出的可用缓冲区的最小值是多少？接收端缓冲区空间不足是否会阻止发送端继续发送数据？

10. 跟踪文件中是否有重传的 TCP 段？为了回答这个问题，你需要检查跟踪文件的哪一个部分？

11. 在 ACK 中接收端通常会确认多少数据？你可以找出接收端利用 ACK 来确认已接收的其他 TCP 段的情形吗（即**累计确认**，参见教材 247 页的表 3.2）？

12. TCP 连接的吞吐量（每单位时间内传输的字节数）是多少？说明一下你的计算方法。

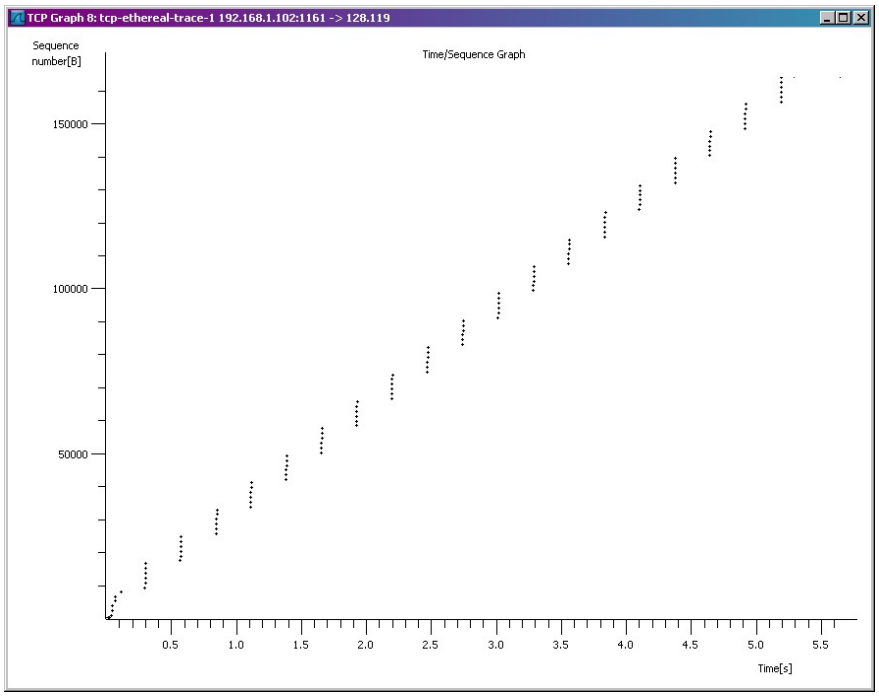
---

[1] 我们提供的 tcp-ethereal-trace-1 跟踪文件中，TCP 段长度都少于 1460 字节。这是因为我们完成本实验所使用的计算机有一个以太网卡，它将最大 IP 数据包的长度限制为 1500 字节（40 字节的 TCP + IP 包头和 1460 字节的 TCP 有效载荷）。此 1500 字节值是以太网允许的标准最大长度。如果您的跟踪文件显示 TCP 段长度大于 1500 字节，且您的计算机正在使用以太网连接，则 Wireshark 将报告错误的 TCP 段长度；它也可能只显示一个大的 TCP 段而不是多个较小的段。 您的计算机确实可能会发送多个较小的段，如其接收到的 ACK 所示。这种 TCP 段长度的不一致性是以太网驱动程序和 Wireshark 软件之间相互作用的结果。因此建议您使用我们提供的跟踪文件来完成本实验。

四、 TCP 拥塞控制

现在来研究每单位时间从客户端发送到服务器的数据量。我们将使用 Wireshark 的 TCP 图形实用程序——时间序列图（Stevens）来绘制出数据，而不是直接使用 Wireshark 窗口中的原始数据来进行计算。

在 Wireshark 的“listing of captured-packets”窗口中选择一个 TCP 段，然后选择*统计→ TCP 流图形→ 时间序列图（Stevens）*。你应该看到一个与下图相似的图形，该图是根据 http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip 中的 tcp-ethereal-trace-1 跟踪文件中的捕获数据包创建的（参见前面的脚注）：



这里，每个点表示一个发送的 TCP 段，横轴为发送的时间，纵轴为 TCP 段的序号（按每字节的顺序编号）。请注意，彼此堆叠的一组点表示发送方连续发送的一系列数据包。

13. 请使用 http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip 中的跟踪文件 tcp-ethereal-trace-1，回答以下关于 TCP 报文段的问题。使用时间序列图（*Stevens*）绘图工具来查看从客户端发送到 gaia.cs.umass.edu 服务器的 TCP 段序号与发送时刻，你能找出 **TCP 慢开始（slowstart）阶段的起始时刻**和**拥塞避免阶段开始的时刻**吗？该测量数据所显现的 TCP 行为与教材中讨论的理想化 TCP 行为之间存在着差异吗？如果存在，请尝试解释一下。
14. 使用你自己的跟踪文件或抓包结果，重新回答一下问题 13。