## 5.5 流跟踪

Wireshark 分析功能中令人满意的一点就是它能够将来自不同包的数据重组成统一易读的格式，一般称作 packet transcript。流跟踪功能可以把从客户端发往服务端的数据都排好序使其变得更易查看，这样你就不需要从一堆小块数据里一个包一个包地跟踪了。

现有 4 种类型的流可以被跟踪。

**TCP**流：重组使用 TCP 协议的数据，比如 HTTP 和 FTP。

**UDP**流：重组使用 UDP 协议的数据，比如 DNS。

**SSL**流：重组加密的协议，比如 HTTPS。你必须提供密钥来解密流量。

**HTTP**流：从 HTTP 协议中重组和解压数据。当使用 TCP 流跟踪但又没有完全解码出 HTTP 数据时，这个功能就派上用场了。

我们以一个简单的 HTTP 交互举例来说，打开 http_google.pcapng，并在文件中单击任意一个 TCP 或者 HTTP 数据包，右键单击这个文件并选择 Follow TCP Stream。这时 TCP 流就会在一个单独的窗口中显示出来（见图 5-14）。

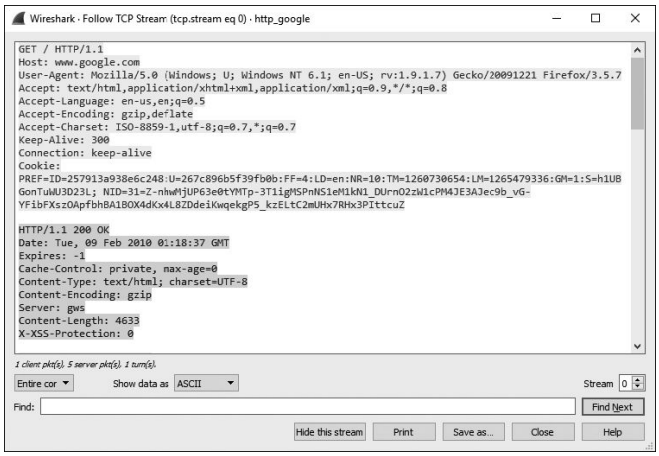


图 5-14　跟踪 TCP 流窗口将通信内容以更简单可读的方式进行了重新组织

我们注意到这个窗口中的文字以两个颜色显示，其中红色用来标明从源地址前往目标地址的流量，而蓝色用来区分出相反方向，也就是从目标地址到源地址的流量。这里的颜色标记以哪方先开始通信为准，在我们的例子中，客户端最先建立了到服务器的连接，所以显示为红色。

在这个 TCP 流中，你可以清晰地看到这两台主机之间进行的绝大多数通信。在这些通信开始的时候，最初是对 Web 根目录的 GET 请求，然后是来自服务器的一个用 HTTP/1.1 200 OK 表示请求成功的响应。当客户端请求另一个文件并由服务器给予响应的时候，这个简单模式就会重复出现。你可以看到一个用户正在浏览 Google 首页，但你不需要遍历每个数据包，就可以轻松地滚动文本，事实上你和这个用户看到的别无二致，只不过是以更深入的形式去看。

在这个窗口中除了能够看到这些原始数据，你还可以在文本间进行搜索，将其保存成一个文件，打印出来，也可以用 ASCII 码、EBCDIC、十六进制或者 C 数组的格式去看。这些选项都可以在跟踪 TCP 流窗口的下面找到。

跟踪 TCP 流在你和一些协议打交道的时候，绝对是一个好方法。