计算机网络lab2实验报告

姓名：罗昕珂

学号：2013622

一、IIS服务器的搭建

1. 启用功能：

我们以Windows 桌面版系统为例，进入Windows的“控制面板”

进入“程序”，然后点“启用或关闭Windows功能”,找到Internet Information Services,在Internet Information Services之前的选项框上打勾。

2. 设置网站路径和端口

将网站挂在8080端口上，我们需要在“网站”中新建一个网站。

大致操作为：右键“网站”->“添加网站”->输入信息->点击“确定”即可，其中，“网站名称”任意填写，“物理路径”选择上面放了网站相关文件的目录路径，剩下端口改为8080，点击确定，访问http://localhost:8080/或者IP地址即可看到网站

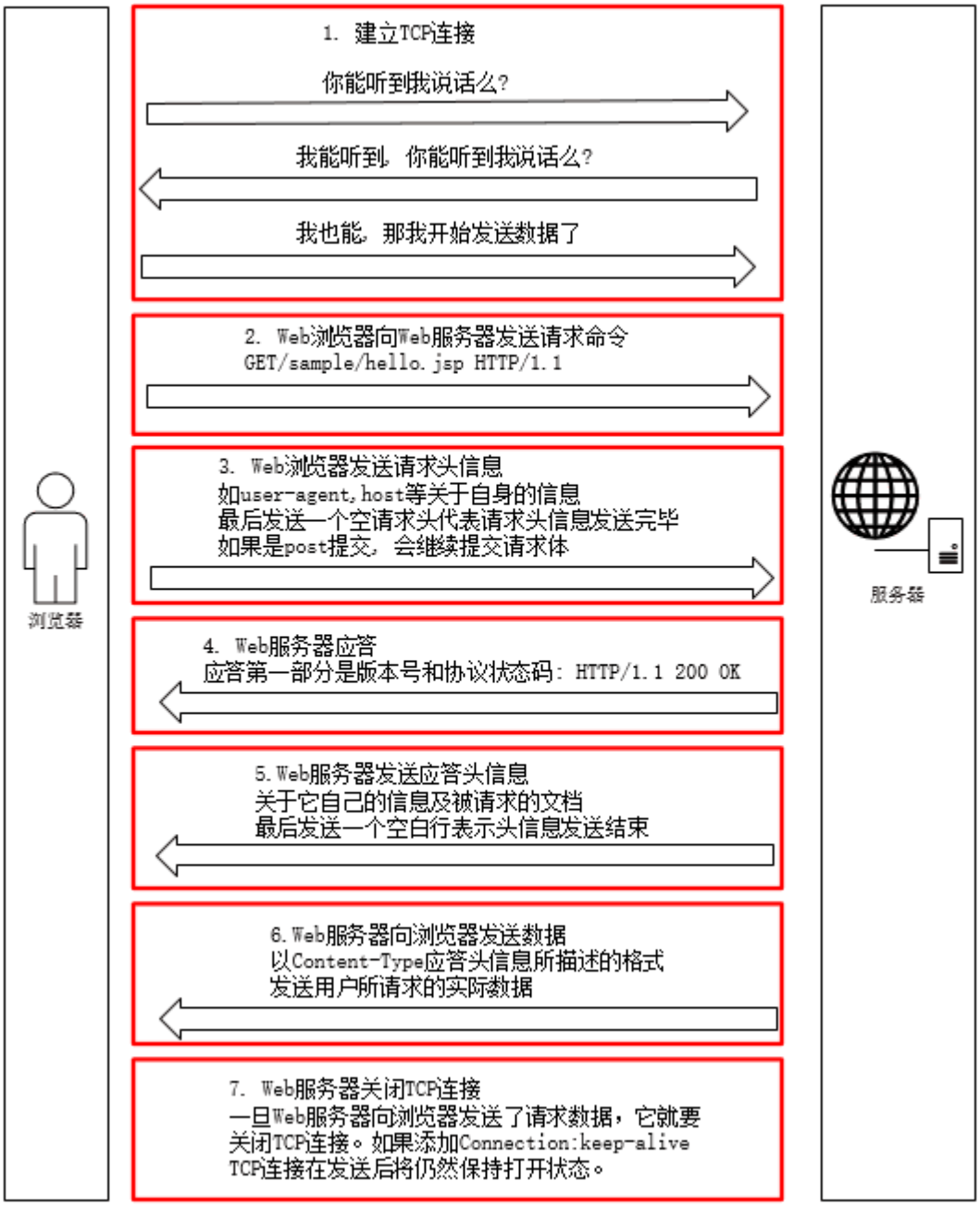
二、制作简单的Web页面

test.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Title</title>
</head>
<body>
    <div style="text-align:center;">
      <p>
        姓名:罗昕珂<br><br>
        学号:2013622<br><br>
        专业:计算机科学与技术专业<br>
      </p>
        <img src="1.png",alt="1.png"/>
    </div>
</body>
</html>
```

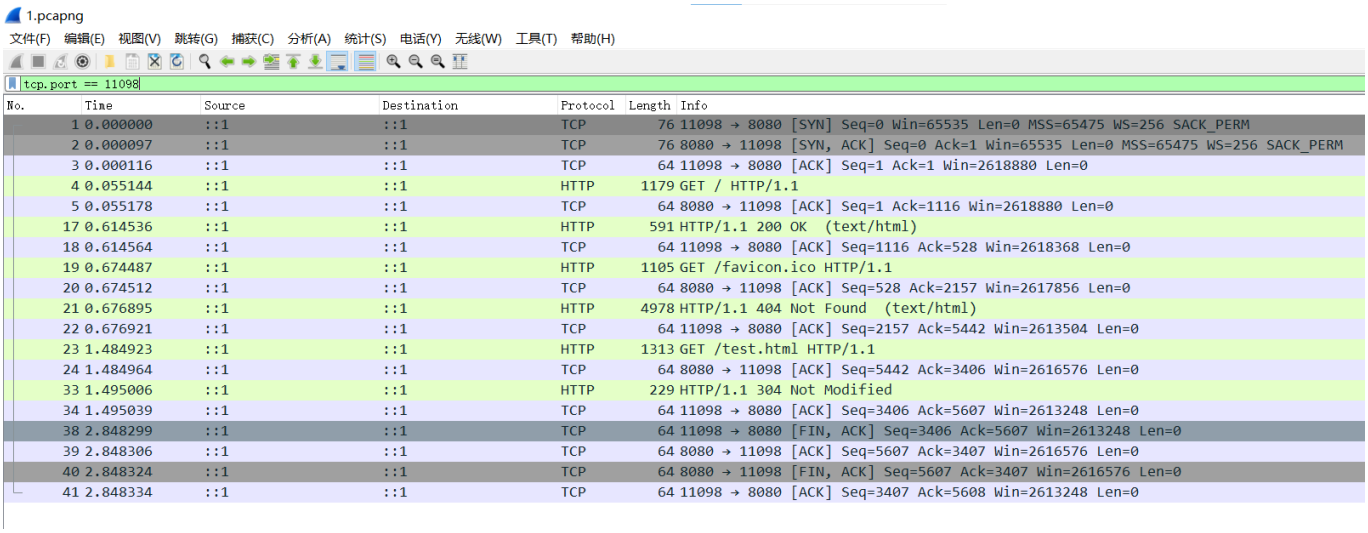
三、使用Wireshark捕获浏览器与Web服务器的交互过程

一次完整的HTTP请求过程：



报文分析

过滤器：tcp.port == 11098



1、通信建立成功报文（三次握手）

127.0.0.1的端口 11098为客户端 127.0.0.1的端口8080为服务端

第一次握手：

客户端主动打开。发送SYN=1，协商window size 、TCP MSS（最大报文长度） seq=0 len=0 MSS=65475 win=65535最大窗口大小。

> TCP(Transmission Control Protocol)传输控制协议的几个状态（SYN，FIN，ACK，PSH，RST，URG)
> SYN表示建立连接，FIN表示关闭连接，ACK表示响应，PSH表示有 DATA数据传输，RST表示连接重置。

客户端发送位码为syn = 1，随机产生seq number的数据包到服务器，服务端由SYN=1知道，客户端要求建立联机；

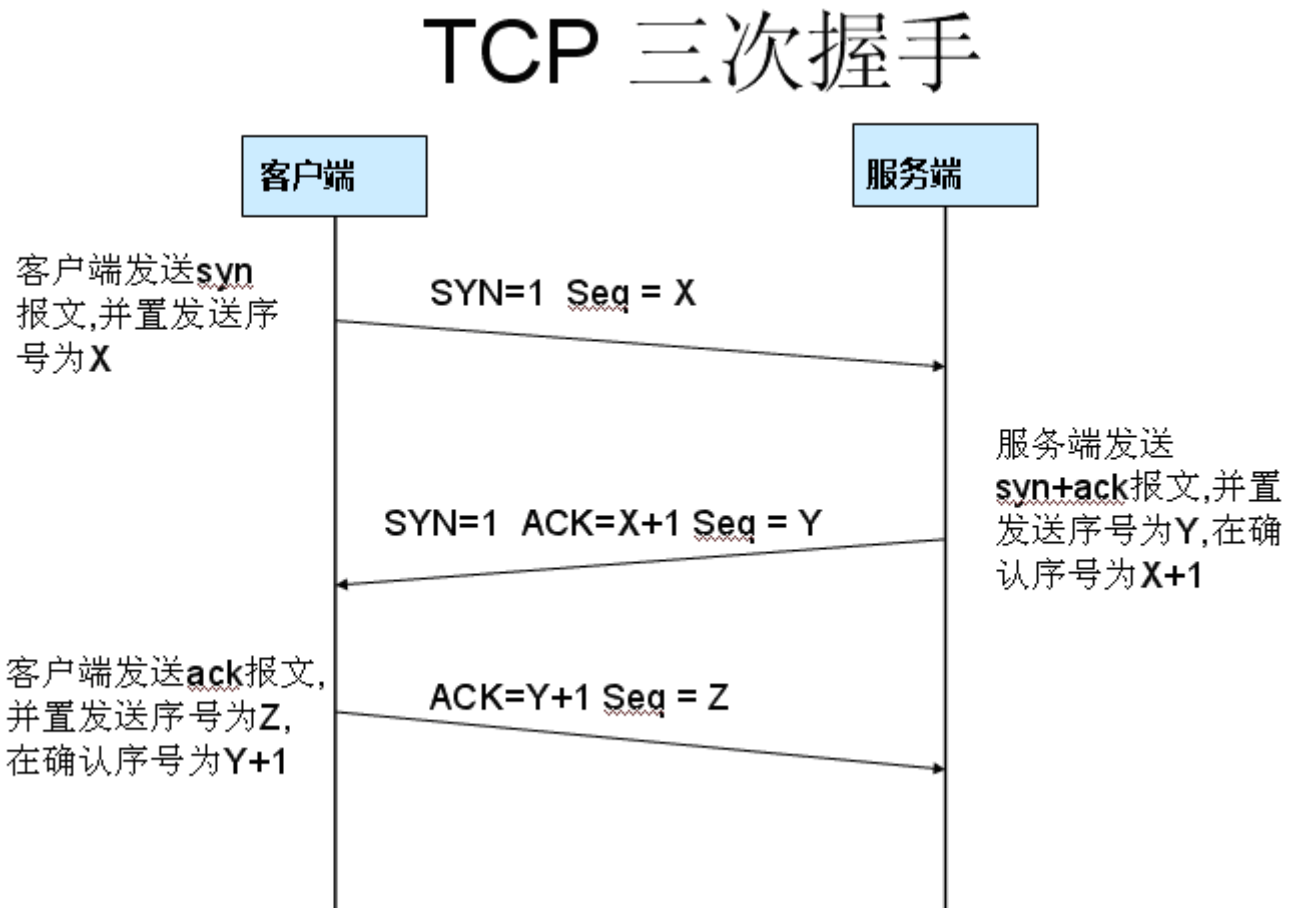
第二次握手：

服务端接收到syn。回复syn ack=0+1 (ack number=客户端的seq+1)并发送自身seq=0 确认自己的最大win=65535 MSS=65475

第三次握手：

客户端接收到服务端发送来的ack，检查ack number是否正确，和服务端自身的seq，客户端要发送ack=0+1（ack number=服务端的seq+1)，给服务端发送确认报文。

服务端接收后，通信建立成功



为什么要三次握手

为了防止已失效的连接请求报文段突然又传送到了服务端，因而产生错误。

例如：

client发出的第一个连接请求报文段没有丢失，但在某个网络结点长时间的滞留了，导致连接释放以后才到达server。

server收到这个失效的连接请求报文段后，就误认为是client再次发出的一个新的连接请求。于是就向client发出确认报文段，同意建立连接。

假设不采用"三次握手"，那么只要server发出确认，新的连接就建立了，就产生了错误。

2、数据收发报文（HTTP）

- server→client（PSH、ACK）：服务器推送数据最后一个分段给客户端
- client→server（ACK）：客户端对第1个报文进行接收确认

HTTP协议

请求与响应：客户端发送请求，服务器端响应数据

无状态的：协议对于事务处理没有记忆能力，客户端第一次与服务器建立连接发送请求时需要进行一系列的安全认证匹配等，因此增加页面等待时间，当客户端向服务器端发送请求，服务器端响应完毕后，两者断开连

接，也不保存连接状态，一刀两断！恩断义绝！从此路人！下一次客户端向同样的服务器发送请求时，由于他们之前已经遗忘了彼此，所以需要重新建立连接。

应用层：Http是属于应用层的协议，配合TCP/IP使用。

TCP/IP：Http使用TCP作为它的支撑运输协议。HTTP客户机发起一个与服务器的TCP连接，一旦连接建立，浏览器（客户机）和服务器进程就可以通过套接字接口访问TCP。

3、通信断开报文(TCP四次挥手)

第一次挥手：

TCP客户端发送一个FIN=1 +ACK=Z +SEQ=X（发送序号为x），用来传输关闭客户端到服务端的数据。进入FIN_WAIT1状态。

第二次挥手：

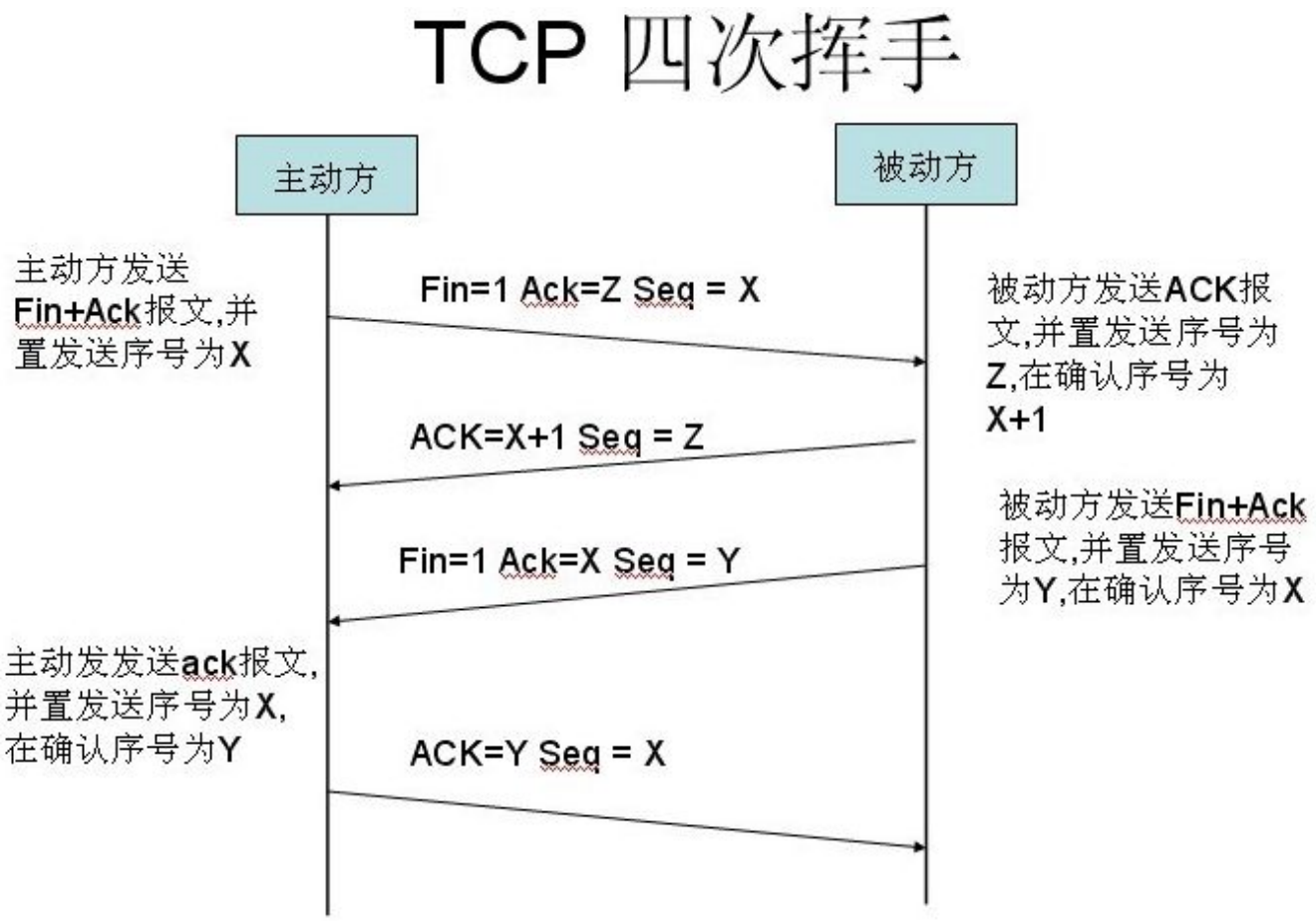
服务端收到FIN，被动发送一个ACK（SEQ+1），进入CLOSE_WAIT状态，客户端收到服务端发送的ACK，进入FIN_WAIT2状态。

第三次挥手：

服务器关闭客户端连接，发送一个 FIN+ACK+SEQ 给客户端。进入 LAST_ACK 状态。

第四次挥手：

客户端发送 ACK（ACK=SQE序号+1）报文确认，客户端进入 TIME_WAIT 状态，服务端收到 ACK 进入 CLOSE 状态。



由于TCP连接是双向的，因此每个方向都需要单独进行关闭。原则是当一方完成它的数据发送任务后就能发送一个FIN来终止这个方向的连接。

若服务端主动关闭同理：

- 服务端主动关闭，发送FIN。Seq=328
- 服务端状态为FIN_wait1 处于半关闭状态
- 客户端状态为closed_wait 处于半关闭状态
- 客户端发送确认ack ack=328+1
- 服务端状态为FIN_wait2
- 客户端发送FIN seq=133
- 客户端状态为LAST_ack
- 服务端状态为time_wait
- 服务端发送ack ack=133+1
- 客户端状态closed
- 服务端等待2ms后状态closed，至此本次通信结束

为什么要四次挥手

TCP协议是一种面向连接的、可靠的、基于字节流的运输层通信协议。

1. TCP是全双工模式，当主机1发出FIN报文段时，只是表示主机1已经没有数据要发送了，但是，这个时候主机1还是可以接受来自主机2的数据；

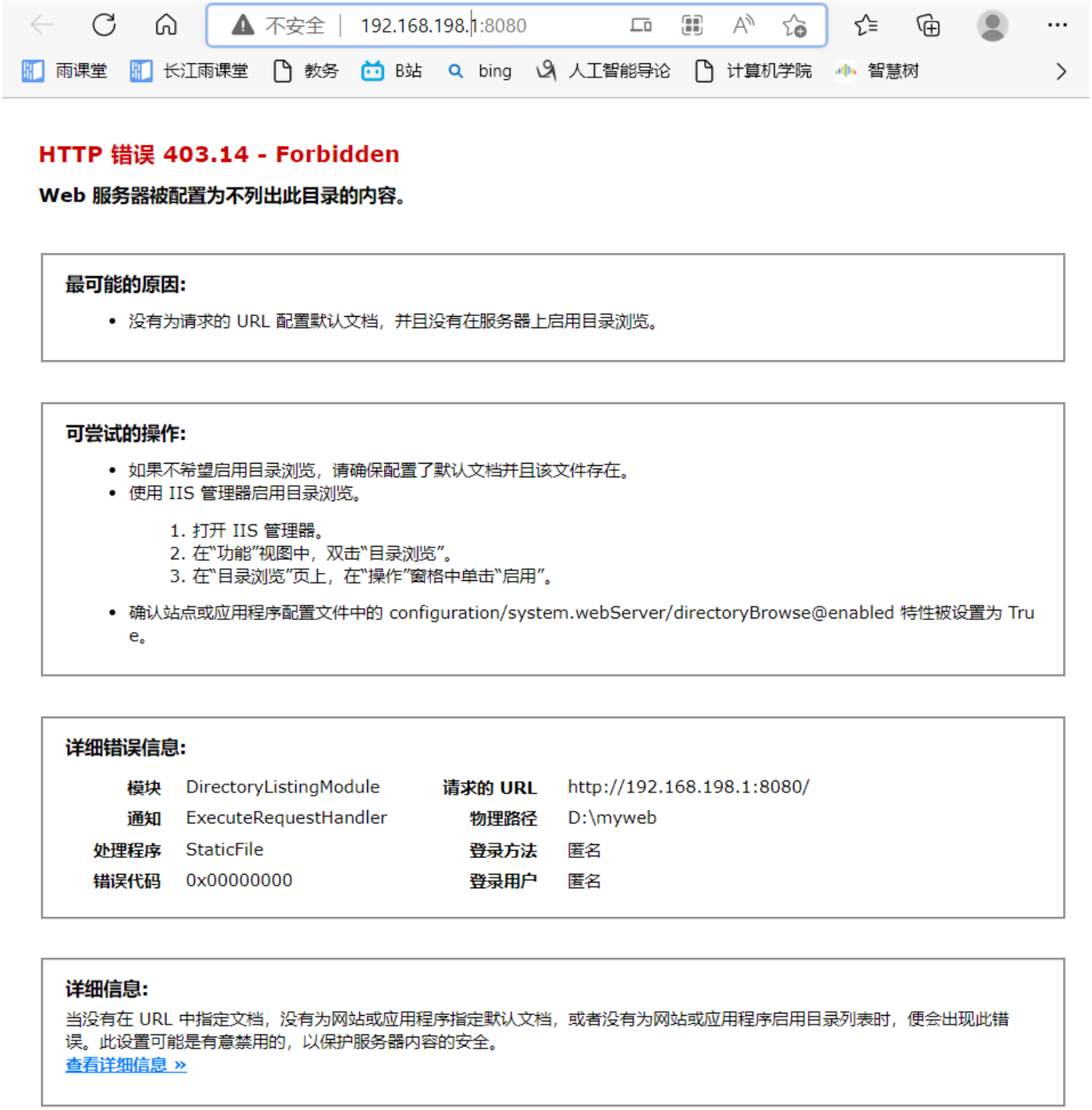
2. 当主机2返回ACK报文段时，表示它已经知道主机1没有数据发送了，但是主机2还是可以发送数据到主机1的；

3. 当主机2也发送了FIN报文段时，这个时候就表示主机2也没有数据要发送了，就会告诉主机1，我也没有数据要发送了，

4. 之后主机1也返回ACK报文段，表示它已经知道主机1没有数据发送了，彼此就会愉快的中断这次TCP连接。

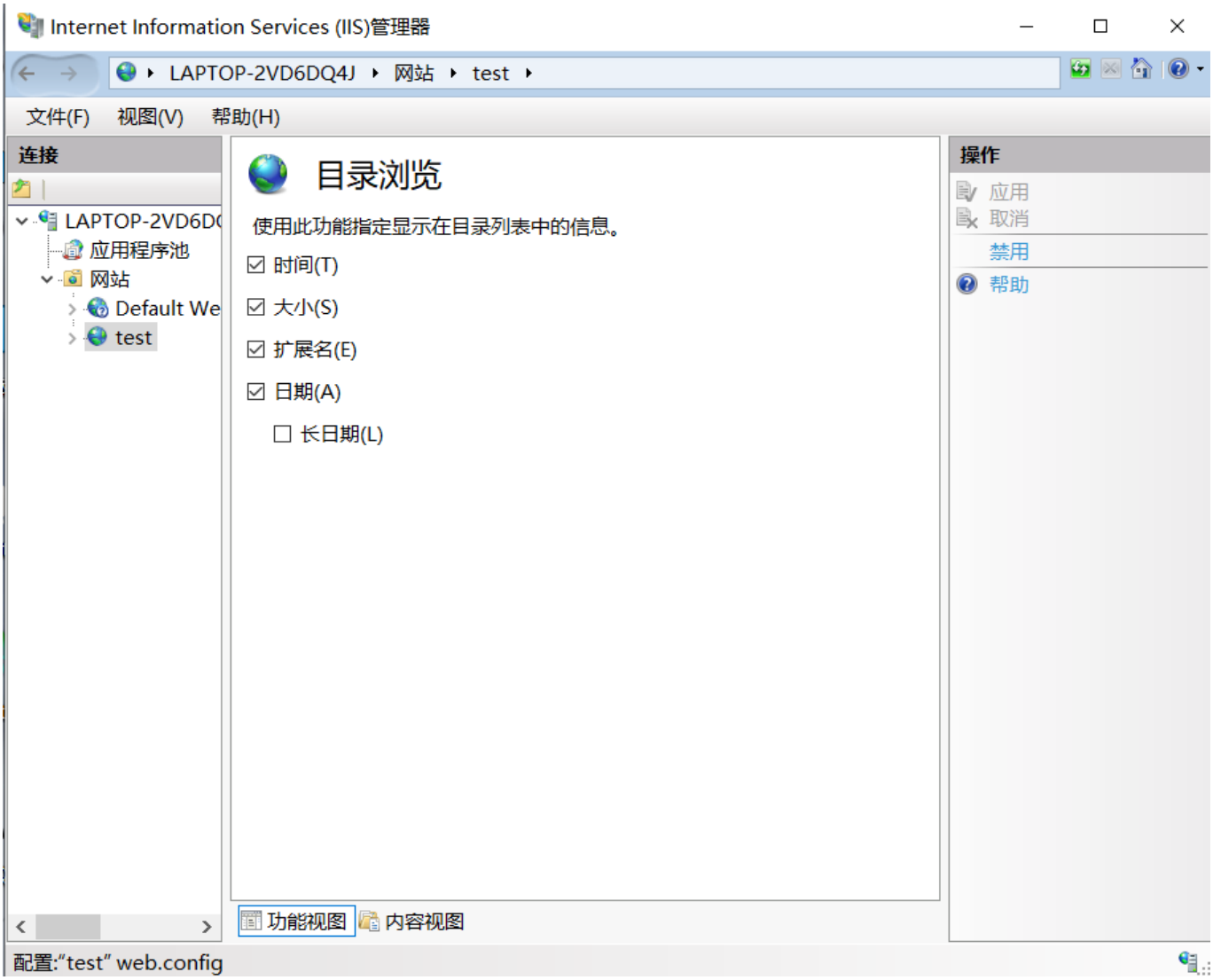
遇到的问题：

1、出现如图所示的页面，无法访问目录



解决方法：

默认新建网站都禁用了目录浏览，需要在iis中启动



2、本地连接中WireShark无法抓包

由于WireShark只能抓取经过电脑网卡的包，使用localhost或者127.0.0.1进行测试的，流量是不经过电脑网卡的，所以WireShark要使用 Adapter for loopback traffic capture:： 迂回路线，就是本机自己的网络，抓的是127.0.0.1 的包

3、抓包中出现rst报文

client→server（RST）：客户端向服务器发送一个RST报文，其中seq为server挥手ack包的ack值。

这是因为对服务器侧而言，对应的四元组仍然处于TIME_WAIT状态，而客户端侧并不存在这个四元组的socket信息。

TCP接收到一个数据段，但是这个数据段所标识的连接不存在。

于是客户端使用ACK报文中的ack值作为seq，发送RST报文给服务器，

当服务器收到RST报文后，无论处在TCP的哪个状态，都会立即进入close状态，进而服务器侧对应被TIME_WAIT状态冻结的四元组得以被释放，客户端侧的复用就成功了。

4、客户端向服务端发送http请求时，返回状态码为304 Not Modified

在客户端向服务端发送http请求时，若返回状态码为304 Not Modified 则表明此次请求为条件请求。

服务器判断出客户端缓存的资源是否是最新的,如果是的话,服务器就会返回HTTP/304 Not Modified响应头, 但没有响应体.客户端收到304响应后,就会从本地缓存中读取对应的资源。

127.0.0.1	TCP	44	11104 → 11105 [FIN, ACK] Seq=2 Ack=1 Win=2619648
127.0.0.1	TCP	44	11105 → 11104 [ACK] Seq=1 Ack=3 Win=2619648 Len=
127.0.0.1	TCP	44	11105 → 11104 [RST, ACK] Seq=1 Ack=3 Win=0 Len=0
::1	HTTP	229	HTTP/1.1 304 Not Modified
::1	TCP	64	11098 → 8080 [ACK] Seq=3406 Ack=5607 Win=2613248
::1	TCP	64	11099 → 8080 [FIN, ACK] Seq=1 Ack=1 Win=2618880

所以：当访问资源出现304访问的情况下其实就是先在本地缓存了访问的资源。

5、只出现了两次挥手的情况

发现服务端没有发送fin报文给客户端，后来发现是要将浏览器全部关掉，才能让服务端发送断连报文。