

Emails in Python

Ruoyu Wang

Maths and Stats Programming Club
University College London

January 14, 2015

Abstract

This note talks about Python in sending emails, and introduces *smtplib* package.

1 Preface

It would be very useful if we could send emails directly via Python, i.e. without any other access to mailbox through web pages or email clients. This note is to indicate a nice and easy way to deal with emails using Python, more precisely, with *smtplib* package.

2 Example code

```
import smtplib #Using package smtplib, a Python library dealing with emails

email_user = raw_input('Email Account: ') #Input your email address
email_pwd = raw_input('Password: ') #Input your email password

smtpserver = smtplib.SMTP("smtp.gmail.com",587) #Gmail default, changable

#Call the server to verify your identity
smtpserver.ehlo()
smtpserver.starttls()
smtpserver.ehlo
smtpserver.login(email_user, email_pwd)

print('Logged in! \n') #Notify if identification success

to = raw_input('To: ') #Input recipient's email address
subject = raw_input('Subject: ') #Input email title
print('\n')

content = ''

#Scrap email content from outside text file
with open('email_content', 'r') as f:
    for line in f:
        content += line

#Make the head to be understandable by the server
header = 'To:' + to + '\n' + 'From:' + email_user + '\n' + 'Subject:' + subject + '\n'
```

```

print header
msg = header + content

print "Content: \n"
print content
print('-----')

#Confirm the email with caution
confirm = raw_input("Y to confirm and send, N to cancel. ")

if (confirm=="y") or (confirm=="Y"):
    smtpserver.sendmail(email_user, to, msg) #Send the email
    print 'Done!'
else:
    print 'Canceled.'
smtpserver.close() #Break link with the server

```

3 How to use the code

Firstly change the outside text file *email content* into what your email content is to be. Run the code, and input your email address and password. Then the computer will authenticate your identity with the server (Default gmail). Then input your recipient address and email title, and review your email and confirm it. After that your email will be sent to the server to process.

4 Trouble shooting

If you are using a normal gmail account to be the sender, you should get to <https://www.google.com/settings/security/lesssecureapps> to enable less secure application to authenticate. The example code is *insecure* and please turn the setting back to disable after trying the example.

5 Variable parameters

There are some variable parameters in this code, namely:

- Email services

```
smtplib.SMTP([host[, port[, local_hostname[, timeout]]]])
```

You can use mail accounts from mail services other than Google, as long as you have the SMTP server address and port number. Simply replace them in the code.

- Carbon Copy (Cc:) and Blind Carbon Copy (Bcc:)
You can add "CC:" and "BCC:" in the *header* string before the Subject tag.

6 Further references

You can find more interesting information in the documentation of *smtplib* package.