北京邮电大学实验报告

<table>
<tr><td>实验名称</td><td colspan="2">计算机网络实验一</td><td>学 院</td><td>网络空间
安全学院</td><td>指导教师</td><td>杨震</td></tr>
<tr><td>班 级</td><td>班内序号</td><td colspan="2">学 号</td><td>学生姓名</td><td colspan="2">成绩</td></tr>
<tr><td>2023211805</td><td>24</td><td colspan="2">2023211595</td><td>李昊伦</td><td colspan="2"></td></tr>
<tr><td>实
验
内
容</td><td colspan="6">1、使用 Wireshark 软件捕获 HTTP 消息，分析其消息头，理解 HTTP 的通信原理；
2、使用 Wireshark 软件捕获一次从客户端发送 Email 的过程，分析 SMTP 消息，理解 Email 系统中发送邮件的通信原理；
3、使用 Telnet 软件访问 Email 服务器，输入 SMTP 命令与 Email 服务器交互，理解 SMTP 的通信过程和 Base64 编码的概念。</td></tr>
<tr><td>学生
实验
报告
(附页)</td><td colspan="6">《详见下方“实验报告”》</td></tr>
<tr><td>实
验
成
绩
评
定</td><td colspan="6">评语：

成绩：
指导教师签名：

年 月 日</td></tr>
</table>

注：评语要体现每个学生的工作情况，可以加页。

实验一 应用层协议消息的捕获和解析

1 实验内容和实验环境

1.1 实验目的

深入理解典型的应用层协议——HTTP 和 SMTP 的要点。

1.2 实验内容

（1）使用 Wireshark 软件捕获 HTTP 消息，分析其消息头，理解 HTTP 的通信原理；
（2）使用 Wireshark 软件捕获一次从客户端发送 Email 的过程，分析 SMTP 消息，理解 Email 系统中发送邮件的通信原理；
（3）使用 Telnet 软件访问 Email 服务器，输入 SMTP 命令与 Email 服务器交互，理解 SMTP 的通信过程和 Base64 编码的概念。

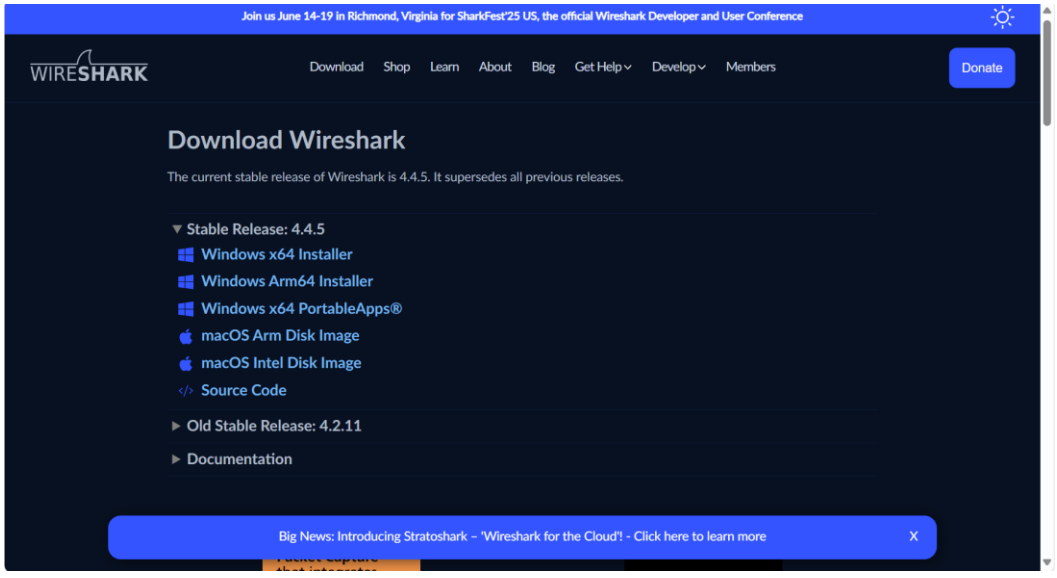
1.3 实验环境

一台装有 MS Windows 系列操作系统的计算机，能够连接到因特网，并安装 Wireshark 软件。

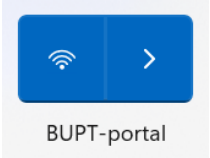
2 实验步骤

2.1 实验装备工作

（1）下载 Wireshark 软件并了解其功能和使用方法。



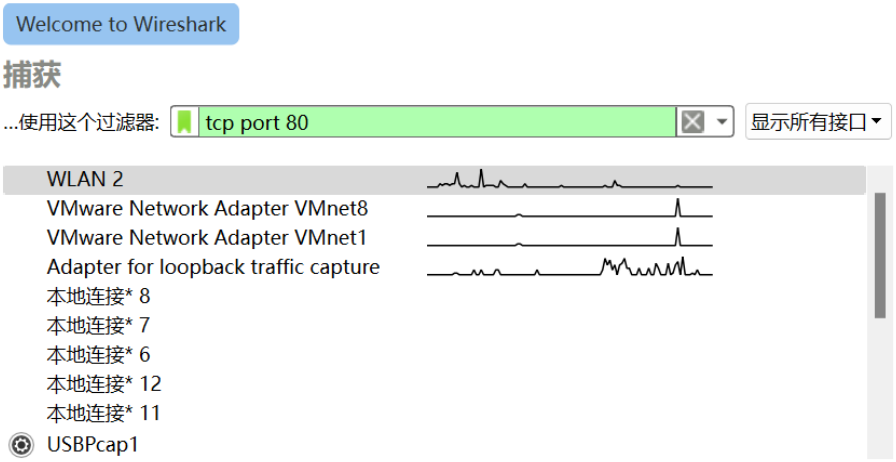
（2）确保计算机已经连接到网络。



2.2 捕获 HTTP 协议数据

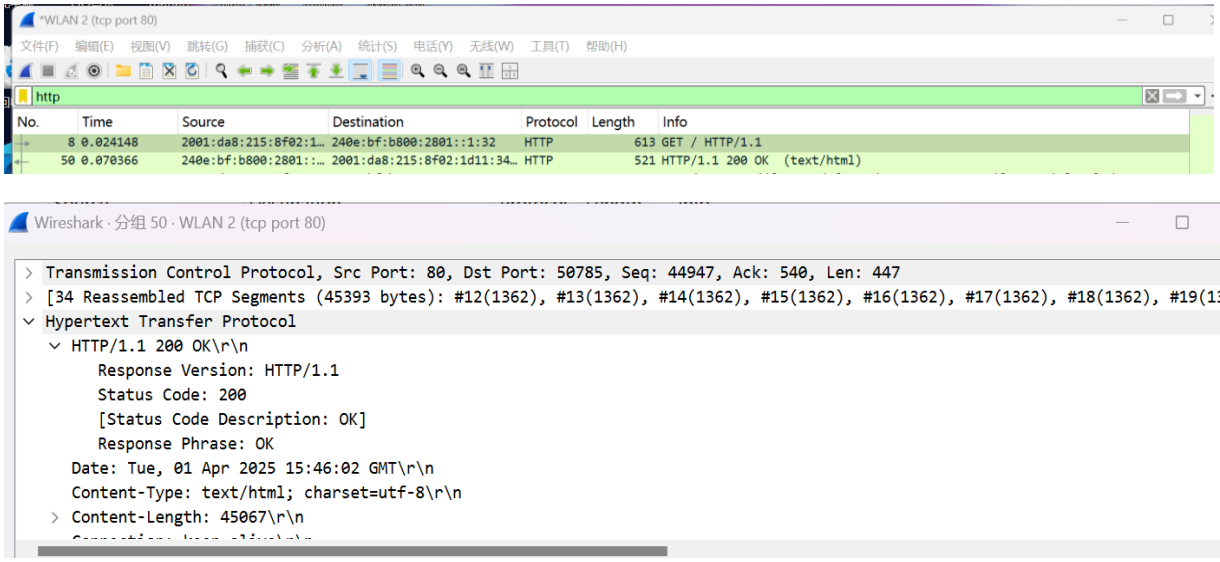
（1）清除浏览器 cookie 数据。

（2）启动 Wireshark，选择捕获接口为 WLAN，设置捕获过滤器为 tcp port 80。



（3）用浏览器打开 http://www.xinhuanet.com/

（4）抓包捕获 HTTP 协议数据。

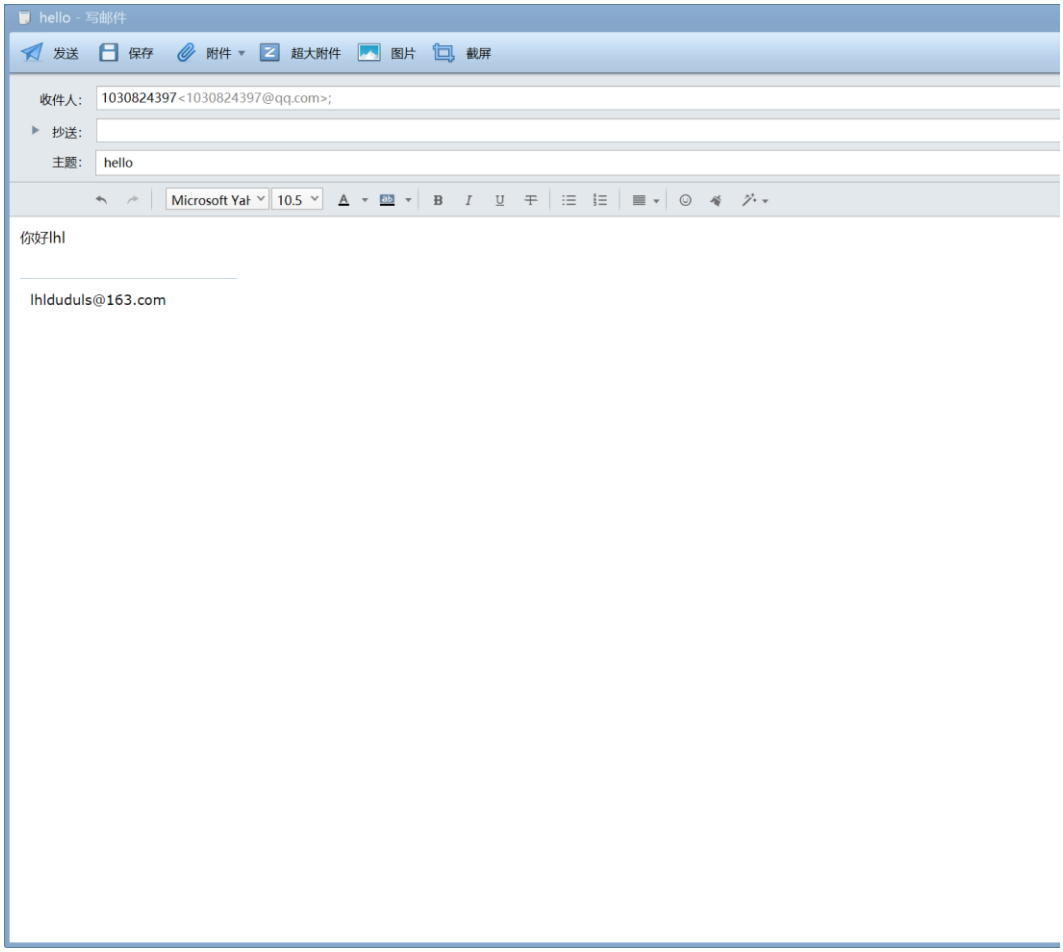


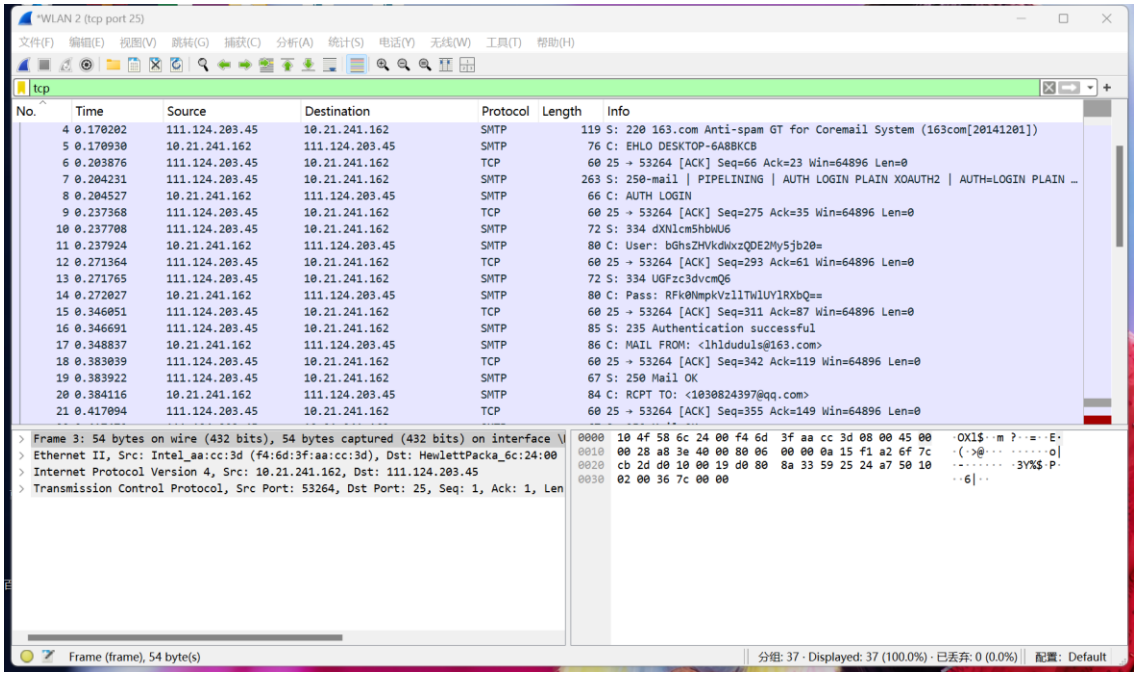
2.3 捕获 SMTP 协议数据

（1）下载并安装邮件客户端软件 Foxmail，配置用户账户，设置发件服务器

（2）配置 Wireshark，设置捕获过滤器为 tcp port 25

（3）开始捕获，用 Foxmail 发送一封邮件，邮件发送成功后停止捕获。



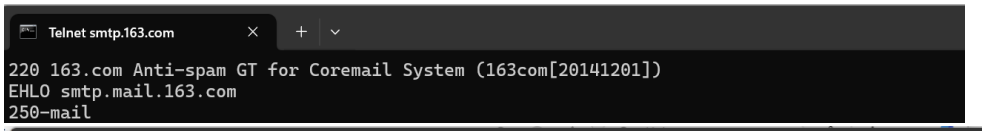


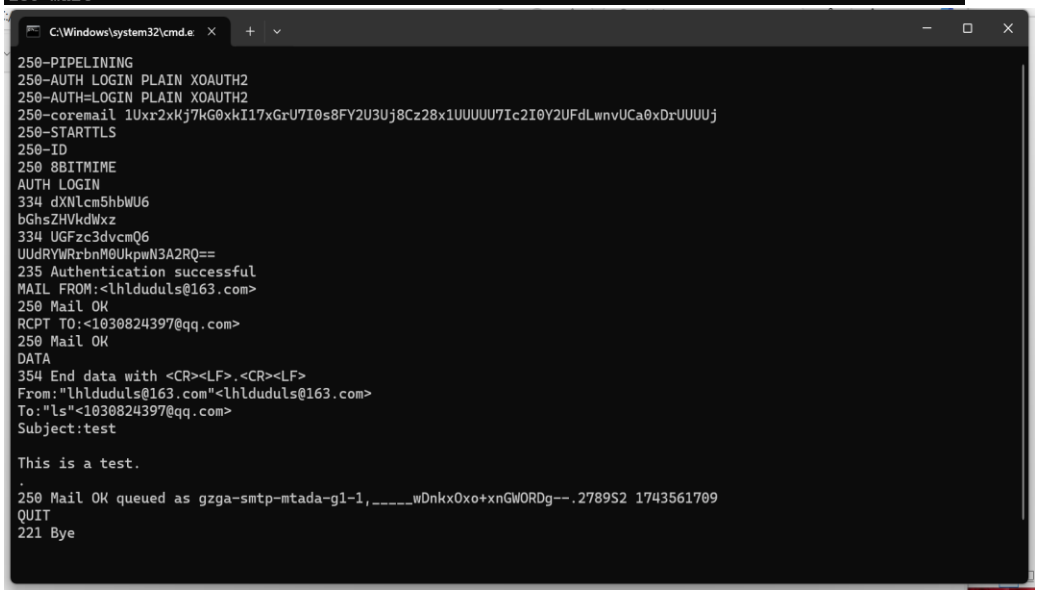
2.4 使用 SMTP 命令与邮件服务器交互

（1）在命令行模式，使用 telnet 程序连接到发件服务器

（2）在新窗口中，输入 SMTP 命令，与邮件服务器交互.

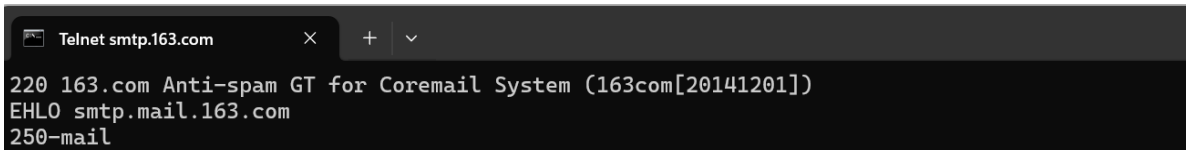
以下是完整过程。

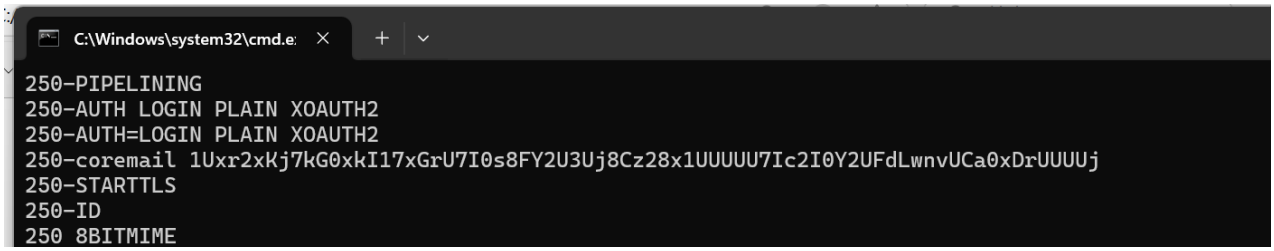




下面对每个过程分析：

首先为通知服务器用户身份





输入经过 Base64 编码加密过的发件人地址和授权码

指明发件人和收件人身份，写正文

结束会话

之后可在收件人邮箱中发现刚刚发的邮件

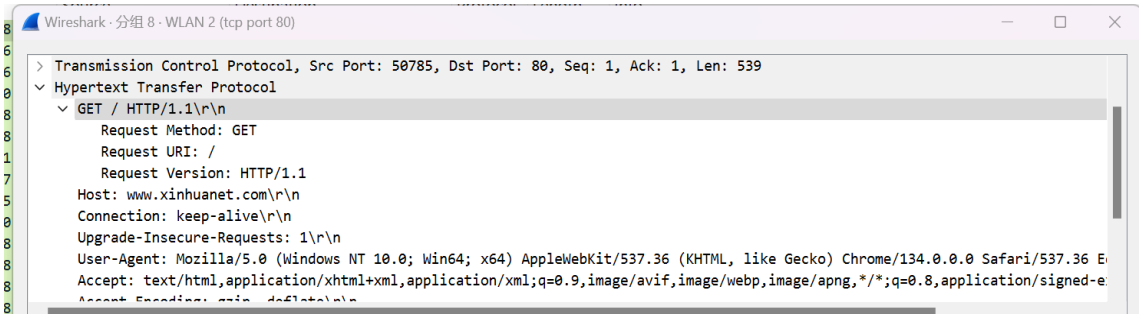
3 协议分析

3.1 HTTP 协议分析

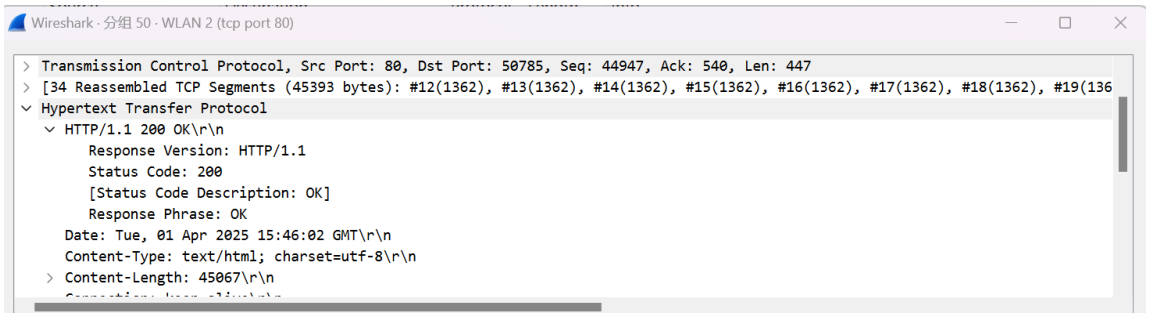
（1）设置了 tcp port 80，捕捉到的数据没有 DNS 类型的请求。浏览器先分析 URL 得到域名，再根据 DNS 得到新华网服务器的 IP 地址。

（2）然后发生了 TCP 连接过程。发生了 TCP 三次握手，建立了本机和新华网服务器的 TCP 连接。

（3）建立的 TCP 连接后，浏览器向服务器发送了 HTTP 类型的 GET 请求。GET 为 http 的方法，用于从服务器上下载 URL 对应的网页，接着每个 HTTP 消息头及其值如下图：



（4）新华网的服务器接收到请求后，对于请求进行回应 200 OK，其中包含了网页文件的数据，消息头及其值如下图：



（5）请求与应答消息中的各个字段与消息头的功能列表如下：

Host	请求	Web 服务器的域名
Cache-Control	请求	指定请求和响应遵循的缓存机制
Connection	两个消息都有	TCP 连接的类型
Upgrade	两个消息都有	发送方要使用的协议
User-Agent	请求	浏览器的类型
Accept	请求	浏览器能处理的网页类型
Accept-Encoding	请求	浏览器能处理的网页编码类型
Accept-Language	请求	浏览器能处理的自然语言
Server	响应	Web 服务器的类型
Date	两个消息都有	消息发送的日期时间
Content-type	响应	网页的 MIME 类型
Expires	响应	指定一个日期/时间
Transfer-Encoding	响应	表示实体传输给用户的编码形式
Location	响应	通知客户将请求发送给别的服务器
Content-Encoding	响应	内容的编码类型
Content-Language	响应	网页中的自然语言类型

3.2 SMTP 协议分析

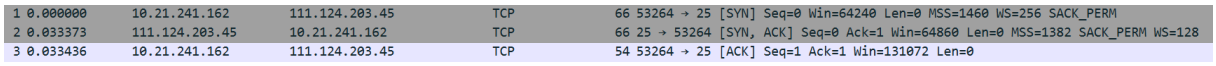
SMTP 命令功能

命令	功能
EHLO	通知发件人的用户身份
AUTH	指出本地 SMTP 虚拟服务器支持 SMTP 身份验证服务扩展
MAIL FROM	通知服务器写信人的邮件地址，并开始邮件服务
RCPT TO	通知收信人地址
DATA	通知邮件正文开始
QUIT	要求关闭 TCP 连接

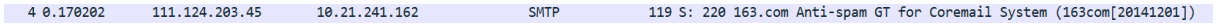
SMTP 状态码

状态码	含义
220	可以提供邮件服务
250	命令成功执行
235	认证通过
354	通知客户端开始发送邮件，以只包含“.”的一行作为结束
221	服务器端结束传输，关闭 TCP 连接

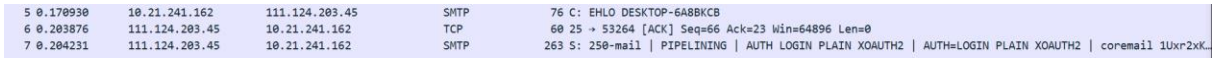
（1）根据 ACK 和 SYN 确定第 1、2、3 行为客户与邮件服务器之间建立 TCP 连接。



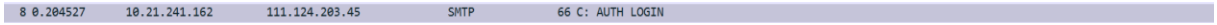
（2）第 4 行的数据是为服务器的回复：



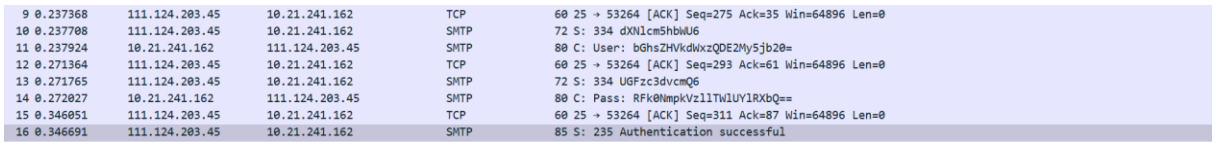
（3）第 5 行客户端接收到后回复 EHLO 通知发件人的邮件服务器域名，再加上操作 PC 信息，之后服务器端回复 250 表示连接成功



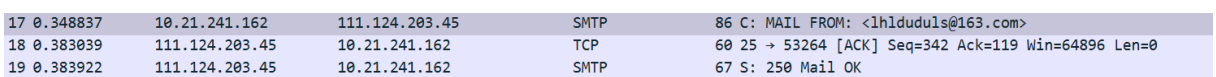
（4）第 8 行为用户发送 AUTH 用户登录命令给服务器。



（5）用户通过使用用户名和密码登录服务器，第 16 行中服务器回复 235，表示登陆成功。



（6）第 17 行是客户端发送 MAIL FROM，声明写信人的邮件地址，准备传输。之后第 19 行服务器回复 250 OK 表示命令成功。



（7）第 20 行客户端发送 RCPT TO 命令，注明收信人的地址。之后第 22 行服务器回

复 250 OK 表示命令成功。

20 0.384116	10.21.241.162	111.124.203.45	SMTP	84 C: RCPT TO: <1030824397@qq.com>
21 0.417094	111.124.203.45	10.21.241.162	TCP	60 25 → 53264 [ACK] Seq=355 Ack=149 Win=64896 Len=0
22 0.417470	111.124.203.45	10.21.241.162	SMTP	67 S: 250 Mail OK

（8）第 23 行客户端发送 DATA 命令通知正文开始，之后服务器回复 354，通知客户端可以发送正文了。

23 0.417826	10.21.241.162	111.124.203.45	SMTP	60 C: DATA
24 0.452887	111.124.203.45	10.21.241.162	TCP	60 25 → 53264 [ACK] Seq=368 Ack=155 Win=64896 Len=0
25 0.453333	111.124.203.45	10.21.241.162	SMTP	91 S: 354 End data with <CR><LF>.<CR><LF>

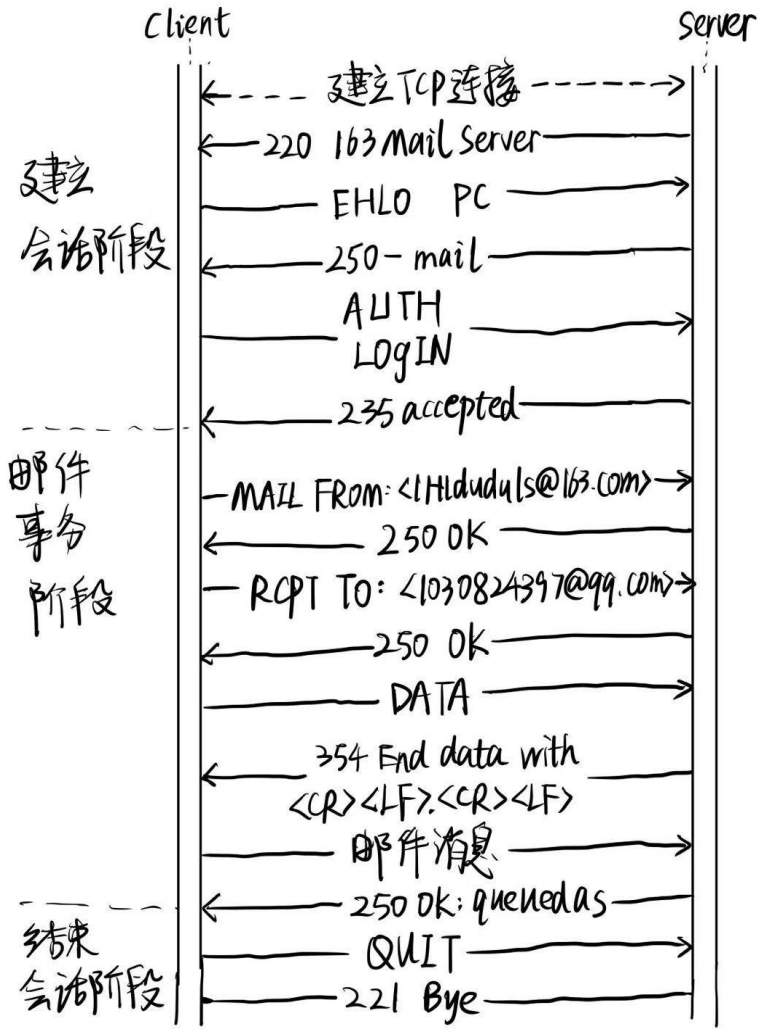
（9）第 26 行客户端开始发送正文，并标明正文的大小，之后第 30 行服务器发送 250 OK 表示确认收到了邮件。

26 0.453519	10.21.241.162	111.124.203.45	SMTP	458 C: DATA fragment, 404 bytes
27 0.526943	111.124.203.45	10.21.241.162	TCP	60 25 → 53264 [ACK] Seq=405 Ack=559 Win=64512 Len=0
28 0.526960	10.21.241.162	111.124.203.45	SMTP/IMF	1008 from: "lhlduduls@163.com" <lhlduduls@163.com>, subject: hello, (text/plain) (text/html) | .
29 0.560183	111.124.203.45	10.21.241.162	TCP	60 25 → 53264 [ACK] Seq=405 Ack=1513 Win=64128 Len=0
30 0.571668	111.124.203.45	10.21.241.162	SMTP	142 S: 250 Mail OK queued as gzga-smtp-mtada-g1-3,_____wD3agDVGuxncMW9DQ--.53672S2 1743526613

（10）第 31 行客户端发送 Quit 命令要求关闭 TCP 连接，之后 33 行显示服务器返回 221 Bye 表示服务器段结束传输，关闭 TCP。至此完成邮件传输。

31 0.574348	10.21.241.162	111.124.203.45	SMTP	60 C: QUIT
32 0.609060	111.124.203.45	10.21.241.162	TCP	60 25 → 53264 [ACK] Seq=493 Ack=1519 Win=64128 Len=0
33 0.609350	111.124.203.45	10.21.241.162	SMTP	63 S: 221 Bye

（11）画出通信的过程



4 实验结论与实验心得

4.1 实验结论

两个应用层协议在运作时，首先都需要建立 TCP 连接，这就说明了下层为上层提供服务且脱离了传输层，应用层是无法运作的。

4.2 实验心得

通过以上实验，我深入了解了 HTTP 和 SMTP 两个协议的通信原理，并学习了如何使用 Wireshark 和 Telnet 工具进行网络数据包捕获和交互式通信。

在实验过程中，我也遇到了一些问题。在做捕获 SMTP 协议数据实验中，我登录账号时输入的密码多次输入错误，经过一番研究后发现需要输入的是授权码；在进行使用 SMTP 命令与邮件服务器交互的实验中，我多次发送失败，结果发现是输入格式错误。

在分析 HTTP 消息之后，我了解到 HTTP 是一种无状态的协议，客户端与服务器之间通过请求和响应来进行通信。通过分析消息头，我了解到 HTTP 消息头中包含许多元数据，如请求的方法、URI、请求头、响应状态码等，这些元数据对于实现有效的网络通信非常重要。

在分析 SMTP 消息之后，我认识到 SMTP 是一种用于发送邮件的协议，它是基于文本的协议，包含邮件头和邮件主体。通过分析 SMTP 消息，我了解到邮件的发送过程涉及到发送邮件服务器和接收邮件服务器之间的交互，这些交互包括 HELO/EHLO、MAIL。

总之，我对这两个应用层协议在 C/S 交互中每一步的详细作用有了更深的理解，且掌握了使用 wireshark 进行简单抓包的技能。