

华东师范大学数据科学与工程学院实验报告

课程名称：计算机网络与编程

年级：2020 级

上机实践成绩：

指导教师：张召

姓名：张熙翔

学号：10205501427

上机实践名称：HTTP、SMTP、POP3 协议分析

上机实践日期：2022/4/18

一、实验目的

熟悉 HTTP 协议的工作原理

了解 HTTP 协议在实际网络中的运行过程

熟悉 SMTP 和 POP3 协议的工作原理

了解 SMTP 和 POP3 协议在实际网络中的运行过程

二、实验任务

通过 Wireshark 分析 HTTP 协议

通过 Wireshark 分析 SMTP 和 POP3 协议

三、使用环境

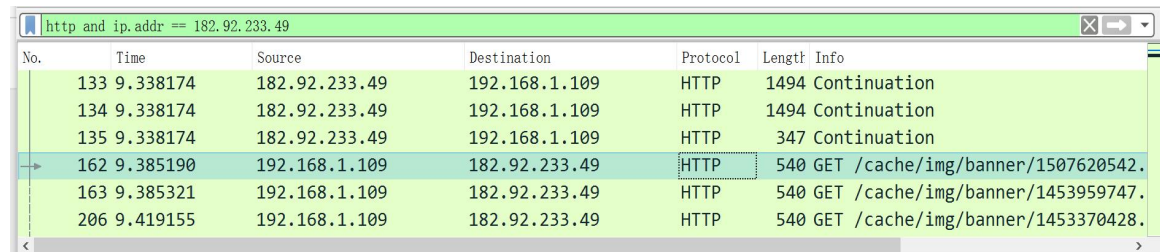
Wireshark

四、实验过程

task1: 利用 Wireshark 抓取一条 HTTP 请求网络包，分析 HTTP 请求网络包的组成（要求根据报文结构正确表示每个部分），请将实验结果附在实验报告中。

访问网址：<http://www.chinesemooc.org/>根据条件 `http and ip.addr == 182.92.233.49` 过滤网络包

选取一条 HTTP 请求网络包



No.	Time	Source	Destination	Protocol	Length	Info
133	9.338174	182.92.233.49	192.168.1.109	HTTP	1494	Continuation
134	9.338174	182.92.233.49	192.168.1.109	HTTP	1494	Continuation
135	9.338174	182.92.233.49	192.168.1.109	HTTP	347	Continuation
162	9.385190	192.168.1.109	182.92.233.49	HTTP	540	GET /cache/img/banner/1507620542.jpg
163	9.385321	192.168.1.109	182.92.233.49	HTTP	540	GET /cache/img/banner/1453959747.jpg
206	9.419155	192.168.1.109	182.92.233.49	HTTP	540	GET /cache/img/banner/1453370428.jpg

Info 为请求行，其中 GET 为请求方法，`/cache/img/banner/1507620542.jpg` 为 URL，HTTP/1.1 为协议版本。

HTTP 请求报文的全部内容为：

```

Hypertext Transfer Protocol
> GET /cache/img/banner/1507620542.jpg HTTP/1.1\r\n
Host: www.chinesemooc.org\r\n
Connection: keep-alive\r\n
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/100.0.4896\r\n
Accept: image/avif,image/webp,image/apng,image/svg+xml,image/*,*/*;q=0.8\r\n
Referer: http://www.chinesemooc.org/\r\n
Accept-Encoding: gzip, deflate\r\n
Accept-Language: zh-CN,zh;q=0.9,en;q=0.8\r\n
Cookie: Hm_lvt_ff4f6e9862a4e0e16fd1f5a7f6f8953b=1648610064,1650273733\r\n
\r\n
[Full request URI: http://www.chinesemooc.org/cache/img/banner/1507620542.jpg]
[HTTP request 2/2]
[Prev request in frame: 114]
[Response in frame: 1209]
    
```

对照 HTTP 请求报文通用格式：



此报文分析如下：

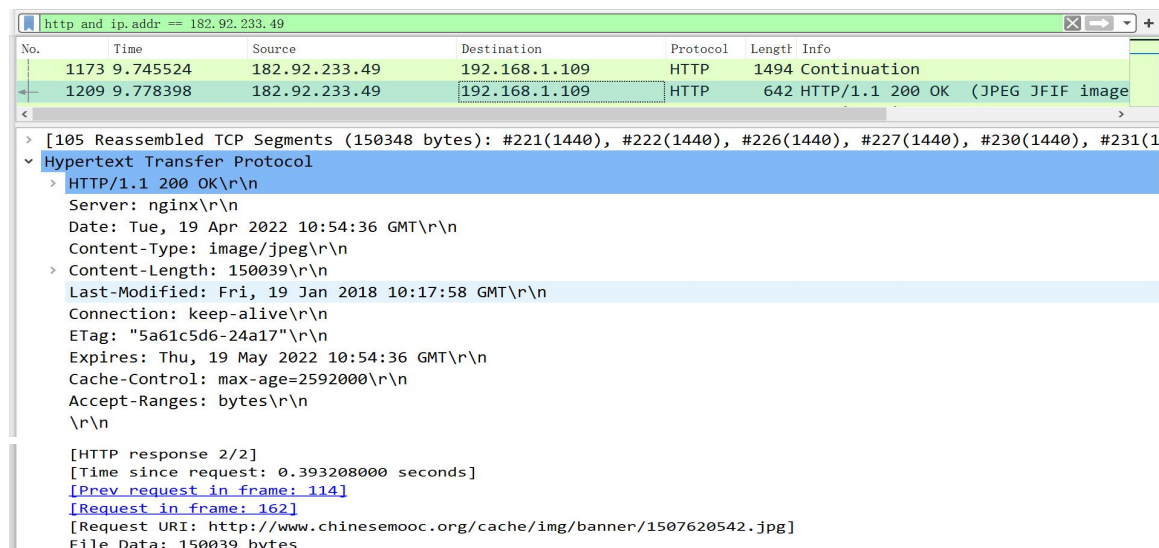
```

Hypertext Transfer Protocol
> GET /cache/img/banner/1507620542.jpg HTTP/1.1\r\n
Host: www.chinesemooc.org\r\n
Connection: keep-alive\r\n
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/100.0.4896\r\n
Accept: image/avif,image/webp,image/apng,image/svg+xml,image/*,*/*;q=0.8\r\n
Referer: http://www.chinesemooc.org/\r\n
Accept-Encoding: gzip, deflate\r\n
Accept-Language: zh-CN,zh;q=0.9,en;q=0.8\r\n
Cookie: Hm_lvt_ff4f6e9862a4e0e16fd1f5a7f6f8953b=1648610064,1650273733\r\n
\r\n
[Full request URI: http://www.chinesemooc.org/cache/img/banner/1507620542.jpg]
[HTTP request 2/2]
[Prev request in frame: 114]
[Response in frame: 1209]
    
```

（注：图中包含手写标注，如“请求行”、“请求头部”、“回车符 换行符”等，用于对应通用格式表）

task2: 利用 Wireshark 抓取上述请求的网络包相对应的 HTTP 响应网络包，然后对比分析两个网络包的组成，请在实验报告中说明两者之间的区别。

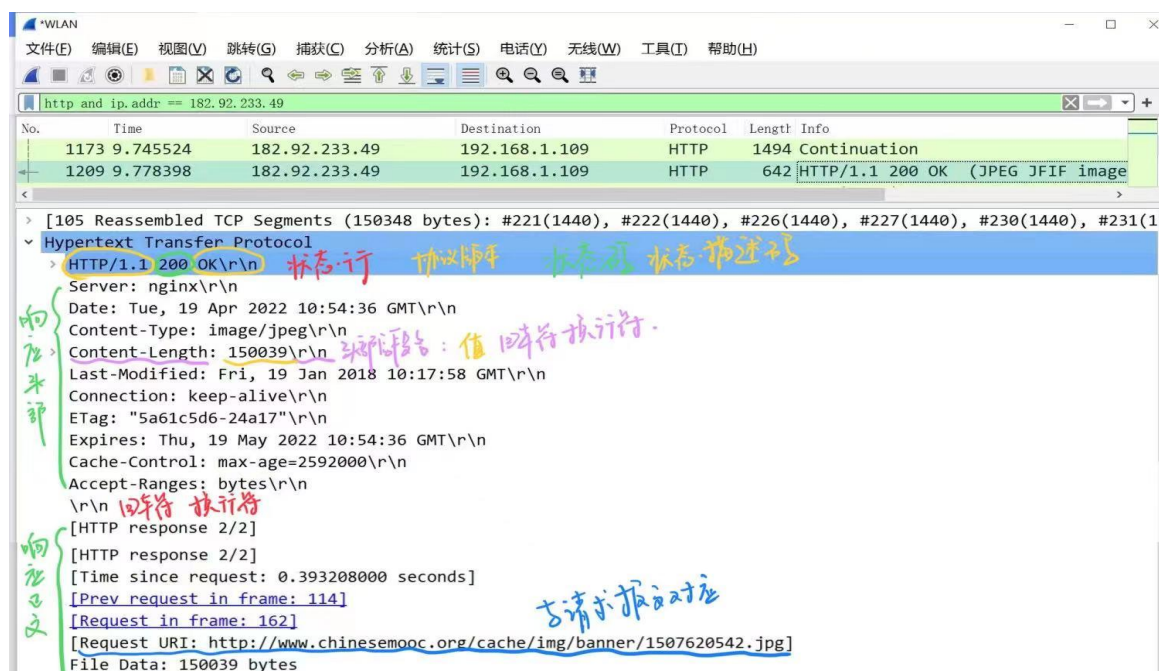
找到 task1 请求的网络包相对应的 HTTP 响应网络包：



对照 HTTP 响应报文通用格式：



此报文分析如下：



与请求报文对比后发现，响应报文首行即状态行格式为：

协议版本	空格	状态码	空格	状态码描述	回车符	换行符
------	----	-----	----	-------	-----	-----

请求报文首行即请求行格式为：

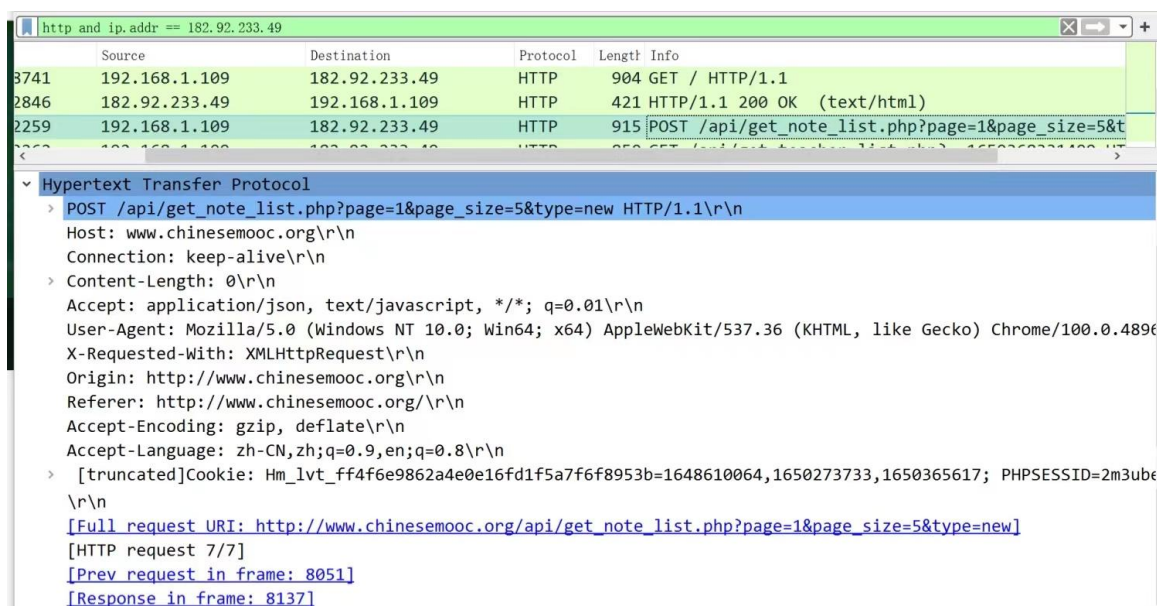
请求方法	空格	URL	空格	协议版本	回车符	换行符
------	----	-----	----	------	-----	-----

头部以及正文部分类似：

头部字段名	:	值	回车符	换行符
...				
头部字段名	:	值	回车符	换行符
回车符	换行符			

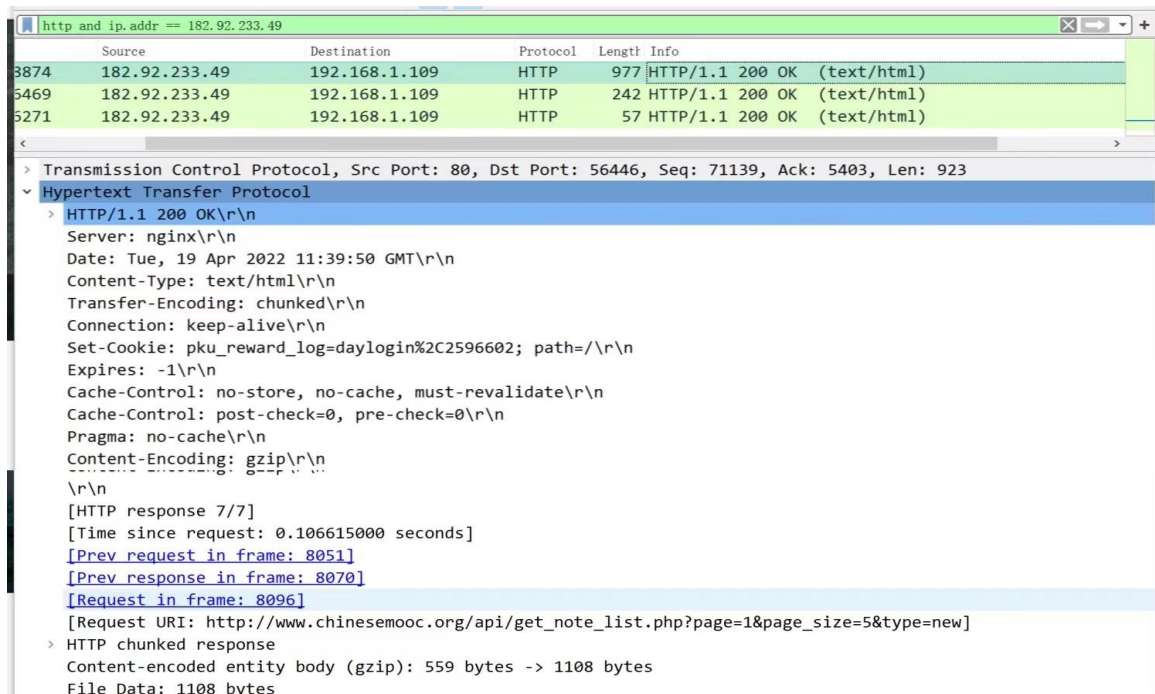
task3: 学习了解 GET 和 POST 方法，请在实验报告中分析对比 GET 和 POST 方法的请求报文，以及 GET 和 POST 方法的响应报文之间的区别。

POST 方法的请求报文：



GET 请求和 POST 请求的请求报文结构上是相同的，但内容是不同的 GET 请求一般没有请求体，数据一般放在 URL 中，而 POST 请求要传递的数据一般放在请求体中。请求体中的数据一般由请求头中的 Content-Type 来决定类型。

POST 方法的响应报文：



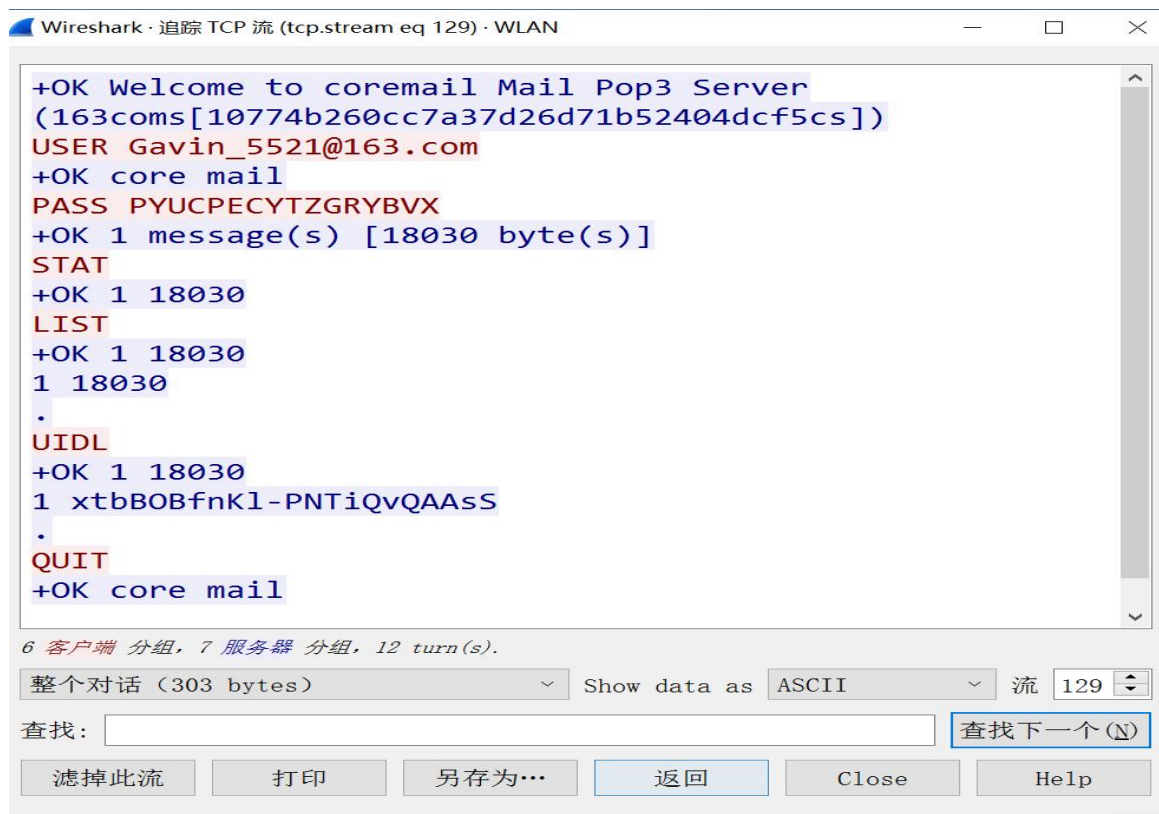
GET 表示从服务器获取资源，而 POST 表示向指定的服务器资源提交数据。get 传送的数据量较小，不能大于 2KB。post 传送的数据量较大，一般被默认为不受限制。

task4: 利用 Wireshark 抓取 SMTP 和 POP3 网络包，分析 SMTP 和 POP3 数据包的组成（要去根据报文结构正确标识每个部分），请将实验结果附在实验报告中。

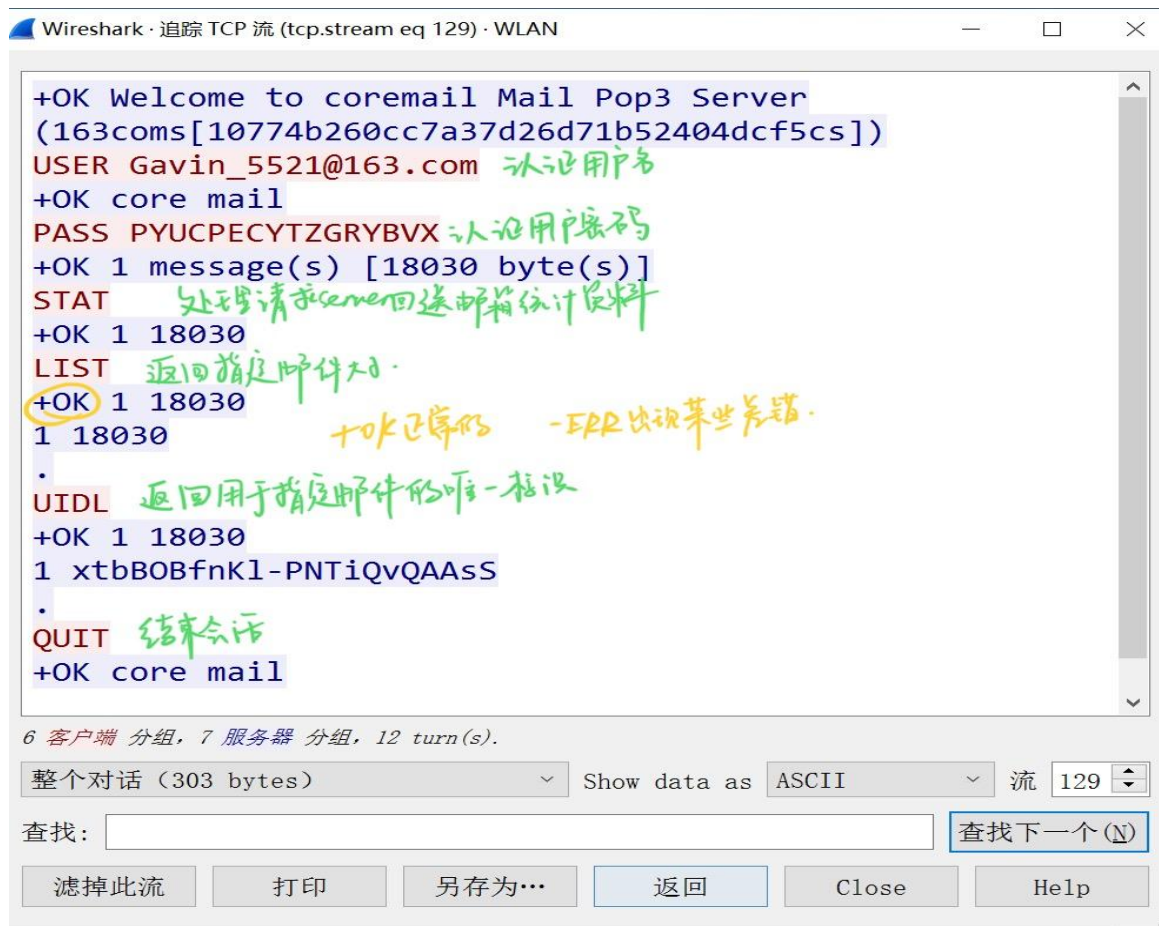
通过追踪流获取交互信息：

No.	Time	Source	Destination	Protocol	Length	Info
1353...	364.280800	192.168.1.109	220.181.15.161	SMTP	86 C:	RCPT TO: <16678985521@163.com>
1353...	364.310596	220.181.15.161	192.168.1.109	SMTP	67 S:	250 Mail OK
1353...	364.311368	192.168.1.109	220.181.15.161	SMTP	60 C:	DATA
1353...	364.339358	220.181.15.161	192.168.1.109	SMTP	91 S:	354 End data with <CR><LF>.<CR><L
1353...	364.339864	192.168.1.109	220.181.15.161	SMTP	418 C:	DATA fragment, 364 bytes
1353...	364.406843	192.168.1.109	220.181.15.161	SMTP/I...	1001 from:	"Gavin_5521@163.com" <Gavin_55
1353...	364.447054	220.181.15.161	192.168.1.109	SMTP	128 S:	250 Mail OK queued as smtp14,EsCo
1353...	364.450584	192.168.1.109	220.181.15.161	SMTP	60 C:	QUIT
1353...	364.481969	220.181.15.161	192.168.1.109	SMTP	63 S:	221 Bye
1353...	367.006997	220.181.12.110	192.168.1.109	POP	141 S:	+OK Welcome to coremail Mail Pop3
1353...	367.007666	192.168.1.109	220.181.12.110	POP	79 C:	USER Gavin_5521@163.com
1354...	367.037500	220.181.12.110	192.168.1.109	POP	69 S:	+OK core mail
1354...	367.037683	192.168.1.109	220.181.12.110	POP	77 C:	PASS PYUCPECYTZGRYBVX
1354...	367.037500	220.181.12.110	192.168.1.109	SCTP		[ACK] Seq=8
1354...	367.037500	220.181.12.110	192.168.1.109	追踪流		mail
1354...	367.037683	192.168.1.109	220.181.12.110	追踪流		DEFVTZGRYBVX

抓取 POP3 网络包：



分析如下：



抓取 SMTP 网络包：

```
220 163.com Anti-spam GT for Coremail System (163com[20141201])
EHLO LAPTOP-KTBHC18V
250-mail
250-PIPELINING
250-AUTH LOGIN PLAIN XOAUTH2
250-AUTH=LOGIN PLAIN XOAUTH2
250-coremail
1Uxr2xKj7kG0xkI17xGrU7I0s8FY2U3Uj8Cz28x1UUUUU7Ic2I0Y2UFilta0UCa0xDrUUUUj
250-STARTTLS
250-ID
250 8BITMIME
AUTH LOGIN
334 dXNlcm5hbWU6
R2F2aW5fNTUyMUAxNjMuY29t
334 UGFzc3dvcmQ6
UF1VQ1BFQ1lUWkdSWUJWWA==
235 Authentication successful
MAIL FROM: <Gavin_5521@163.com>
250 Mail OK
RCPT TO: <16678985521@163.com>
250 Mail OK
DATA
chase=
t=3Dus-ascii"><style>body { line-height: 1.5; }body { font-size: 14px;
fon=
t-family: "Microsoft YaHei UI"; color: rgb(0, 0, 0); line-height: 1.5; }
</=
style></head><body>=0A<div><span></span>test1</div>=0A<div><br></div><hr>
s=
tyle=3D"width: 210px; height: 1px;" color=3D"#b5c4df" size=3D"1"
align=3D"=
left">=0A<div><span><div style=3D"MARGIN: 10px; FONT-FAMILY: verdana;
FONT=
-SIZE: 10pt"><div>Gavin_5521@163.com</div></div></span></div>=0A</
body></h=
tml>
-----=_001_NextPart355645554504_-----
.
250 Mail OK queued as smtp14,EsCowAAX30hXu15iYI3aBw--.37113S2 1650375538
QUIT
221 Bye
```


分析如下：

```

220 163.com Anti-spam GT for Coremail System (163com[20141201])
EHLO LAPTOP-KTBHC18V
250-mail 标识用户身份
250-PIPELINING
250-AUTH LOGIN PLAIN XOAUTH2 250-请求命令完成
250-AUTH=LOGIN PLAIN XOAUTH2
250-coremail
1Uxr2xKj7kG0xkI17xGrU7I0s8FY2U3Uj8Cz28x1UUUUU7Ic2I0Y2UFilta0UCa0xDrUUUUj
250-STARTTLS
250-ID
250 8BITMIME
AUTH LOGIN 认证连接
334 dXNlcm5hbWU6
R2F2aW5fNTUyMUAxNjMuY29t
334 UGFzc3dvcmQ6
UF1VQ1BFQ1lUWkdSWUJWWA==
235 Authentication successful
MAIL FROM: <Gavin_5521@163.com> 发件人地址
250 Mail OK
RCPT TO: <16678985521@163.com> 接收人地址
250 Mail OK
DATA 消息内容
chase=
t=3Dus-ascii"><style>body { line-height: 1.5; }body { font-size: 14px;
fon=
t-family: "Microsoft YaHei UI"; color: rgb(0, 0, 0); line-height: 1.5; }
</=
style></head><body>=0A<div><span></span>test1</div>=0A<div><br></div><hr
s=
tyle=3D"width: 210px; height: 1px;" color=3D"#b5c4df" size=3D"1"
align=3D"=
left">=0A<div><span><div style=3D"MARGIN: 10px; FONT-FAMILY: verdana;
FONT=
-SIZE: 10pt"><div>Gavin_5521@163.com</div></div></span></div>=0A</
body></h=
tml>
-----=_001_NextPart355645554504_-----
.
250 Mail OK queued as smtp14,EsCowAAX30hXu15iYI3aBw--.37113S2 1650375538
QUIT 关闭连接
221 Bye 221-服务器关闭传输通道

```

task5: 利用 Wireshark 抓取 SMTP 网络包，分析一个在 SMTP (C) 和 SMTP 服务器 (S) 之间交换报文文本的例子（参考书本 P77-78），请将实验结果附在实验报告中。


```

220 163.com Anti-spam GT for Coremail System (163com[20141201])
EHLO LAPTOP-KTBHC18V
250-mail
250-PIPELINING
250-AUTH LOGIN PLAIN XOAUTH2
250-AUTH=LOGIN PLAIN XOAUTH2
250-coremail
1Uxr2xKj7kG0xkI17xGrU7I0s8FY2U3Uj8Cz28x1UUUUU7Ic2I0Y2UFilta0UCa0xDrUUUUj
250-STARTTLS
250-ID
250 8BITMIME
AUTH LOGIN
334 dXNlcm5hbWU6
R2F2aw5fNTUyMUAXNjMuY29t
334 UGFzc3dvcmQ6
UF1VQ1BFQ1lUWkdSWUJWWA==
235 Authentication successful
MAIL FROM: <Gavin_5521@163.com>
250 Mail OK
RCPT TO: <16678985521@163.com>
250 Mail OK
DATA
chase=
t=3Dus-ascii"><style>body { line-height: 1.5; }body { font-size: 14px;
fon=
t-family: "Microsoft YaHei UI"; color: rgb(0, 0, 0); line-height: 1.5; }
</=
style></head><body>=0A<div><span></span>test1</div>=0A<div><br></div><hr
s=
tyle=3D"width: 210px; height: 1px;" color=3D"#b5c4df" size=3D"1"
align=3D"=
left">=0A<div><span><div style=3D"MARGIN: 10px; FONT-FAMILY: verdana;
FONT=
-SIZE: 10pt"><div>Gavin_5521@163.com</div></div></span></div>=0A</
body></h=
tml>
-----=_001_NextPart355645554504_-----
.
250 Mail OK queued as smtp14,EsCowAAX30hXu15iYI3aBw--.37113S2 1650375538
QUIT
221 Bye

```

S: 220 163.com Anti-spam GT for Coremail System (163com[20141201])

C: EHLO LAPTOP-KTBHC18V

S: 250 Hello LAPTOP-KTBHC18V, please to meet you

C: AUTH LOGIN

S: 334 dXNlcm5hbWU6

C: MAIL FROM: <Gavin_5521@163.com>

S: 250 Mail OK

C: RCPT TO: <16678985521@163.com>

S: 250 Mail OK

C: DATA

S: 354 End data with <CR><LF>.<CR><LF>

C: .

S: 250 Mail OK queued as smtp14,EsCowAAX3Ohxu15iYI3aBw--.37113S2 1650375538

C: QUIT

S: 221 Bye