

实验十一设计文档

实验任务

在所给的python代码框架的基础上，实现SMTP 邮件客户端，使其能够发送文字和图片。

在本实验中，发送者的邮箱为50627xxxx@qq.com，接受者的邮箱为kairosxxxx@163.com。

实验过程

一个具体的SMTP通信过程如下：

1. 发送端邮件服务器（以下简称客户端）与接收端邮件服务器（以下简称服务器）建立 TCP 连接
 - 发送端登陆的是qq邮箱，所以我们需要知道qq邮箱的邮件服务器是什么，可以在qq邮箱的常见问题中找到如下信息

QQ邮箱的POP3与SMTP服务器是什么？

QQ邮箱 POP3 和 SMTP 服务器地址设置如下：

邮箱	POP3服务器（端口995）	SMTP服务器（端口465或587）
qq.com	pop.qq.com	smtp.qq.com

SMTP服务器需要身份验证。

- 设置邮箱服务器

```
# Choose a mail server (e.g. Google mail server)
# and call it mailserver
mailserver = ('smtp.qq.com', 587)
```

- 建立TCP连接

```
# Create socket called clientSocket
# and establish a TCP connection with mailserver
clientSocket = socket(AF_INET, SOCK_STREAM)
clientSocket.connect(mailserver)

recv = clientSocket.recv(1024).decode()
print(recv)
if recv[:3] != '220':
    print('220 reply not received from server.')
```

2. SMTP 邮件发送程序与 SMTP 邮件接收程序建立连接后必须发送的第一条 SMTP 命令的helo命令，相当于和邮件服务器打招呼。
 - 代码如下

```
# Send HELO command and print server response.
heloCommand = 'HELO Alice\r\n'
clientSocket.send(heloCommand.encode())
recv1 = clientSocket.recv(1024).decode()
print(recv1)
if recv1[:3] != '250':
    print('250 reply not received from server.')
```

3. 选择登录认证方式

- qq邮箱服务器需要登陆认证，所以我们选择login认证方法
- 代码如下

```
login = 'AUTH LOGIN\r\n'
clientSocket.send(login.encode())
recv2 = clientSocket.recv(1024).decode()
print('login: ', recv2)
```

4. 分别输入经过 Base64加密后的用户名和密码

- 用户名即发送者的邮箱50627xxxx@qq.com。密码部分并不是我们正常登陆时候的密码，而是要在邮箱中开启smtp权限，然后系统会给你相应的授权码，以授权码作为密码





```
# Sender and reciever
sender = '50627xxxx@qq.com'
senderpwd = '*****'
```

- 账户和密码需要用base64加密

```
# Auth information (Encode with base64)
# 账户和密码需要经过base64编码加密
user = base64.b64encode(sender.encode()) + b'\r\n'
password = base64.b64encode(senderpwd.encode()) + b'\r\n'
```

- 发送账户和密码

```
clientSocket.send(user)
recv2 = clientSocket.recv(1024).decode()
print('user: ', recv2)
clientSocket.send(password)
recv3 = clientSocket.recv(1024).decode()
print('password: ', recv3)
```

- 5. 用mail from指令和rcpt to指明邮件的发送人和收件人

```
mailFrom = 'MAIL FROM: <50627xxxx@qq.com>\r\n'
clientSocket.send(mailFrom.encode())
recv4 = clientSocket.recv(1024).decode()
print('mail from: ', recv4)

# Send RCPT TO command and print server response.
reptTo = 'RCPT TO: <kairoxxxx@163.com>\r\n'
clientSocket.send(reptTo.encode())
recv5 = clientSocket.recv(1024).decode()
print('rcpt to: ', recv5)
```

6. 输入 data 命令，此命令用于表示 SMTP 邮件发送程序准备开始输入邮件内容，在这个命令后面发送的所有数据都将被当做邮件内容，直至遇到“.”标志符，则表示邮件内容结束

- 输入data命令

```
# Send DATA command and print server response.
data = 'DATA\r\n'
clientSocket.send(data.encode())
recv6 = clientSocket.recv(1024).decode()
print('data: ', recv6)
```

- 编写邮件内容，并发送

From字段填写发送者的邮箱；To字段填写接收者的邮箱；Subject字段填写邮件主题；Content-Type字段填写text/plain表示纯文本；然后填写内容。

```
# Send message data.
msg = 'From: ' + sender + '\r\n'
msg += 'To: ' + receiver + '\r\n'
msg += 'Subject: ' + subject + '\r\n'
msg += 'Content-Type: ' + contenttype + '\r\n'
msg += text_msg
clientSocket.send(msg.encode())
```

- 输入“.”表示邮件内容输入完毕

```
# Message ends with a single period.
clientSocket.send(endmsg.encode())
recv9 = clientSocket.recv(1024).decode()
print('mail: ', recv9)
```

6.1 如果发送的是文本和图片，那么需要修改邮件内容的格式

From字段填写发送者的邮箱；To字段填写接收者的邮箱；Subject字段填写邮件主题。

Content-Type字段填写multipart/mixed，表示接受文本以及附件资源；同时该字段的boundary参数表示之后将以----=_Part_000_0012345DWL作为不同类型内容之间的分割线。除了最后一段，每段的前后应该是--boundary，最后一段的末尾是--boundary--。

在本次实验中，第一段的类型是text/plain，然后空行，然后文本内容，然后再空两行。第二段类型是image/jpeg，然后空行，然后图片内容，然后空行。

```
# Send message data.
msg = 'From: ' + sender + '\r\n'
msg += 'To: ' + receiver + '\r\n'
msg += 'Subject: ' + subject + '\r\n'
msg += 'Content-Type: ' + contenttype + '; boundary="----'
msg += '_Part_000_0012345DWL"\r\n'
msg += 'MIME-Version: 1.0\r\n'
```

```

msg += '\r\n'
msg = msg.encode()
msg += '-----_Part_000_0012345DWL\r\n'.encode()
msg += 'Content-Type: text/plain\r\n'.encode()
msg += '\r\n'.encode()
msg += text_msg.encode()
msg += '\r\n'.encode()
msg += '\r\n'.encode()
msg += '-----_Part_000_0012345DWL\r\n'.encode()
msg += 'Content-Type: image/jpeg; name="networks.jpg"\r\n'.encode()
msg += 'Content-Transfer-Encoding: base64\r\n'.encode()
msg += '\r\n'.encode()
msg += image_msg + '\r\n'.encode()
msg += '\r\n'.encode()
msg += '-----_Part_000_0012345DWL--\r\n'.encode()
clientSocket.send(msg)

```

7. 输入 quit 命令断开与邮件服务器的连接并断开tcp连接

```

# Send QUIT command and get server response.
quitCommand = 'QUIT\r\n'
clientSocket.send(quitCommand.encode())
recv10 = clientSocket.recv(1024).decode()
print('quit: ', recv10)

# Close connection
clientSocket.close()

```

实验结果

1. text.py为只能发送文本的smtp客户端程序

运行结果如下：

```

/Users/DWL/opt/anaconda3/python.app/Contents/MacOS/python /Users/DWL/Desktop/projects/net_lab11/text.py
220 newxmesmtplgicsvrszb5.qq.com XMail Esmtp QQ Mail Server.

250-newxmesmtplgicsvrszb5.qq.com-100.66.14.230-128219582

login: 250-SIZE 73400320
250 OK

user: 334 VXNlcm5hbWU6

password: 334 UGFzc3dvcmQ6

mail from: 235 Authentication successful

rcpt to: 250 OK.

data: 250 OK

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

quit: 250 OK: queued as.

Process finished with exit code 0

```

可以通过相应码来判断是否发送成功以及发送失败的原因是什么。



收件人: 我<

时 间: 2020年2月

附 件: 1个 ( networks.jpg) [查看附件](#)

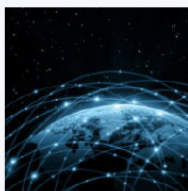
 这个OA系统已打通微信、钉钉。 [免费试用>>](#)

 [翻译成中文](#)

I love computer networks!

[通知: 2020教师证政策改革](#)

 附件 (1)



networks.jpg

28.76K