|  |  |
| --- | --- |
| 标题 | 实践作业2：HTTP协议分析与编程 |
| 发布时间 | 2016年10月26日 |
| 截止时间 | 2016年12月4日 |
| 评分方式: | 打分制:100.0分 |
| 作业内容 | 分析HTTP协议，编写一个简单的Web服务器。  1）利用Wireshark捕获数据包，分析一个浏览器与Web服务器的交互过程。  2）编写一个简单的Web服务器。  a）实现的服务器应能与标准的浏览器（如IE浏览器）进行简单的交互；  b）记录浏览器与服务的交互过程；  c）利用HTML语音编写网页（可以不包括图像、声音等多媒体文件），浏览器可通过编写的Web服务器正常访问该网页；  d）支持多用户并发访问；  e）可以采用C/C++或Java语音进行实现。  3）选做：扩展编写的简单Web服务器，使浏览器能够浏览Web上存储的图像。  4）提交报告，对以上内容进行总结和分析（包括程序的结构和说明）。 |

**信息安全 1410658 杨旭东**

**一、实验目的及任务**

1、熟悉并掌握WireShark的基本操作，了解网络协议实体间的交互以及报文交换。

2、通过对WireShark抓包实例进行分析，进一步加深对HTTP网络协议的理解。

3、编写一个简单的Web服务器，实现作业（2）（3）项所要求的功能。

4、培养理论联系实践的科学研究精神。

**二、实验环境**

1、系统环境：Windows 10 家庭中文版 1607

2、浏览器：Chrome

3、WireShark：Version 2.2.1

4、IDE：Visual Studio 2015 Update 3

5、编程语言：C++

**三、实验原理**

1、利用Wireshark软件抓取本地PC的数据包，观察其主要使用的网络协议。

2、根据所获数据包的内容分析相关协议，从而加深对HTTP网络协议的理解。

3、利用Winsock实现本地Web服务器，并实现与浏览器的交互。

**四、实验步骤**

**1、实验前的准备工作**

（1）关闭所有可能会请求网络的文件以及程序。

（2）打开浏览器，删除所有的Cookie、临时文件、表单数据以及浏览记录等等。

（3）单击开始，打开运行，输入'cmd'，输入命令'ipconfig/flushdns'，回车，即可看到成功刷新DNS缓存的显示。

**2、WireShark的使用**

（1）启动WireShark。

（2）启动PC上的Chrome浏览器。

（3）开始分组捕获：选择“抓包”下拉菜单中的“抓包参数选择”命令，在“WireShark:抓包选项”窗口中可以设置分组捕获的选项。

（4）在这次实验中，使用窗口中显示的默认值。选择“抓包”下拉菜单中的“网络接口”命令，显示计算机中所安装的网络接口（即网卡）。我们需要选择电脑真实的网卡，点击后显示本机的IP地址。

（5）随后，点击“开始”则进行分组捕获，所有由选定网卡发送和接收的分组都将被捕获。

（6）待捕获一段时间，关闭浏览器，选择主窗口中有的“stop”按钮，可以停止分组的捕获。

（7）选择“文件”下拉菜单中的“另存为”，保存到我的文档中。

**3、编写Web服务器**

（1）建立.cpp文件，引用头文件winsock2.h，利用SOCKET创建套接字，bind绑定端口，listen监听请求，accept接受请求，接受请求后用thread实现多线程，实现多用户并发访问，出现一个请求就创建一个线程返回请求。

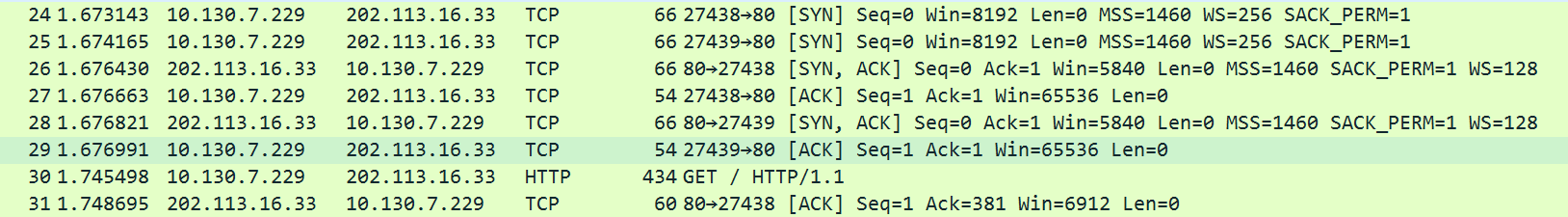
（2）编写一个简单html网页，包含文本图片等元素，放在Web服务器文件夹下。

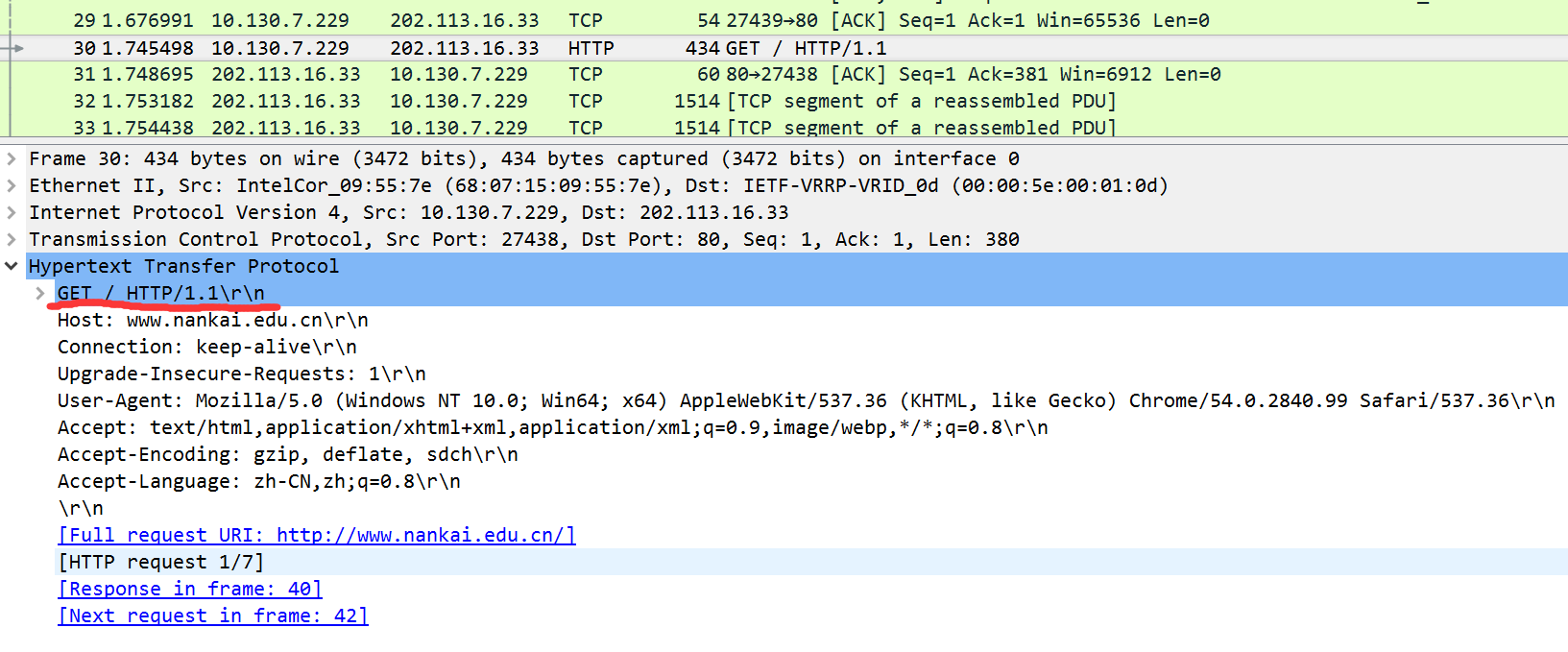
（3）运行Web服务器，开始监听客户请求。

（4）客户在Chrome中输入<http://127.0.0.1:81/index.html>获取网页及各种文本图片等资源，之后显示。

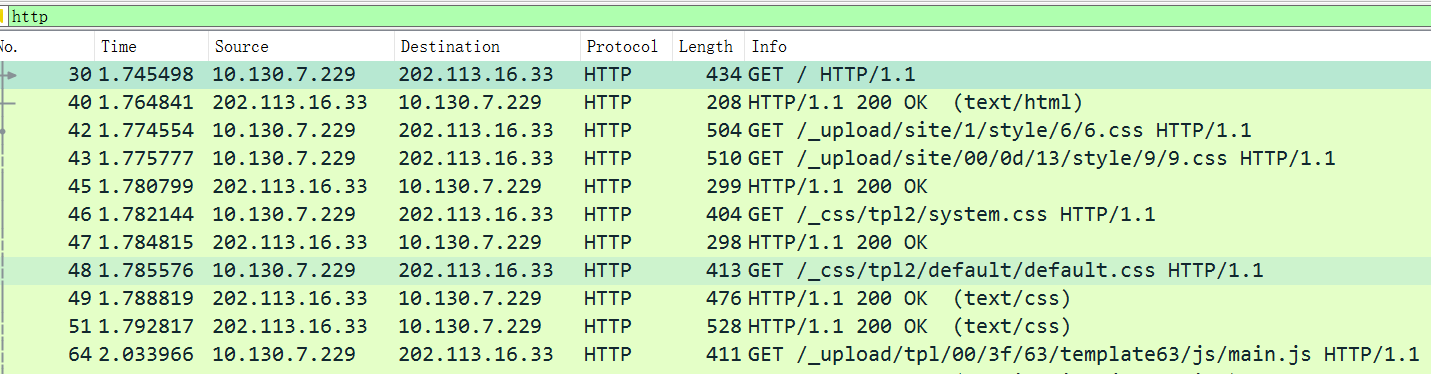
**五、实验结果展示与分析**

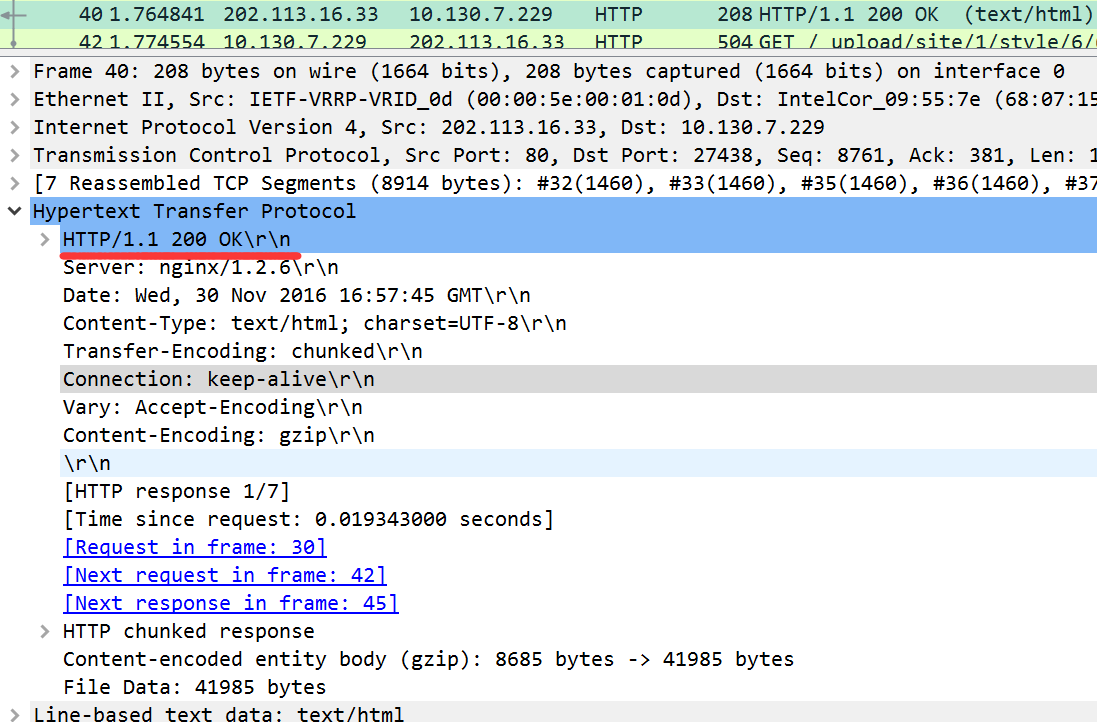
**1、HTTP协议分析**

在获得了要请求网页所在主机的IP后，本地PC将作为客户机向服务器发起TCP连接请求，连接请求建立成功后，客户机即可向服务器返回确认，这就是俗称的“三次握手”过程。

该图片上标号为24,26,27和25,28,29分别为一个三次握手过程，源端口号分别为27438和27439，目的端口均为80。以第二个连接为例，第25个为客户机向服务器发起请求，第28帧为服务器作出确认和响应，第29个为客户机向服务器服务器返回确认。第30个则为一个HTTP请求，其请求报文如下图所示：

划红线标记的那一行为请求行，是后继的行为首部行。HTTP使用传输层协议是TCP，方法字段大多数为GET（如下图），还显示了接受的文件类型text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,\*/\*;q=0.8，接受语言为中文，浏览器类型User-Agent以及使用持久连接（Connection: keep-alive\r\n）。

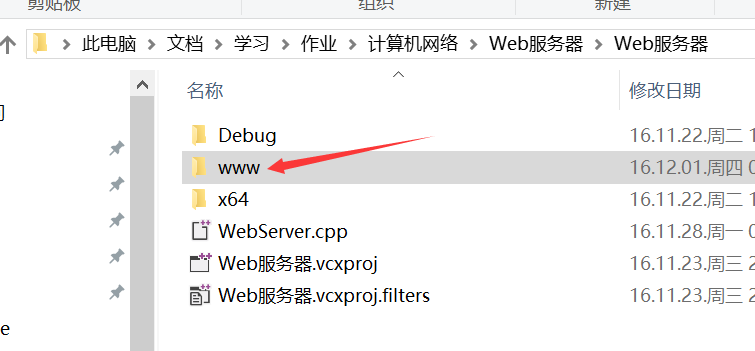
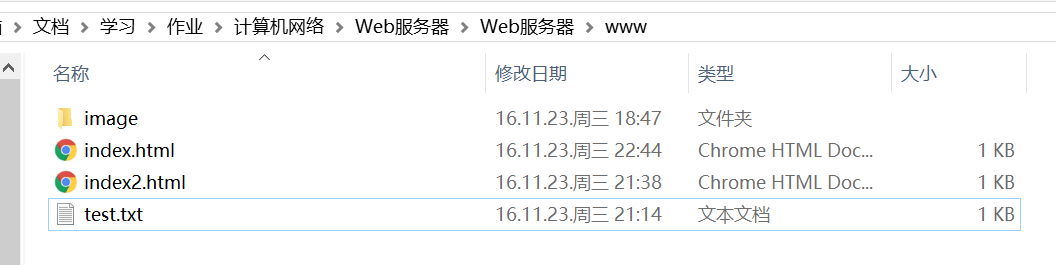
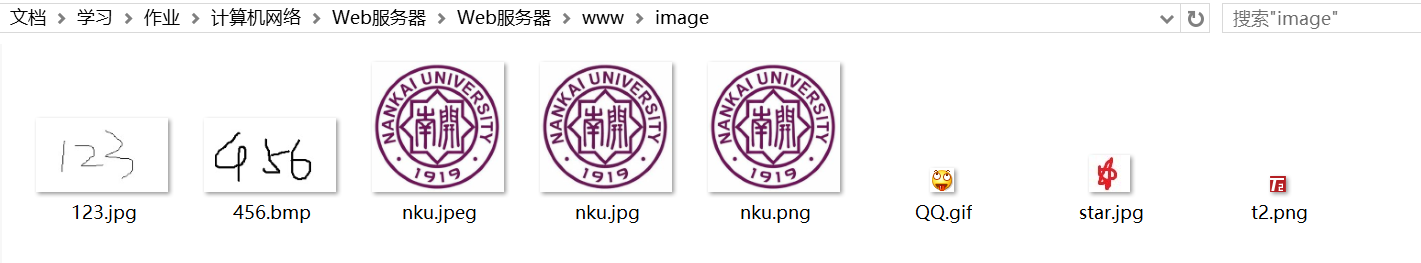
服务器的响应报文如下图所示：划红线标记上的那一行为初始状态行，接下来的六行为首部行。

客户机通过分析html文件，继续向服务器发起连接，并且可能携带CookieID信息。

**2、Web服务器实现**

可以通过浏览器访问本地服务器上的网页。

网页文件及其中的文本图片等资源放在Web服务器目录下，名为www。

网页中已包含.jpg\.bmp\.png\.gif等图片格式，网页样式如图。



打开Web服务器.exe，开始监听，在浏览器输入<http://127.0.0.1:81/index.html>，可以获得响应。

服务器（为win32控制台程序）记录响应请求如下（请求较多，节选）：

