**暨南大学本科实验报告专用纸**

课程名称 计算机网络实验 成绩评定

实验项目名称 Internet应用与应用层协议分析 指导教师 潘冰

实验项目编号 实验项目类型 实验地点

学生姓名 陈星池 学号 2019051102

学院 智能科学与工程学院 系 专业 信息安全

实验时间 年 月 日 午～ 月 日 午 温度 ℃湿度

1. **实验目的**

1.理解WWW 、 DNS服务、FTP服务、SMTP的作用和原理；

2.学会使用wireshark分析HTTP、FTP、SMTP和DNS协议的工作过程，加深对协议格式和工作原理的理解。

1. **实验内容**

1.通过域名访问WWW、FTP服务器，分析DNS、WWW、FTP工作过程，并使用WireShark分析相关协议格式；

2.在客户端访问SMTP服务器，使用wireshark分析SMTP、POP3协议的工作过程。（可以在客户端安装outlook或使用QQ邮件服务器或自己编程）

1. **实验环境**

笔记本电脑连接校园网JNU或JNU-Secure，使用wireshark捕获相关数据包

HTTP协议分析实验访问网站<http://www.jiaozuoye.com/joj/classRoom/experiments/84>

DNS协议分析实验访问news.jnu.edu.cn

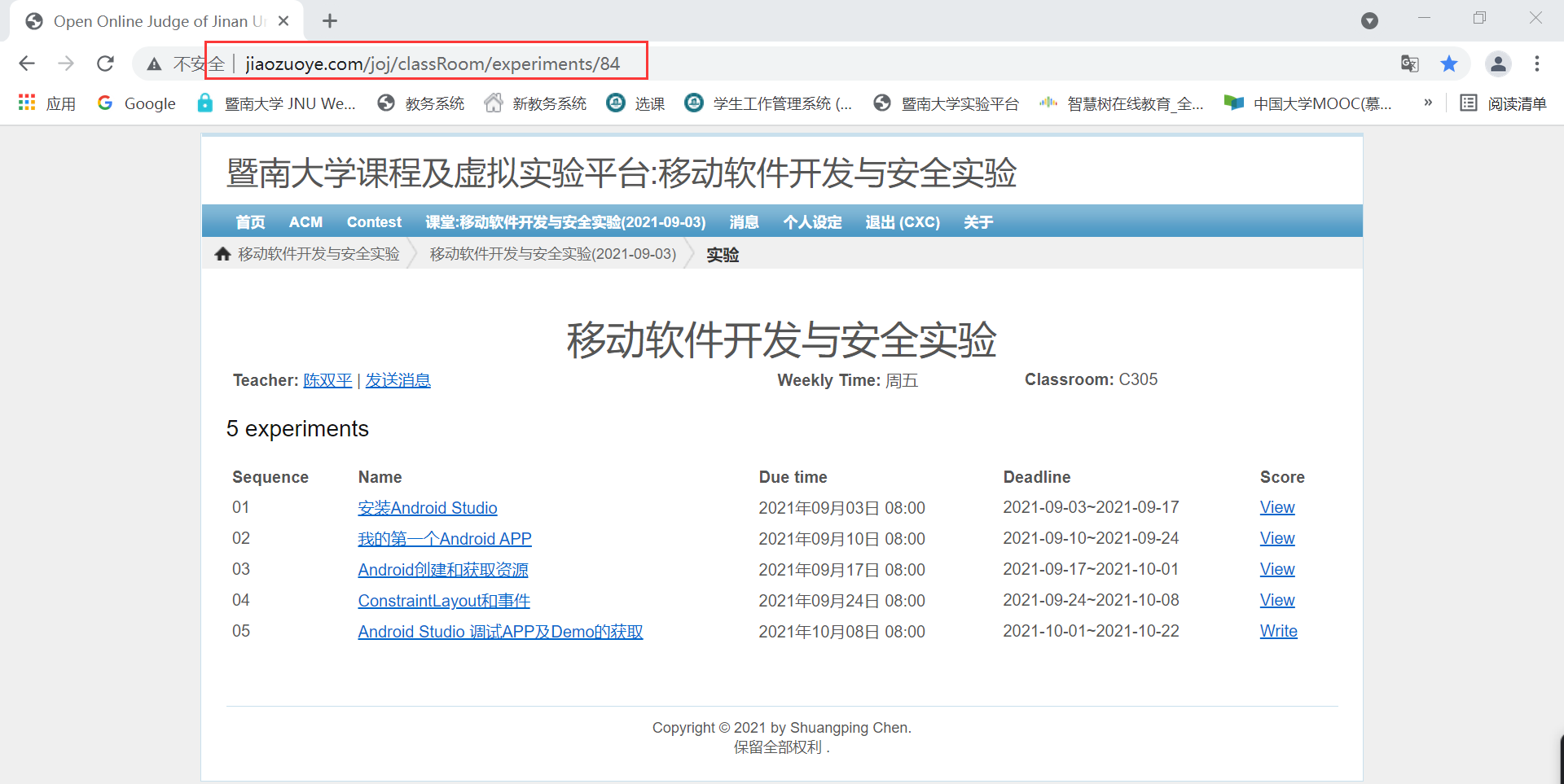
FTP协议分析实验访问FTP服务器ftp://ftp.jnu.edu.cn

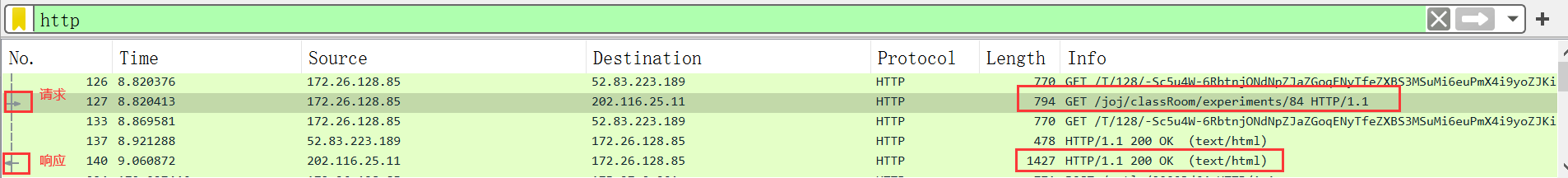
SMTP和POP协议分析实验下载客户端的邮件软件Foxmail，并对服务器进行配置

1. **实验步骤**

**一、HTTP协议分析**

访问网站<http://www.jiaozuoye.com/joj/classRoom/experiments/84>

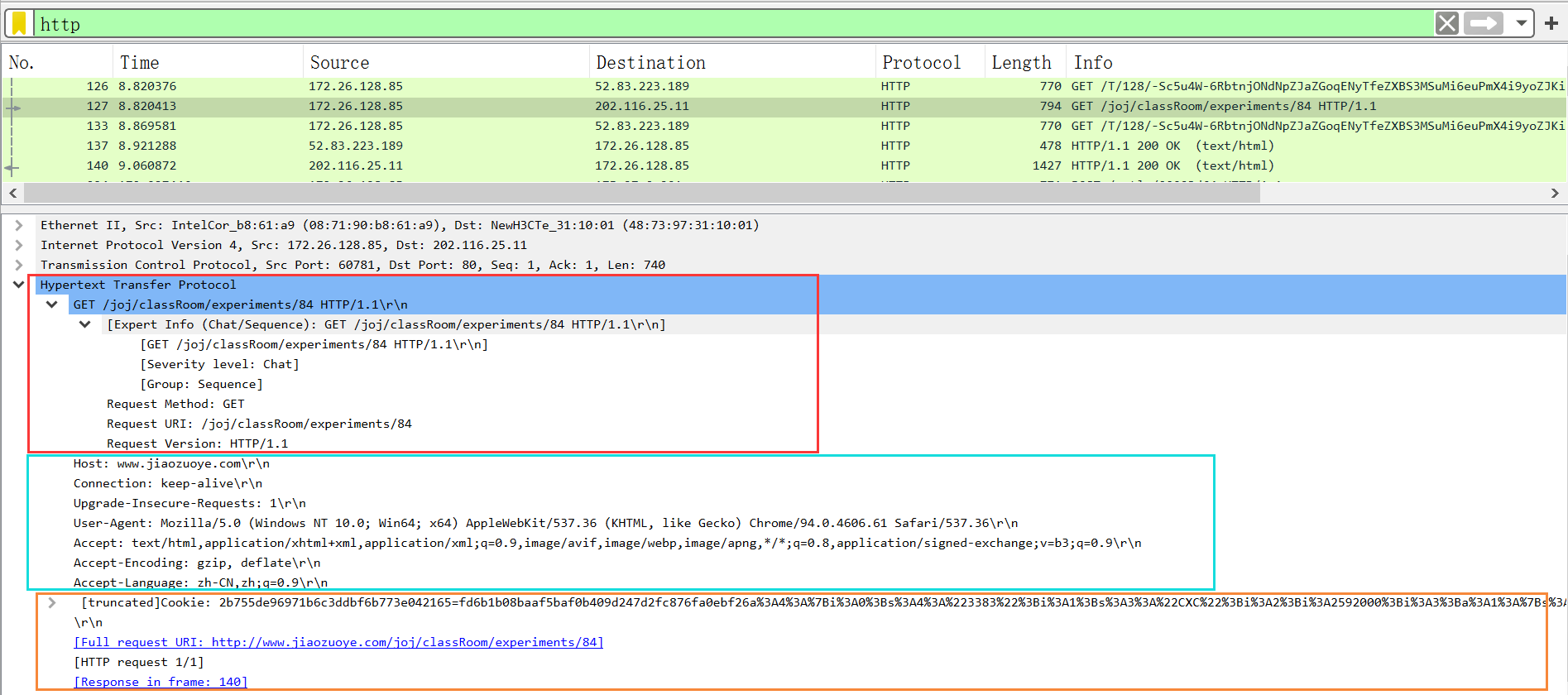


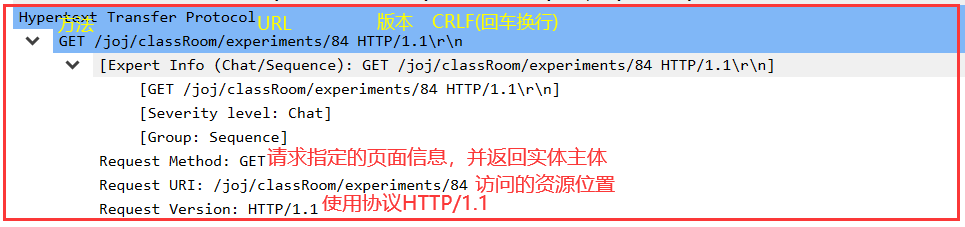


由截图可以看到HTTP工作模式为：客户端发起请求，服务器回送响应。

根据Info初步分析，请求的Info包括方法、URL、版本，响应的Info包括版本、状态码、短语、响应的内容类型。

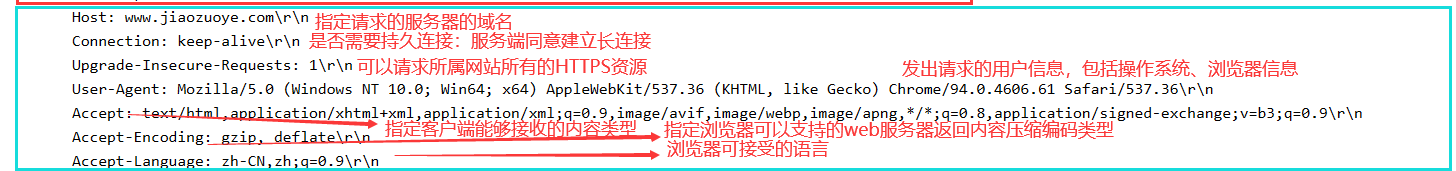
**具体分析请求报文：**





HTTP请求方法表

|  |  |
| --- | --- |
| 方法 | 描述 |
| GET | 请求指定的页面信息，并返回实体主体。 |
| HEAD | 类似于 GET 请求，只不过返回的响应中没有具体的内容，用于获取报头 |
| POST | 向指定资源提交数据进行处理请求（例如提交表单或者上传文件）。数据被包含在请求体中。POST 请求可能会导致新的资源的建立和/或已有资源的修改。 |
| PUT | 从客户端向服务器传送的数据取代指定的文档的内容。 |
| DELETE | 请求服务器删除指定的页面。 |
| CONNECT | HTTP/1.1 协议中预留给能够将连接改为管道方式的代理服务器。 |
| OPTIONS | 允许客户端查看服务器的性能。 |
| TRACE | 回显服务器收到的请求，主要用于测试或诊断 |
| PATCH | 是对 PUT 方法的补充，用来对已知资源进行局部更新 。 |

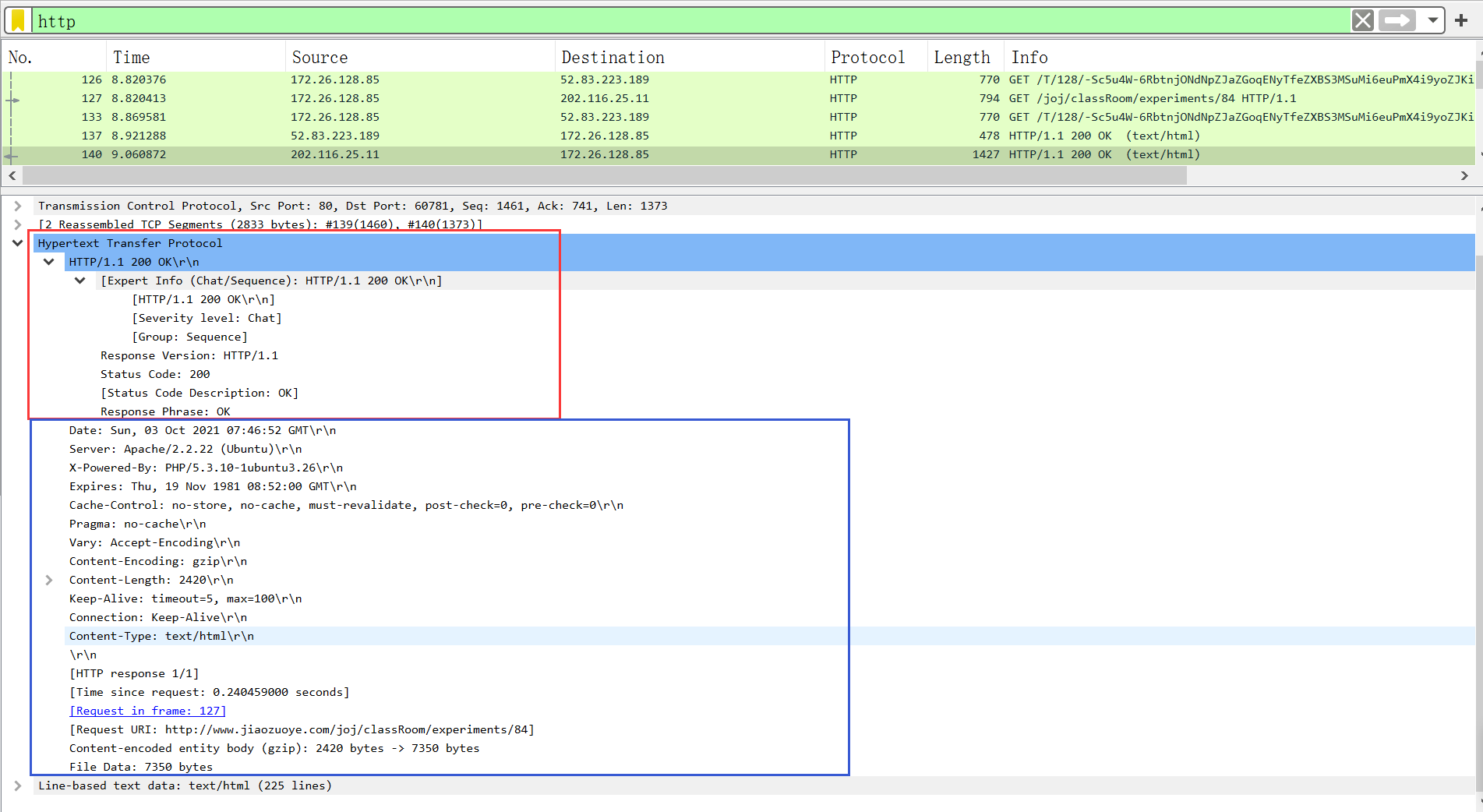




其他请求报文首部信息及解释如下

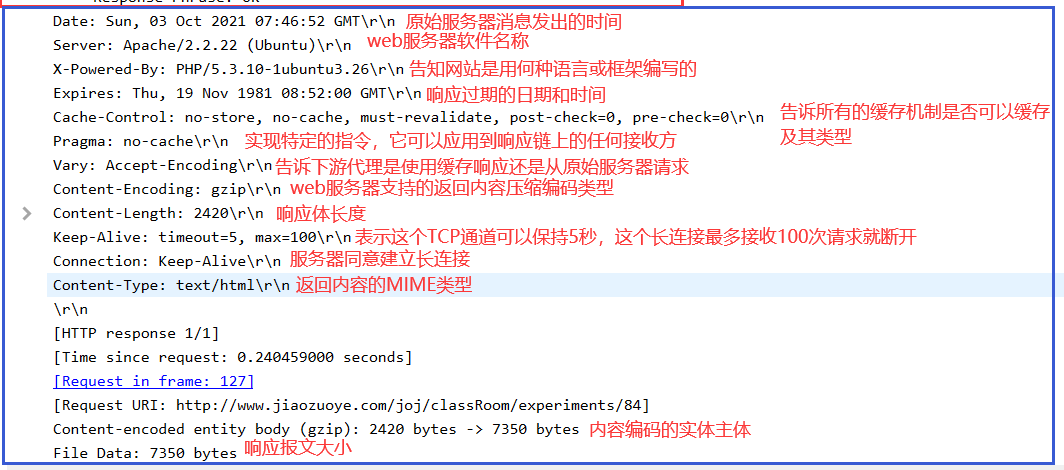
|  |  |
| --- | --- |
| Accept-Charset | 浏览器可以接受的字符编码集 |
| Accept-Ranges | 可以请求网页实体的一个或者多个子范围字段 |
| Authorization | HTTP授权的授权证书 |
| Cache-Control | 指定请求和响应遵循的缓存机制 |
| Content-Length | 请求的内容长度 |
| Content-Type | 请求的与实体对应的MIME信息 |
| Date | 请求发送的日期和时间 |
| Expect | 请求的特定的服务器行为 |
| From | 发出请求的用户的Email |
| If-Match | 只有请求内容与实体相匹配才有效 |
| If-Modified-Since | 如果请求的部分在指定时间之后被修改则请求时间，未被修改则返回304代码 |
| If-None-Match | 如果内容未改变返回304代码，参数为服务器先前发送的Etag，与服务器回应的Etag比较判断是否改变 |
| If-Range | 如果实体未改变，服务器发送客户端丢失的部分，否则发送整个实体。参数也为Etag。 |
| If-Unmodified-Since | 只有实体在指定时间之后未被修改才请求成功 |
| Max-Forwards | 限制信息通过代理和网关传送的时间 |
| Pragma | 用来包含实现特定的指令 |
| Proxy-Authorization | 连接到代理的授权证书 |
| Range | 只请求实体的一部分，指定范围 |
| Referer | 先前网页的地址，当前请求网页紧随其后，即来路 |
| TE | 客户端愿意接受的传输编码，并通知服务器接受尾加头部信息 |
| Upgrade | 向服务器指定某种传输协议以便服务器进行转换（如果支持） |
| Via | 通知中间网关或代理服务器地址，通信协议 |
| Warning | 关于消息实体的警告信息 |

**具体分析响应报文：**





|  |  |
| --- | --- |
| 状态码分类 | 解释 |
| 1\*\* | 通知信息，服务器收到请求，需要请求者继续执行操作 |
| 2\*\* | 成功，操作被成功接收并处理 |
| 3\*\* | 重定向，需要进一步的操作以完成请求 |
| 4\*\* | 客户端错误，请求包含语法错误或无法完成请求 |
| 5\*\* | 服务器错误，服务器在处理请求的过程中发生了错误 |

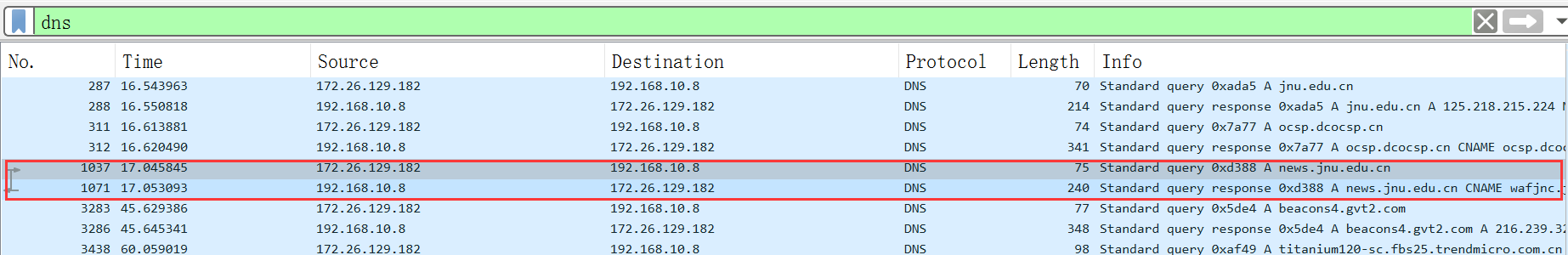


其他响应报文首部信息及解释如下

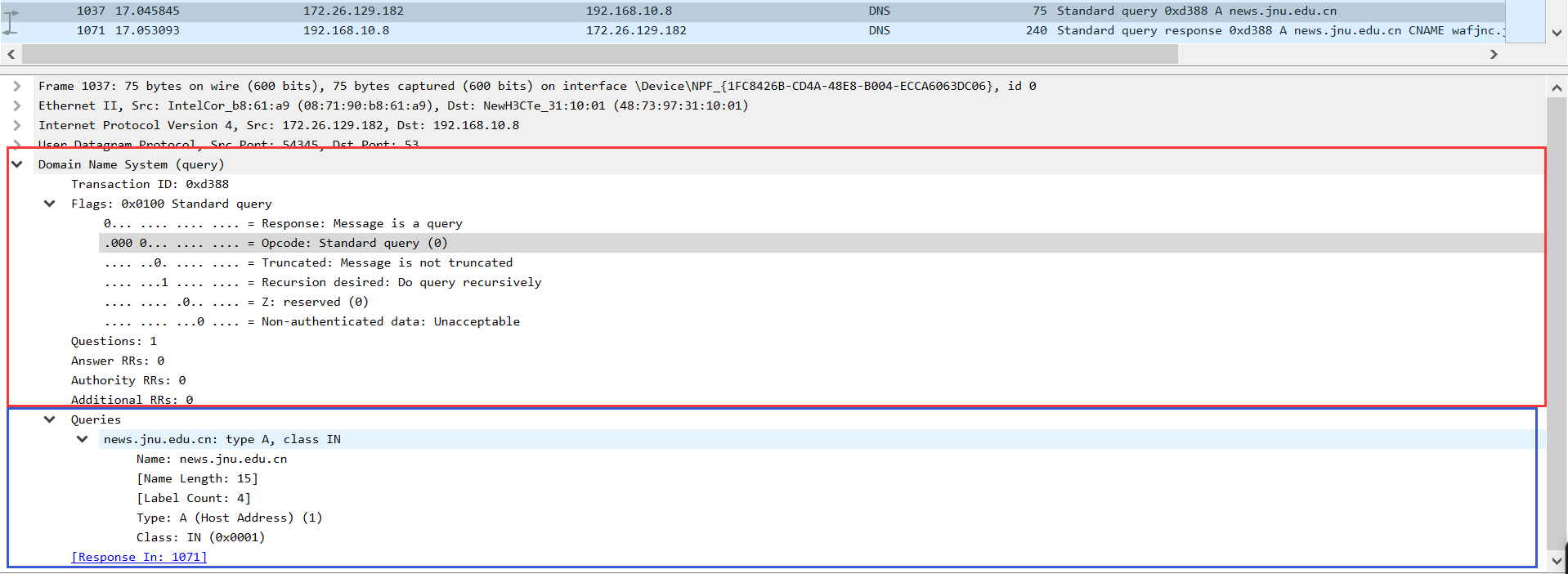
|  |  |
| --- | --- |
| Accept-Ranges | 表明服务器是否支持指定范围请求及哪种类型的分段请求 |
| Age | 从原始服务器到代理缓存形成的估算时间（单位为秒） |
| Allow | 对某网络资源有效的请求行为，不允许则返回405 |
| Content-Language | 响应体的语言 |
| Content-Location | 请求资源可替代的备用的另一地址 |
| Content-MD5 | 返回资源的MD5校验值 |
| Content-Range | 在整个返回体中本部分的字节位置 |
| ETag | 请求变量的实体标签的当前值 |
| Last-Modified | 请求资源的最后修改时间 |
| Location | 用来重定向接收方到非请求URL的位置来完成请求或标识新的资源 |
| Proxy-Authenticate | 指出认证方案和可应用到代理的该URL上的参数 |
| Retry-After | 如果实体暂时不可取，通知客户端在指定时间之后再次尝试 |
| Server | Web服务器软件名称 |
| Set-Cookie | 设置Http Cookie |
| Trailer | 指出头域在分块传输编码的尾部存在 |
| Transfer-Encoding | 文件传输编码 |
| Via | 告知代理客户端响应是通过哪里发送的 |
| Warning | 警告实体可能存在的问题 |
| WWW-Authenticate | 表明客户端请求实体应该使用的授权方案 |
| Retresh | 应用于重定向或一个新的资源被创造，在5秒之后重定向 |

1. **DNS协议分析**

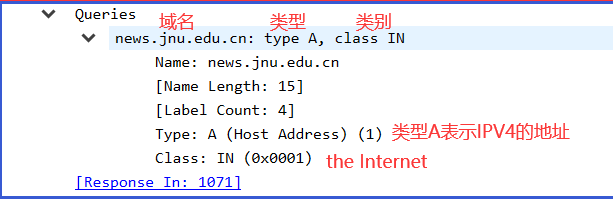
访问news.jnu.edu.cn



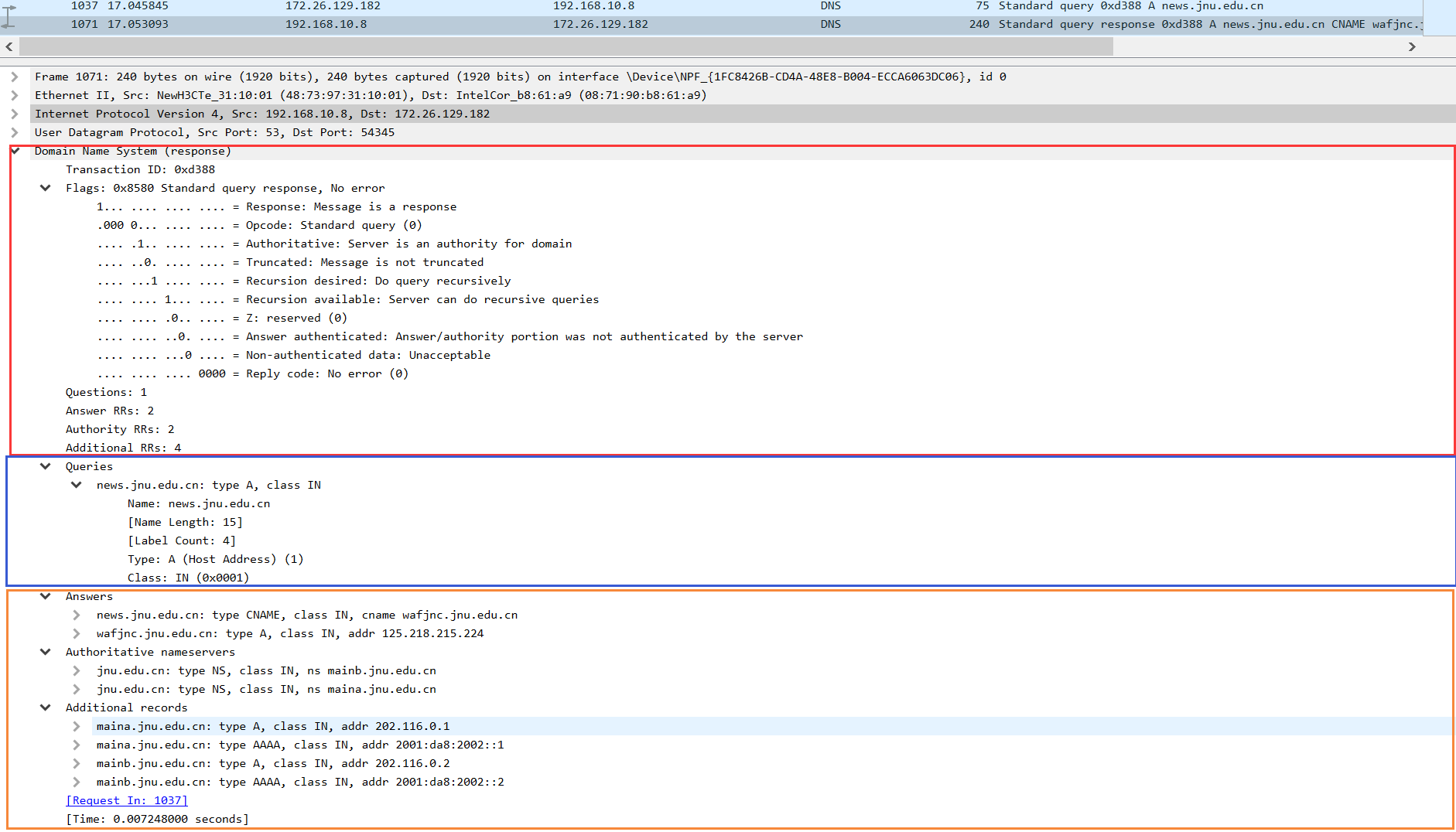
**具体分析DNS请求包：**

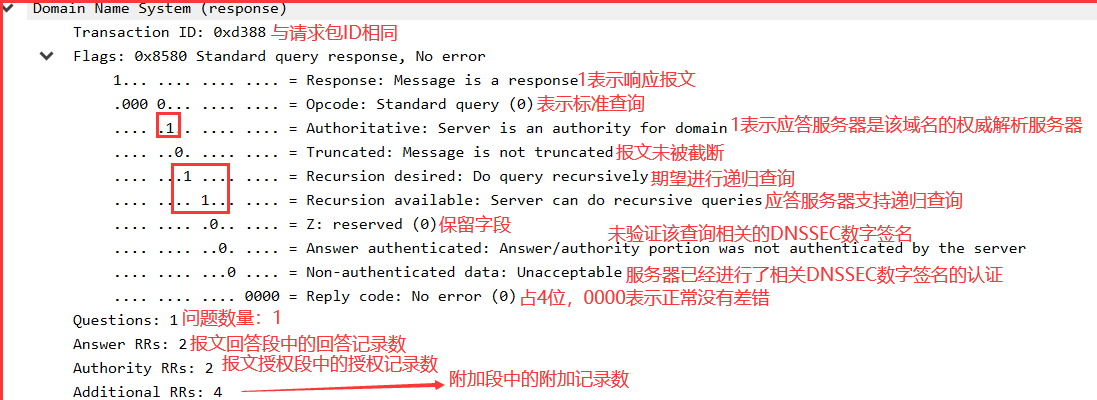


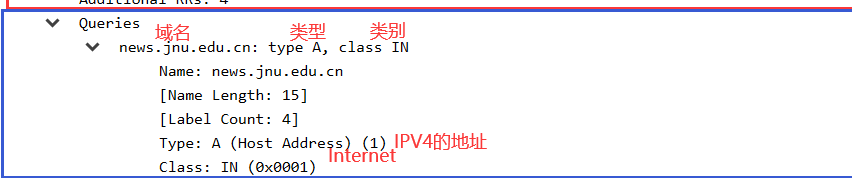




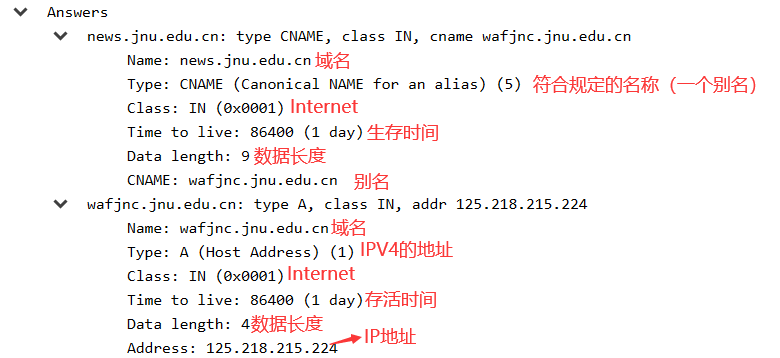
**具体分析DNS响应包：**







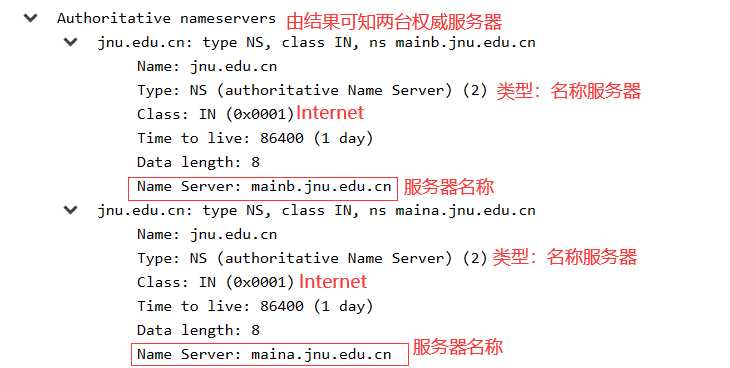
接下来具体分析Answer RRs中的内容：



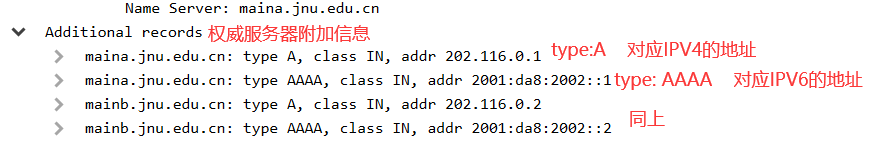
由结果可以看出news.jnu.edu.cn和wafjnc.jnu.edu.cn指向同一个IP地址125.218.215.224，且news.jnu.edu.cn类型为CNAME，即别名记录。这种记录允许将多个名字映射到同一台计算机。

通过查询可知CNAME的作用：可以用于当拥有多个域名需要指向同一服务器IP，此时就可以将一个域名做A记录指向服务器IP，然后将其他的域名作别名(即CNAME)到A记录的域名上；那么当服务器IP地址变更时，就不必对一个一个域名做更改指向了，只需要更改A记录的那个域名到服务器新IP上，其他作为别名（即CNAME）的那些域名的指向将自动更改到新的IP地址上。

接下来具体分析Authority RRs中的内容：



Addition RRs中具体内容：



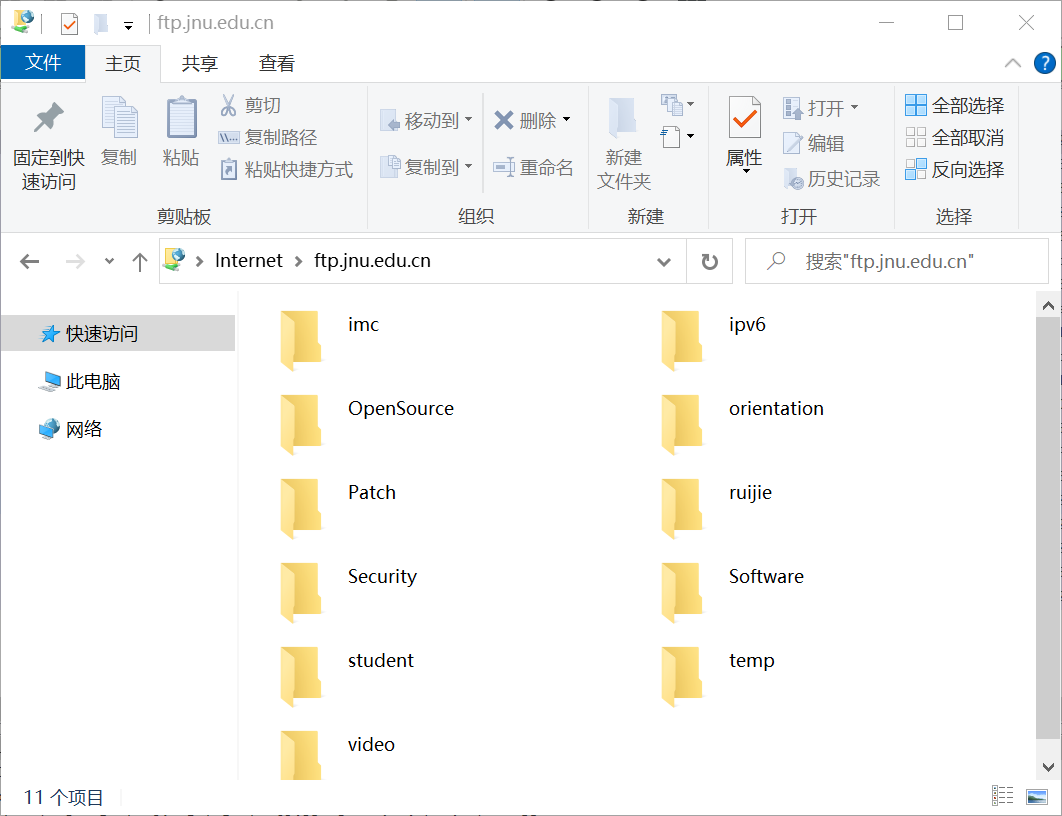
DNS解析news.jnu.edu.cn工作过程为递归查询，向本地域名服务器发出请求查询报文，递归查询到news.jnu.edu.cn与wafjnc.jnu.edu.cn解析到同一个IP地址125.218.215.224。

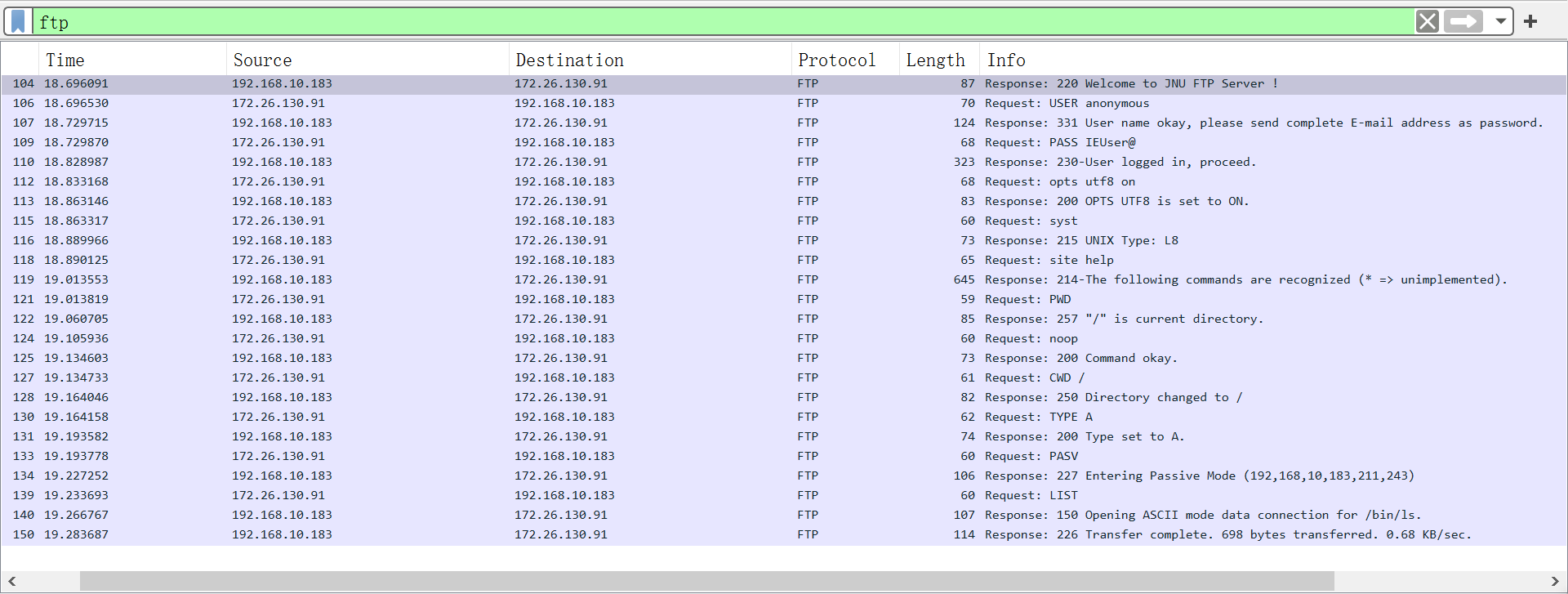
DNS工作过程分为递归查询和迭代查询，主机向本地域名服务器的查询一般采用递归查询，本地域名服务器向根域名服务器的查询通常采用迭代查询。

当访问过一次news.jnu.edu.cn后，再次访问时发现wireshark并没有捕获到DNS数据包，这是因为通过浏览器访问某域名时，浏览器首先会在自己的缓存中查找该域名对应的IP地址，在第二次访问中浏览器有记录。若浏览器缓存中没有，则检查计算机系统hosts文件和本地DNS缓存。当浏览器及系统缓存中均无域名对应IP则进入路由器缓存中检查。当以上3个缓存中都没有时，则进入ISP（互联网服务提供商）DNS缓存中进行查询。当ISP的DNS服务器还找不到时，就对根服务器发起递归查询请求。为提高DNS查询效率，在域名服务器中使用了高速缓存。高速缓存用来存放最近查询过的域名以及从何处获得域名映射信息的记录。

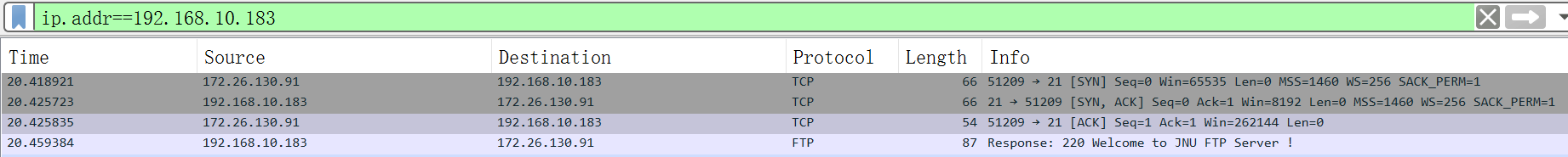
**三、FTP协议分析**

访问FTP服务器，ftp://ftp.jnu.edu.cn

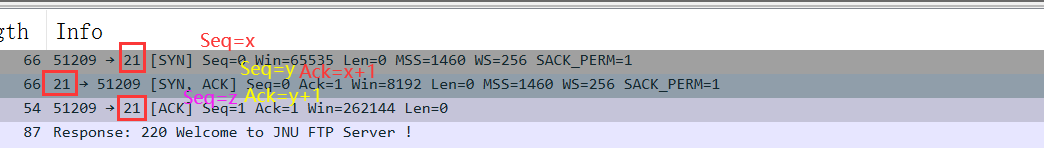




最初在过滤器部分只输入ftp，误以为没有捕获到三次握手的过程，在此次捕获中得知[ftp.jnu.edu.cn](ftp://ftp.jnu.edu.cn)的IP地址为192.168.10.183。后发现FTP的客户和服务器之间需要建立TCP连接，于是更改过滤器条件为ip.addr==192.168.10.183，以此捕获访问ftp的所有数据包。



分析最初捕获到的数据包可知建立FTP连接前需要建立TCP连接，建立一个TCP连接时，需要客户端和服务器总共发送3个包，即三次握手。



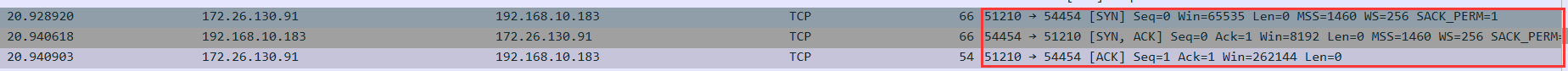
**三次握手**过程：

第1次：客户端发送SYN包(Seq=x)到服务器，并进入SYN\_SEND状态，等待服务器确认；

第2次：服务器收到SYN包，必须确认客户的SYN（Ack=x+1），同时自己也发送一个SYN包（Seq=y），即SYN+ACK包，此时服务器进入SYN\_RECV状态；

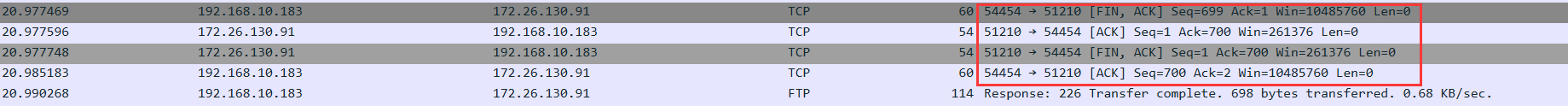
第3次：客户端收到服务器的SYN＋ACK包，向服务器发送确认包ACK(Ack=y+1)，此包发送完毕，客户端和服务器进入ESTABLISHED状态，完成三次握手。

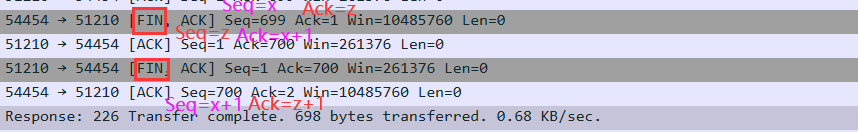
同理建立数据连接时有以下“三次握手”的过程：



与建立连接的“三次握手”类似，断开一个TCP连接则需要“**四次挥手**”。

断开控制数据传送进程时有以下截图





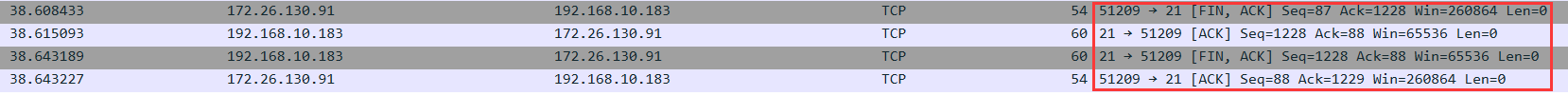
第1次挥手：主动关闭方发送一个FIN包，用来关闭主动方到被动关闭方的数据传送，即主动关闭方告诉被动关闭方：我已经不会再给你发数据了；

第2次挥手：被动关闭方收到FIN包后，发送一个ACK包给主动方，确认序号为收到序号+1

第3次挥手：被动关闭方发送一个FIN，用来关闭被动关闭方到主动关闭方的数据传送，也就是告诉主动关闭方：我的数据也已经发送完毕，不会再给你发数据了；

第4次挥手：主动关闭方收到FIN后，发送一个ACK给被动关闭方，确认序号为收到序号+1

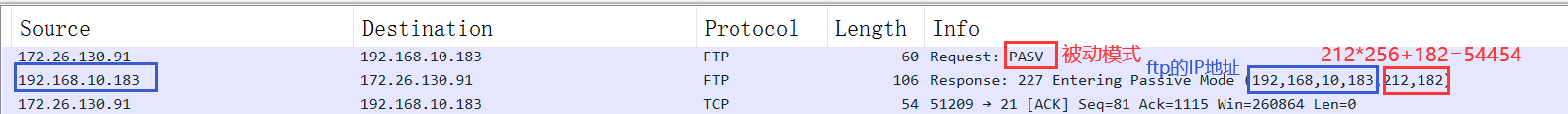
同理断开连接进程时有以下“四次挥手”的截图



综上，在本次实验中：

**客户端控制连接进程的端口号为51209，数据连接的端口号为51210。**

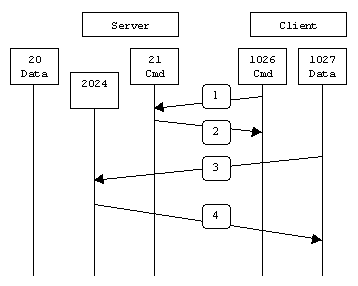
**服务器控制连接进程的端口号为21，数据连接的端口号为54454。**



由捕获到的FTP数据包可知此次访问为PASV被动模式

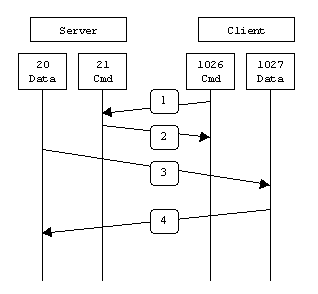
通过计算可知服务器开放的数据连接的端口号为54454

**被动模式PASV：**



控制连接和数据连接都由客户端发起，客户端随机开启一个大于1024的端口N（本次实验为51209）向服务器的21号端口发起连接，同时会开启N+1号（本次实验为51209+1）端口用于数据连接。然后向服务器发送PASV命令，通知服务器自己处于被动模式。服务器收到命令后，会开放一个大于1024的端口P（本次实验为54454）进行监听，然后通知客户端，自己的数据连接端口是P（54454）。客户端收到命令后，会通过N+1号（512010）端口连接服务器的端口P（54454），然后在两个端口之间进行数据传输。

**主动模式PORT：**



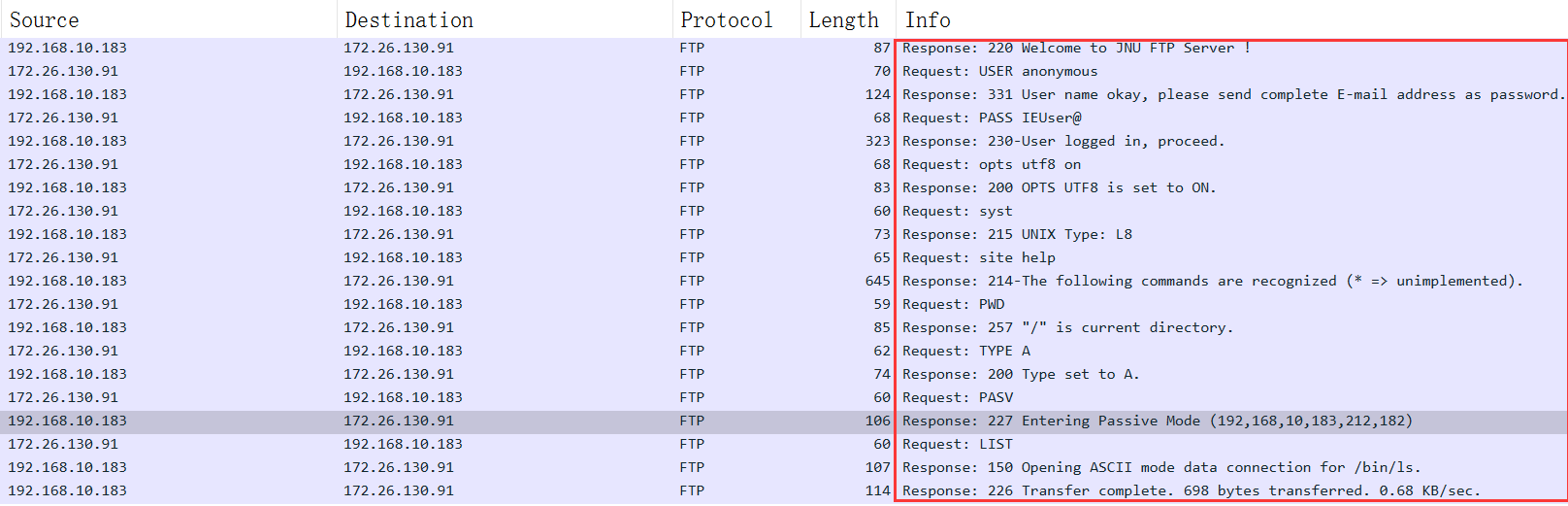
客户端从一个任意的非特权端口N（N>1024）连接到FTP服务器的控制连接端口(即21端口)。紧接着客户端开始监听端口N+1，并发送FTP命令“port N+1”到FTP服务器。最后服务器会从它自己的数据端口（20）连接到客户端指定的数据端口（N+1），这样客户端就可以和ftp服务器建立数据连接通道了。

主动模式PORT和被动模式PASV最主要区别就是数据端口连接方式不同。

主动模式对FTP服务器的管理有利，但对客户端的管理不利。因为FTP服务器企图与客户端的高位随机端口建立连接，而这个端口很有可能被客户端的防火墙阻塞。

被动模式对FTP客户端的管理有利，但对服务器端的管理不利。因为客户端要与服务器端建立两个连接，其中一个连到一个高位随机端口，而这个端口很有可能被服务器端的防火墙阻塞。

过滤FTP协议进行分析：





**四、 SMTP和POP协议分析**

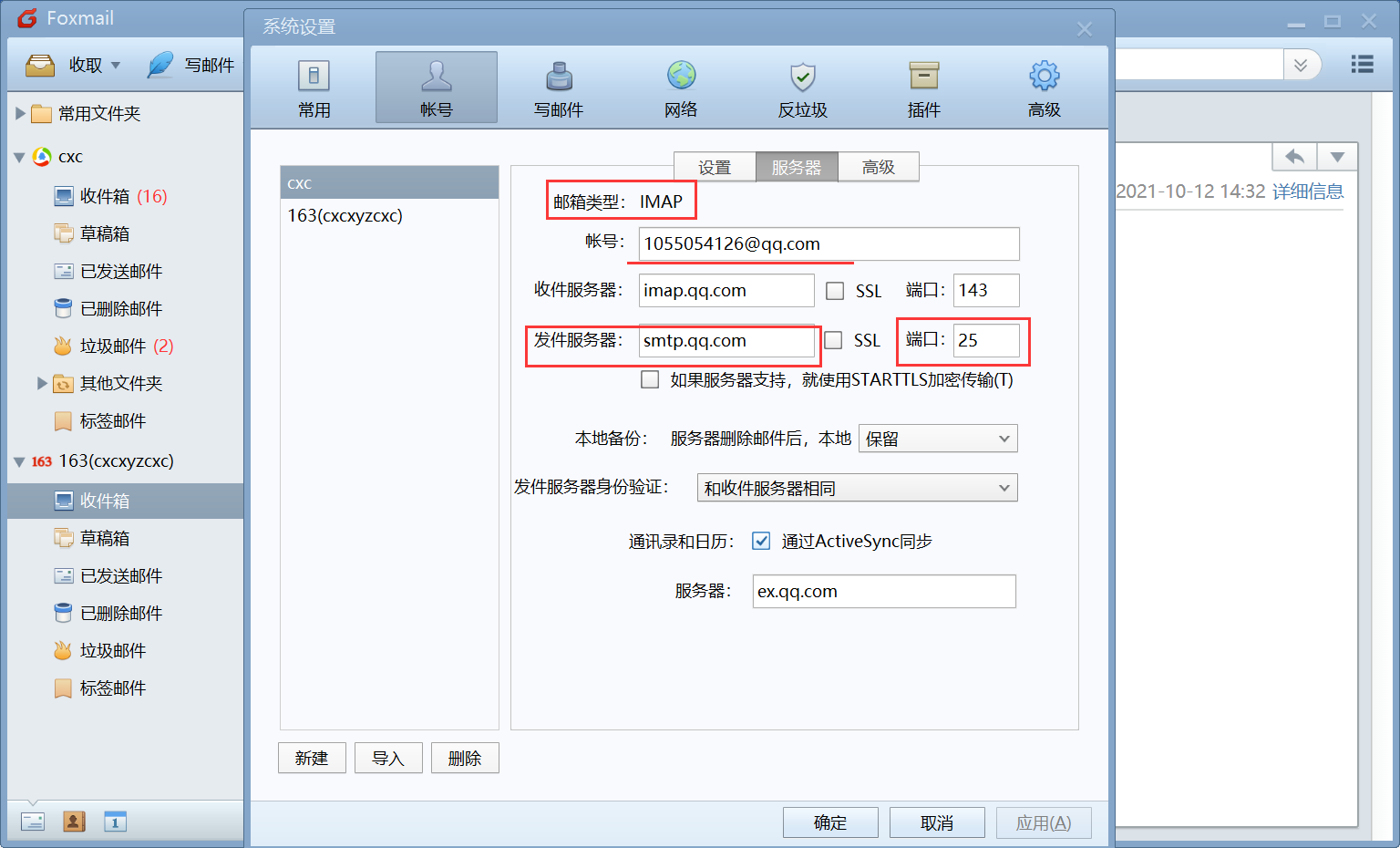
首先下载客户端的邮件软件Foxmail

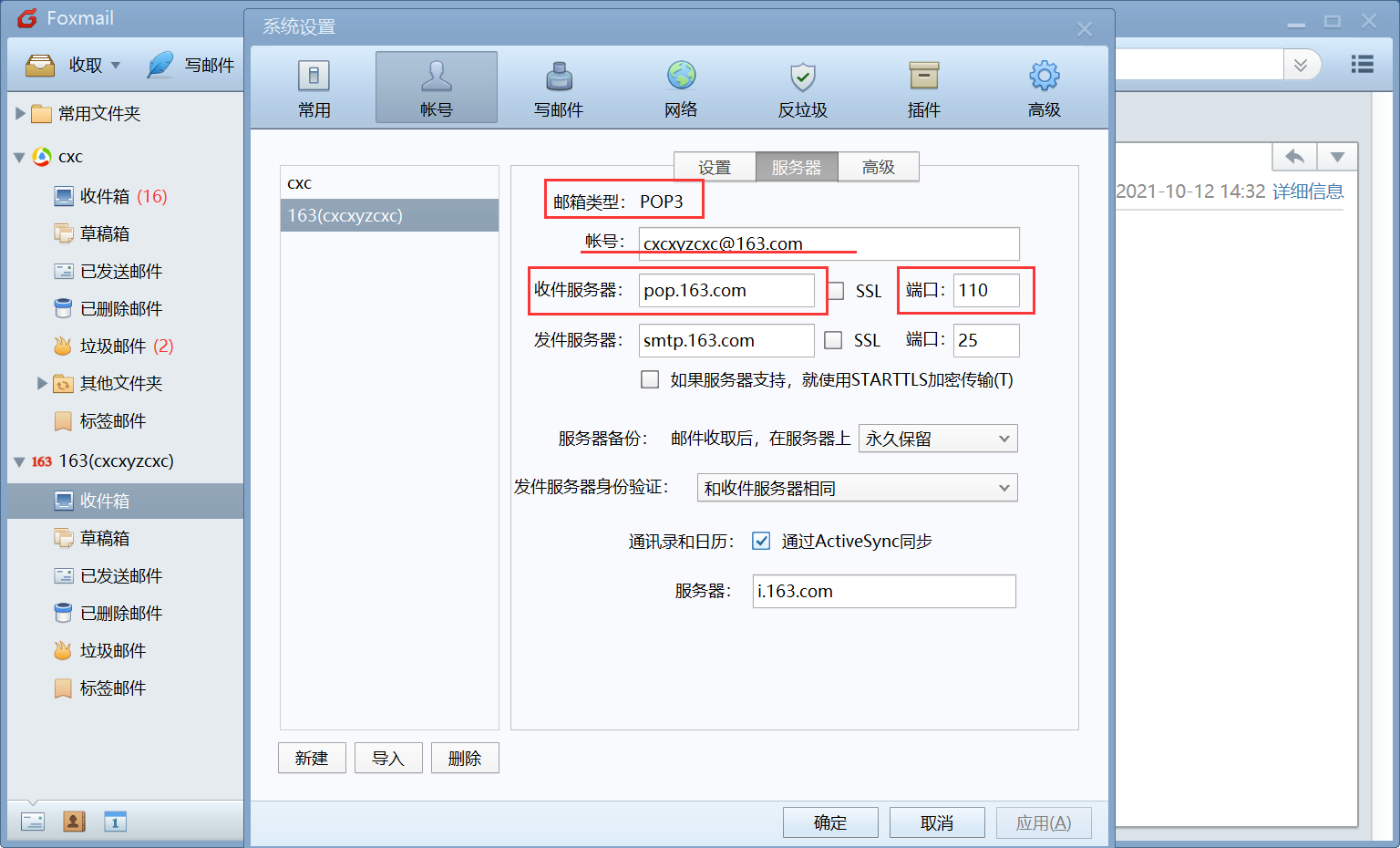
为捕获到SMTP和POP数据包，需开启相应服务



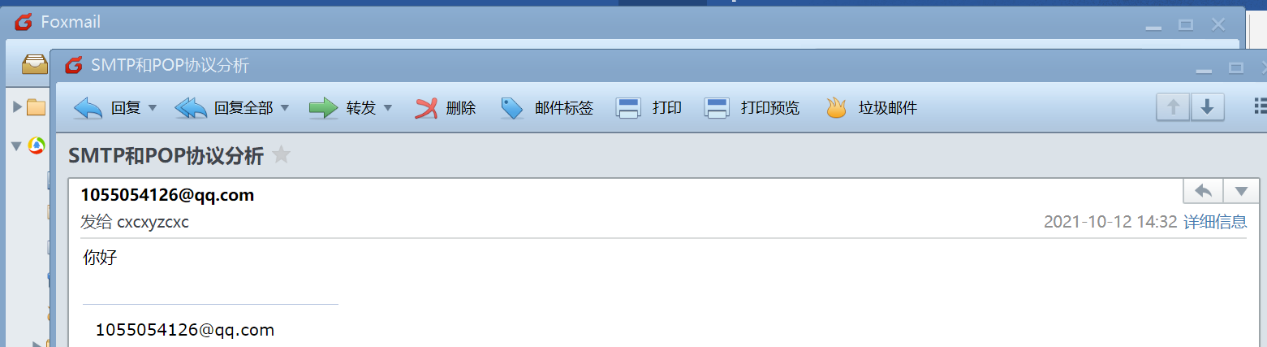


再在Foxmail中绑定我的qq邮箱和163网易邮箱账号（本次实验从qq邮箱发送邮件到163网易邮箱），配置如下





本次实验从qq邮箱发送邮件到163网易邮箱



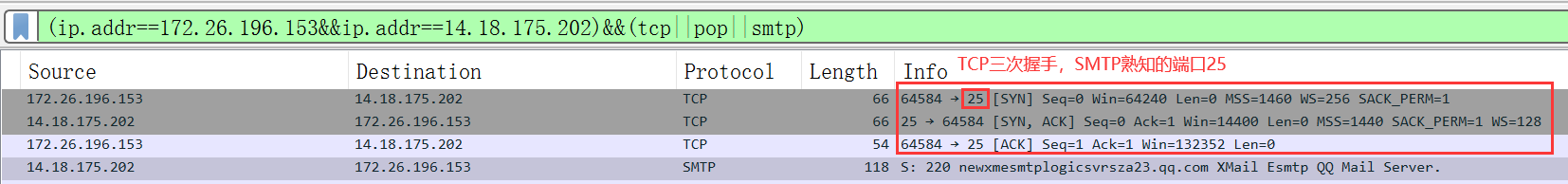
在wireshark上捕获相关数据包

**发送邮件时：**

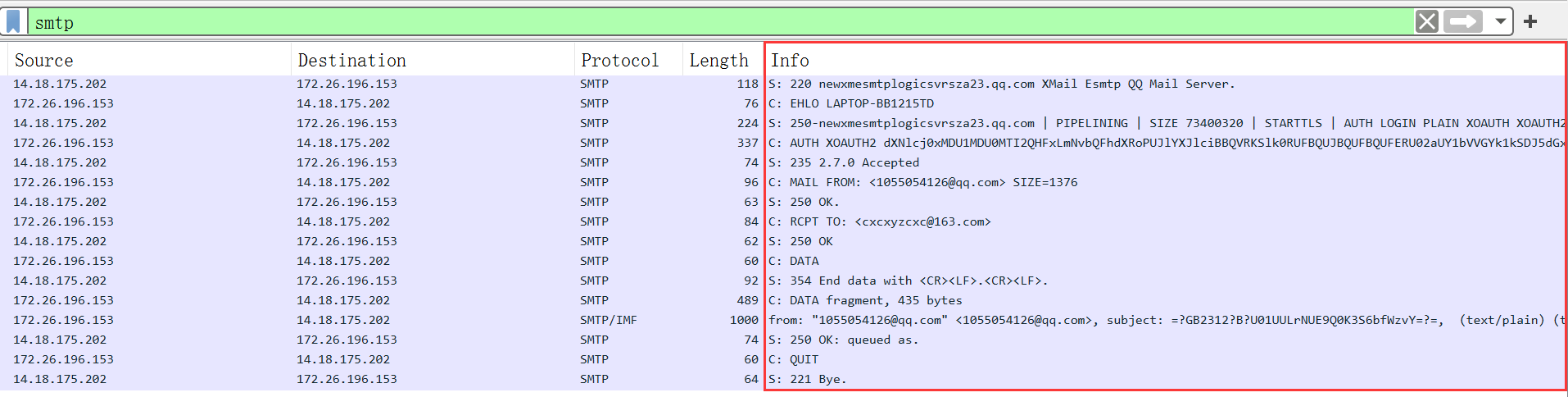
过滤器：(ip.addr==172.26.196.153&&ip.addr==14.18.175.202)&&(tcp||pop||smtp)

发送方邮件服务器向接收方邮件服务器发送邮件使用SMTP协议

TCP三次握手，连接建立：



过滤器条件改为smtp，分析邮件传送过程

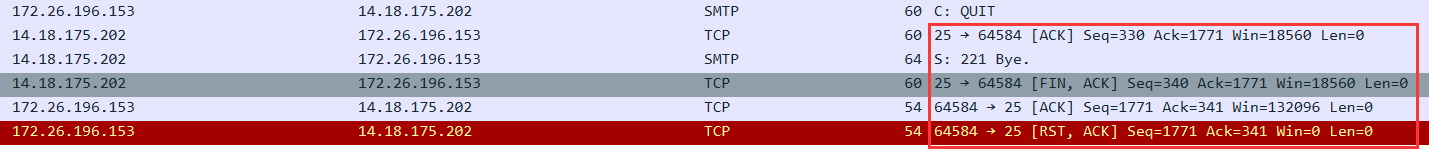




邮件内容：



TCP四次挥手，连接释放：

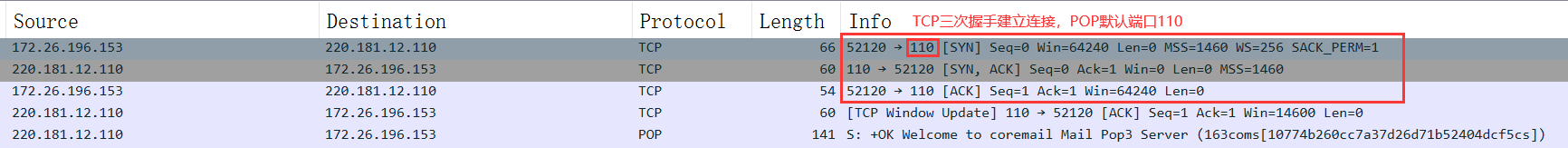


**接收邮件时：**

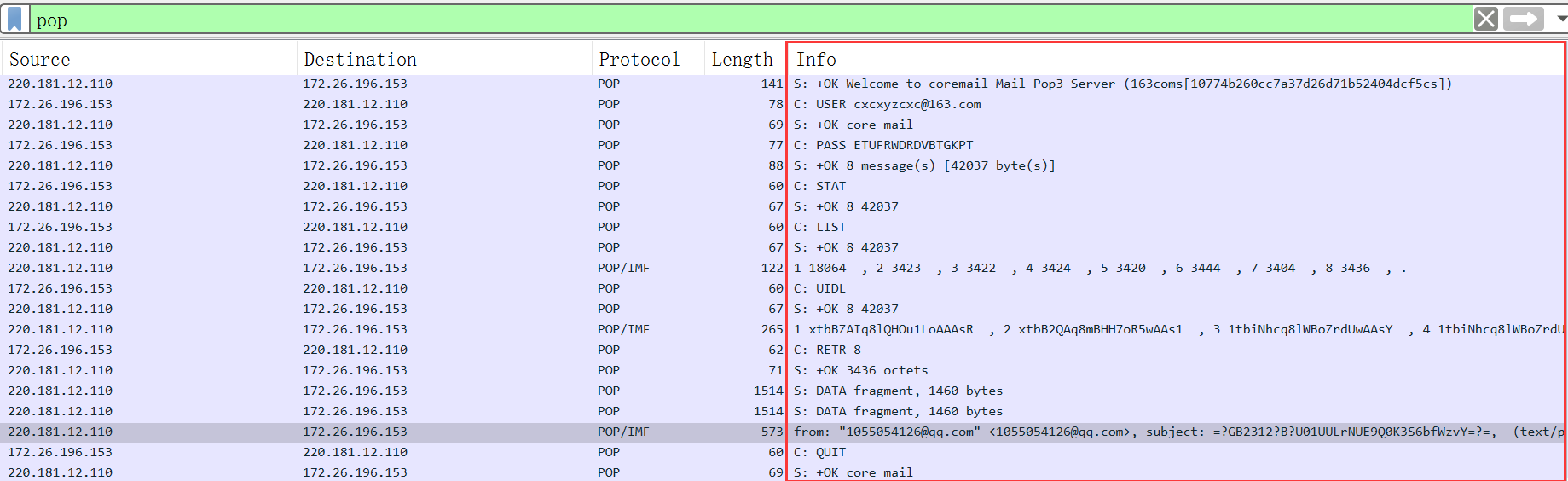
过滤器：(ip.addr==172.26.196.153&&ip.addr==220.181.12.110)&&(tcp||pop||smtp)

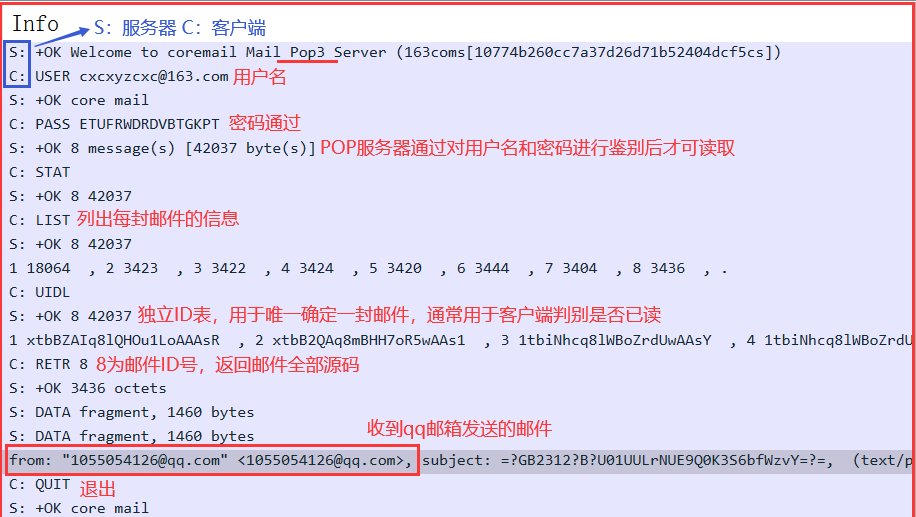
POP3是用户代理从接收方邮件服务器上读取邮件所使用的协议。

TCP三次握手建立连接

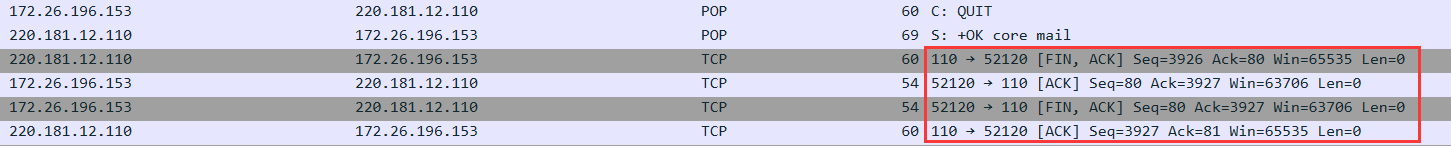


过滤器条件改为POP，分析POP工作过程及协议



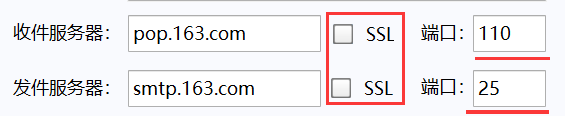
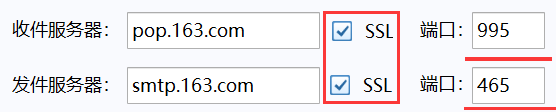


TCP四次挥手，断开连接



当邮件收件方的服务器设置的是SMTP或POP3协议且不加密时才能捕获到相关数据包。

实验中发现，在web直接发邮件或者配置邮箱服务器时使用了SSL协议，就捕获不到POP或SMTP数据包。

如图，在Foxmail中配置邮件服务器时勾选SSL则POP和SMTP不再使用默认端口，经过试验，勾选SSL后再次从qq邮箱发送邮件到163网易邮箱，wireshark中未能捕获到POP或SMTP数据包。

通过查询可知，SSL是Secure Sockets Layer 安全套接字协议，是一个不依赖于平台和运用程序的协议，位于TCP/IP协议与各种应用层协议之间，为数据通讯提供安全支持。SSL协议可分为两层： SSL记录协议（SSL Record Protocol）：它建立在可靠的传输协议（如TCP）之上，为高层协议提供数据封装、压缩、加密等基本功能的支持。 SSL握手协议（SSL Handshake Protocol）：它建立在SSL记录协议之上，用于在实际的数据传输开始前，通讯双方进行身份认证、协商加密算法、交换加密密钥等。

1. **实验总结**

通过HTTP协议分析实验，进一步理解HTTP的工作过程，了解HTTP请求方法GET、HEAD、POST等，同时知道HTTP协议请求和响应拥有许多首部信息，通过查询学习了不同首部信息及其解释。

通过DNS协议分析实验，对DNS请求和响应包进行了详细理解和分析，进一步理解了递归查询和迭代查询，在实验过程中对于域名类型为CNAME的有进一步学习，得知其用于多个域名指向同一服务器IP，同时对于捕获不到DNS的情况进行分析，得知其一步步查询缓存的过程。

通过FTP协议分析实验，捕获数据包分析学习了TCP“三次握手”建立连接和“四次挥手”断开连接的具体过程。实验中产生为什么建立连接协议是三次握手，而断开连接却是四次挥手的疑问，通过查询得知这是因为服务端的LISTEN状态下的SOCKET当收到SYN报文的连接请求后，它可以把ACK和SYN(ACK起应答作用，而SYN起同步作用)放在一个报文里来发送。但断开连接时，当收到对方的FIN报文通知时，它仅仅表示对方没有数据发送给你了；但未必所有的数据都全部发送完成了，所以可能未必会马上会关闭SOCKET,也即你可能还需要发送一些数据给对方之后，再发送FIN报文给对方来表示你同意现在可以关闭连接了，所以它这里的ACK报文和FIN报文多数情况下都是分开发送的，因此关闭连接发送4个数据包。

FTP协议分析实验中，观察到数据连接和控制连接开放的端口不同，也学习了主动模式PASV和被动模式PORT，它们最主要区别就是数据端口连接方式不同。同时，进一步掌握ftp命令及对应服务器的响应信息。

通过SMTP和POP协议分析实验，配置了邮箱收件服务器和发件服务器，进一步理解了SMTP和POP协议细节，同时得知当使用SSL安全套接字协议进行加密时，捕获不到SMTP和POP数据包，进一步保证了通信安全。