
(SMTP Protocol)

Zezafoon rahhal*
Nihal ali*

□ ABSTRACT □

Electronic literature

We present the main features and functionality of the three most widely used message transfer protocols. We study actual implementations and base our analysis on benchmarking. The performance of the protocols is compared in terms of end to end delay, volume of traffic and number of frames generated at the physical layer and frame length distribution. An analytical model is proposed for the approximation of the upper and lower bounds of volume generated by SMTP which can easily be extended for POP3. The model considers both explicit and piggyback acknowledgements at the TCP layer.

Keywords: Simple Mail Transfer Protocol, Post Office Protocol, SMTP,TCP layer, User agent, Message Transfer Agent, Internet Message Access Protocol

(بروتوكول نقل البريد الالكتروني)

المؤلف *زيفون رحال
*نهال علي

□ ملخص □

نقدم الميزات والوظائف الرئيسية لبروتوكولات نقل الرسائل الثلاثة الأكثر استخدامًا. ندرس عمليات التنفيذ الفعلية ونبني تحليلنا على المقارنة المعيارية. تتم مقارنة أداء البروتوكولات من حيث التأخير من طرف إلى طرف وحجم حركة المرور وعدد الإطارات التي تم إنشاؤها في الطبقة المادية وتوزيع طول الإطار. تم اقتراح نموذج تحليلي لتقريب الحدود العليا والسفلى للحجم الذي تم إنشاؤه بواسطة SMTP والذي يمكن تمديده بسهولة لـ POP3. يأخذ النموذج في الاعتبار كلاً من الإقرارات الصريحة والإشعارات على الظهر في طبقة TCP.

الكلمات المفتاحية: بروتوكول نقل البريد البسيط ، بروتوكول مكتب البريد ، SMTP ، طبقة TCP ، وكيل المستخدم ، وكيل نقل الرسائل ، بروتوكول الوصول إلى الرسائل عبر الإنترنت

مقدمة:

SMTP (Simple Mail Transfer Protocol) بروتوكول لإرسال رسائل البريد الإلكتروني البسيط تمتلك بايثون دعماً مدمجاً لبروتوكول SMTP ويمكنها إرسال رسائل بريد إلكتروني نص عادي ورسائل بريد إلكتروني بتنسيق HTML ورسائل تحتوي على مرفقات .

يظهر البريد الإلكتروني كواحد من أكثر الخدمات قيمةً على الإنترنت اليوم. تستخدم معظم أنظمة الإنترنت الـ SMTP كطريقة لنقل البريد من مستخدم إلى آخر. حيث أن SMTP هو بروتوكول دفع "Push" ويُستخدم لإرسال البريد ، بينما يتم استخدام POP (Post Office Protocol) بروتوكول مكتب البريد أو IMAP (Internet Message Access Protocol) بروتوكول الوصول إلى رسائل الإنترنت لاسترداد تلك الرسائل من جانب المُتلقي.

أهمية البحث وأهدافه:

تمتلك بايثون دعماً لبروتوكول SMTP ويمكنها إرسال رسائل بريد إلكتروني ، سنتعرف على كيفية إرسال بريد إلكتروني، مع إضافة مرفقات للبريد الإلكتروني.

أساسيات SMTP:

ينتمي بروتوكول SMTP إلى Application Layer يفتح "Client" الذي يريد إرسال البريد اتصال TCP بخادم SMTP ثم يرسل البريد من خلاله. خادم SMTP في وضع الاستماع دائماً. بمجرد الاستماع إلى اتصال TCP من أي client، يبدأ بإنشاء اتصال على (25) port. وبعد إنشاء اتصال TCP بنجاح، يستطيع client القيام بإرسال البريد على الفور.

نموذج نظام SMTP:

يتعامل المستخدم في نموذج SMTP مع وكيل المستخدم (UA) User Agent على سبيل المثال Microsoft Outlook و Netscape و Mozilla وما إلى ذلك. من أجل تبادل البريد باستخدام TCP، يتم استخدام "MTA Message Transfer Agent". لا يتعين على المستخدمين الذين يرسلون البريد التعامل مع MTA، بل هي من مهام System Admin أن يقوم بإعداد MTA محلي.

يحتفظ MTA بقائمة انتظار صغيرة من رسائل البريد بحيث يمكنه جدولة تسليم البريد المتكرر في حالة عدم توفر جهاز الاستقبال. يقوم MTA بتسليم البريد إلى صناديق البريد ويمكن لوكلاء المستخدم تنزيل المعلومات لاحقاً.

يجب أن يحتوي كل من عميل SMTP وخادم SMTP على مكونين:

• وكيل المستخدم (UA)

• و MTA محلي.

التواصل بين المرسل والمتلقي: يقوم المرسل ووكيل المستخدم بإعداد الرسالة وإرسالها إلى MTA ،يعمل MTA على نقل البريد عبر الشبكة إلى MTA في أجهزة الاستقبال لإرسال البريد، يجب أن يكون لدى النظام العميل MTA، ولاستلام البريد يجب أن يحتوي النظام أيضا على خادم MTA .

إرسال البريد الإلكتروني: يتم إرسال البريد عن طريق سلسلة من رسائل الطلب والرد بين العميل والخادم.

استلام البريد الإلكتروني: يتحقق وكيل المستخدم من جانب الخادم من صندوق البريد خلال فترات زمنية محددة. وفي حال تلقي أي معلومات فإنها تُخبر المستخدم عن الرسالة. عندما يحاول المستخدم قراءة البريد، فإنه يعرض قائمة بالبريد مع وصف موجز لكل رسالة في صندوق البريد.

تطبيقات بايثون عديدة ومتشعبة في أكثر من مجال، وذلك لما تتمتع به لغة البايثون من مزايا وخصائص تجعلها حلاً للعديد من المشاكل والعقبات. سنتناول في هذا البحث أحد تطبيقات بايثون وهو كيفية التعامل مع مكتبة `smtplib` ومكتبة `email` المعياريتان في البايثون والتي تُساعدنا في إرسال بريد إلكتروني بخيارات متعددة.

مكتبة `smtplib`:

تقدم البايثون العديد من الخيارات في مكتباتها للتعامل مع البريد الإلكتروني. أشهر هذه المكتبات هي مكتبة `smtplib` المُضمنة تلقائياً في البايثون. تعمل المكتبة على إنشاء `SMTP client session` نتمكن من خلالها إرسال بريد إلكتروني لأي عنوان بريد إلكتروني في الإنترنت. يتم ذلك عبر استخدام البروتوكول الشهير `SMTP protocol` . عند استخدام المكتبة، يتم إنشاء كائن من نوع الفئة `SMTP` يُمكن استخدامه فيما بعد في إرسال البريد الإلكتروني.

المعطيات التي يُمكن تمريرها للكائن المنشأ من فئة `SMTP` هي:

* `host` : هو عنوان IP الخاص بالخادم الذي يُشغل خدمة `SMTP`

* `port` : وهو رقم المنفذ المستخدم في خادم `SMTP` والذي يتتصت من خلاله على طلبات الإرسال القادمة إليه. في حال أعدنا معطى `host` يجب علينا هنا تحديد رقم المنفذ والذي يكون بالعادة ٢٥.

* `host_localname`: في حالة استضافة خادم `SMTP` على جهازك المحلي، يُمكنك تمرير `localhost` لهذا

المُعطى.

يحتوي الكائن المنشأ من فئة `SMTP` على وظيفة `sendmail` وتستقبل ثلاثة معطيات رئيسية هي:

1-البريد المرسل.

2-البريد المرسل إليه.

3-نص الرسالة.

بروتوكول مكتب البريد POP :

بروتوكول (POP) Post Office Protocol هو بروتوكول معياري لطبقة التطبيقات يستخدمه عملاء البريد الإلكتروني لاسترداد البريد الإلكتروني من خادم البريد .

يوفر بروتوكول Post Office Protocol الوصول عبر شبكة بروتوكول الإنترنت (IP) لتطبيق عميل مستخدم إلى صندوق بريد (Maildrop) يتم الاحتفاظ به على خادم بريد. يدعم البروتوكول عمليات التنزيل والحذف للرسائل. يقوم عملاء POP3 بالاتصال ، واسترداد جميع الرسائل ، وتخزينها على جهاز الكمبيوتر العميل ، وأخيراً حذفها من الخادم . كان الدافع وراء تصميم بروتوكول POP هذا وإجراءاته هو حاجة المستخدمين الذين لديهم اتصالات إنترنت مؤقتة فقط ، مثل الوصول عبر الطلب الهاتفي ، مما يسمح لهؤلاء المستخدمين باسترداد البريد الإلكتروني عند الاتصال ، وبالتالي عرض الرسائل المسترجعة ومعالجتها عندما تكون غير متصل بالإنترنت.

عملاء POP3 لديهم أيضًا خيار ترك البريد على الخادم بعد التنزيل. على النقيض من ذلك ، تم تصميم بروتوكول الوصول إلى الرسائل عبر الإنترنت (IMAP) بحيث يترك بشكل طبيعي جميع الرسائل على الخادم للسماح بالإدارة باستخدام تطبيقات متعددة للعميل.

IMAP

و بروتوكول الوصول إلى الرسائل عبر إنترنت (IMAP) هو بروتوكول الوصول إلى علبة البريد البديل وأكثر حداثة. أبرز الاختلافات هي:

- بروتوكول POP هو بروتوكول أبسط ، مما يجعل التنفيذ أسهل.
- يقوم بروتوكول POP بنقل الرسالة من خادم البريد الإلكتروني إلى الكمبيوتر المحلي ، على الرغم من وجود خيار لترك الرسائل على خادم البريد الإلكتروني أيضًا.
- يتم تعيين IMAP افتراضيًا على ترك الرسالة على خادم البريد الإلكتروني ، ما عليك سوى تنزيل نسخة محلية.
- يتعامل بروتوكول POP مع صندوق البريد كمخزن واحد ، وليس لديه مفهوم للمجلدات
- ينفذ عميل IMAP استعلامات معقدة ، ويسأل الخادم عن الرؤوس ، أو نصوص الرسائل المحددة ، أو للبحث عن الرسائل التي تلبي معايير معينة. يمكن تمييز الرسائل الموجودة في مستودع البريد بعلامات حالة مختلفة (على سبيل المثال "محذوفة" أو "تم الرد عليها") وتبقى في المستودع حتى يتم إزالتها بشكل صريح من قبل المستخدم - والتي قد لا يتم ذلك حتى جلسة لاحقة. باختصار: تم تصميم IMAP للسماح بمعالجة صناديق البريد البعيدة كما لو كانت محلية. اعتمادًا على تنفيذ عميل IMAP وبنية البريد التي يريدها مدير النظام ، يجوز للمستخدم حفظ الرسائل مباشرة على جهاز العميل ، أو حفظها على الخادم ، أو منح خيار القيام بأي منهما.
- يتطلب بروتوكول POP أن يكون العميل المتصل حاليًا هو العميل الوحيد المتصل بصندوق البريد. في المقابل ، يسمح بروتوكول IMAP تحديدًا بالوصول المتزامن لعدة عملاء ويوفر آليات للعملاء لاكتشاف التغييرات التي تم إجراؤها على صندوق البريد بواسطة عملاء آخرين متصلين بشكل متزامن. راجع على سبيل المثال RFC3501 القسم ٥,٢ الذي يشير تحديدًا إلى "الوصول المتزامن إلى نفس صندوق البريد بواسطة وكلاء متعددين" كمثال.
- عندما يسترد بروتوكول POP رسالة ، فإنه يتلقى جميع أجزاءها ، بينما يسمح بروتوكول IMAP4 للعملاء باسترداد أي جزء من أجزاء MIME الفردية بشكل منفصل - على سبيل المثال ، استرداد النص العادي دون استرداد الملفات المرفقة.
- يدعم IMAP العلامات الموجودة على الخادم لتتبع حالة الرسالة: على سبيل المثال ، ما إذا كانت الرسالة قد تمت قراءتها أو الرد عليها أو إعادة توجيهها أو حذفها أم لا.

Simple SMTP server made in python

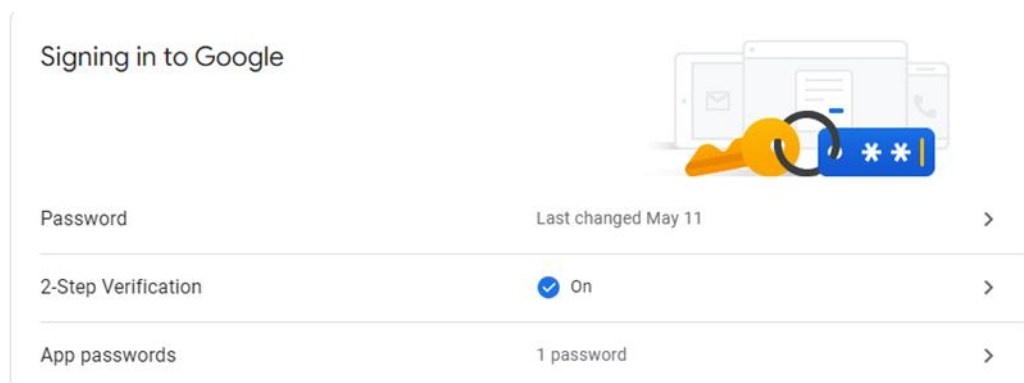
this has both the CLI and GUI version of the simple SMTP:

عندما نريد أن نرسل بريد إلكتروني باستخدام البايثون، علينا في البداية أن نتأكد من إعدادات الأمان الخاصة باتصال smtp .

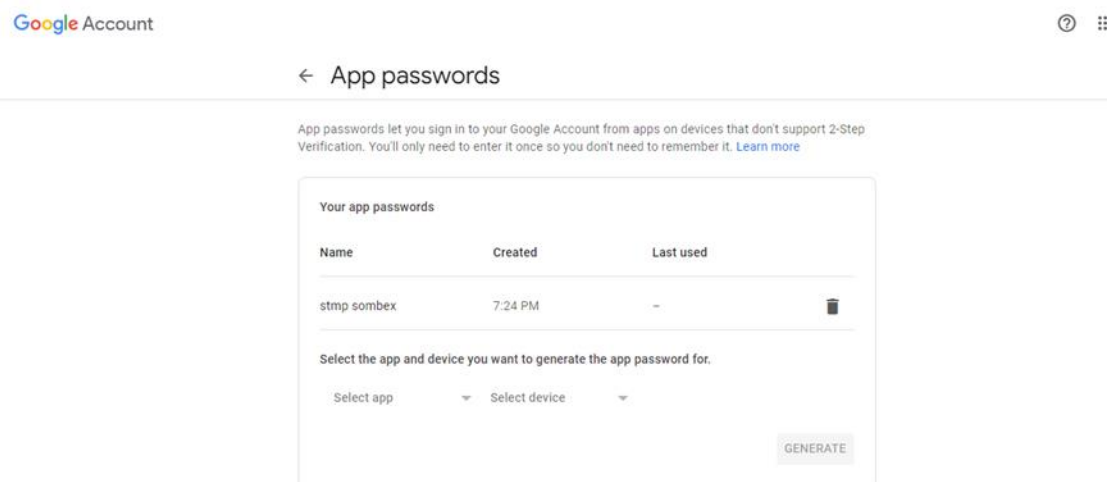
تجهيز إعدادات الأمان للاتصال ستحافظ على سرية البيانات التي يتم انتقالها في الإتصال. سنستخدم لإضافة إعدادات الأمان وحدة Tls التي تُوفر آليات تجهيز الاتصال المُشفّر. في البداية نستورد smtp بعد ذلك نُعرف المنفذ الذي سيتم الاتصال به في خادم smtp (منفذ smtp.gmail.com هو 587 للاتصال عبر Tls)

setup_free_Google_SMTP_Server_:

الخطوة الأولى:



الخطوة الثانية:



الخطوة الثالثة:

SMTP Server: smtp.gmail.com

Port: 587

Security: Tls

Username: fazalurrahman2005@gmail.com

Password:

From email address: fazalurrahman2005@gmail.com

To email address: fazalurrahman2005@gmail.com

Test it

وبذلك نكون أنشأنا حاسبنا كسيرفر SMTP.
كيف نرسل ملف باستخدام SMTP ؟

الكود الأول:

```
#!/usr/bin/python
```

```
# Build input code for customer's email.
```

```
# Build attachment for pdf filetype
```

```
# Build mail server
```

```
import smtplib
```

```
from email.mime.multipart import MIMEMultipart
```

```
from email.mime.text import MIMEText
```

```
from email.mime.base import MIMEBase
```

```
from email import encoders
```

```
body = 'Hello from network programming. Have a great day ;)'
```

```
sender = 'fazalurrahman2005@gmail.com'
```

```
# Get the password in the gmail by: manage your google account >> click on the avatar on the right
```

```
# then go to security (right) >> app password (center)
```

```
# MUST HAVE 2-FA ENABLED FOR THIS TO WORK!!
```

```
# insert the password and then choose mail and this computer and then generate
```

```
# copy the password
```

```
password = 'xipkeefddpqqxul'
```

```
# put the email of the receiver here
receiver = input("What is the Customer's Email?: ")
# subject here
subject = 'Invoice'
# Setup the MIME
message = MIMEMultipart()
message['From'] = sender
message['To'] = receiver
message['Subject'] = subject

message.attach(MIMEText(body, 'plain'))

pdf_name = input("Drop PDF here: ")

# Open PDF to Binary
binary_pdf = open(pdf_name, 'rb')

payload = MIMEBase('application', 'octet-stream', Name=pdf_name)
payload.set_payload(binary_pdf.read())

# binary to base64
encoders.encode_base64(payload)

# add header with pdf name
payload.add_header('Content-Description', 'attachment', filename=pdf_name)
message.attach(payload)

# SMTP setup
session = smtplib.SMTP('smtp.gmail.com', 587)

# Security
session.starttls()

# Login
session.login(sender, password)

text = message.as_string()
session.sendmail(sender, receiver, text)
session.quit()
```

```
print('Mail Sent')
```

الكود الثاني:

```
from tkinter import *
from tkinter import filedialog
import smtplib
from email.message import EmailMessage

#Global variables
attachments = []

#Main Screen Init
master = Tk()
master.title = 'Easyway Email App'

#Functions
def attachFile():
    filename = filedialog.askopenfilename(initialdir='c:/',title='Please select a file')
    attachments.append(filename)
    notif.config(fg='green', text = 'Attached ' + str(len(attachments)) + ' files')

def send():
    try:
        msg = EmailMessage()
        username = 'fazalurrahman2005@gmail.com'
        password = 'alzxibgphgjwjeuz'
        to = temp_receiver.get()
        subject = 'Invoice'
        body = 'Hello from network programming. Have a great day ;)'
        msg['subject'] = subject
        msg['from'] = username
        msg['to'] = to
        msg.set_content(body)

        filename = attachments[0]
        filetype = filename.split('.')
        filetype = filetype[1]
        if filetype == "jpg" or filetype == "JPG" or filetype == "png" or filetype == "PNG":
```

```

import imghdr
with open(filename, 'rb') as f:
    file_data = f.read()
    image_type = imghdr.what(filename)
    msg.add_attachment(file_data, maintype='image', subtype=image_type, filename=f.name)

else:
    with open(filename, 'rb') as f:
        file_data = f.read()
        msg.add_attachment(file_data, maintype='application', subtype='octet-stream', filename=f.name)

if username==" or password==" or to==" or subject==" or body==" :
    notif.config(text="All fields required", fg="red")
    return
else:
    server = smtplib.SMTP('smtp.gmail.com',587)
    server.starttls()
    server.login(username, password)
    server.send_message(msg)
    notif.config(text="Email has been sent successfully", fg="green")
except:
    notif.config(text="Error sending email", fg="red")

def reset():
    usernameEntry.delete(0,'end')
    passwordEntry.delete(0,'end')
    receiverEntry.delete(0,'end')
    subjectEntry.delete(0,'end')
    bodyEntry.delete(0,'end')

#Labels
Label(master, text="Easyway Email App", font=('Calibri',15)).grid(row=0, sticky=N)
Label(master, text="Please use the form below to send an email", font=('Calibri',11)).grid(row=1, sticky=W,
padx=5 ,pady=10)

Label(master, text="mail id", font=('Calibri', 11)).grid(row=4,sticky=W, padx=5)

notif = Label(master, text="", font=('Calibri', 11),fg="red")

```

```
notif.grid(row=7,sticky=S)
```

```
#Storage
```

```
temp_username = 'fazalurrahman2005@gmail.com'
```

```
temp_password = 'alzxibgphgjweuz'
```

```
temp_receiver = StringVar()
```

```
temp_subject = 'Invoice'
```

```
temp_body = 'Hello from network programming. Have a great day ;)'
```

```
#Entries
```

```
receiverEntry = Entry(master, textvariable = temp_receiver)
```

```
receiverEntry.grid(row=4,column=0)
```

```
#Buttons
```

```
Button(master, text = "Send", command = send).grid(row=7, sticky=W, pady=15, padx=5)
```

```
Button(master, text = "Reset", command = reset).grid(row=7, sticky=W, padx=45, pady=40)
```

```
Button(master, text = "Attachments", command = attachFile).grid(row=7, sticky=W, padx=90, pady=40)
```

```
#Mainloop
```

```
master.mainloop()
```

النتائج

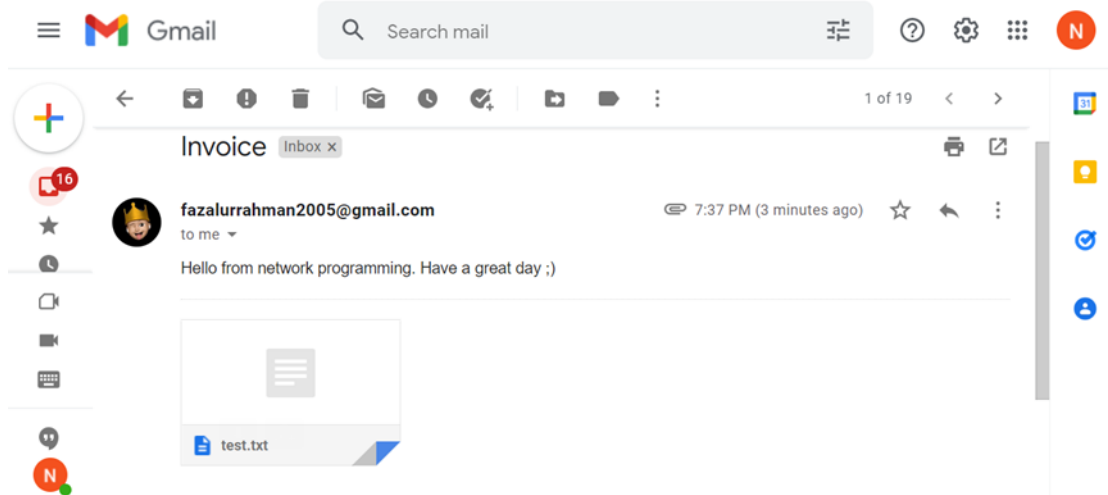
ناتج الكود الأول :

```
IDLE Shell 3.10.4
File Edit Shell Debug Options Window Help
Python 3.10.4 (tags/v3.10.4:9d38120, Mar 23 2022, 23:13:41) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/user/Desktop/last/EmailClient.py =====
What is the Customer's Email?: nihaloali10189@gmail.com
Drop PDF here: test.txt
Mail Sent
>>> |
```

ناتج الكود الثاني:

The screenshot shows a Tkinter window titled "tk" with a light gray background. The window contains the following elements:

- Title:** Easyway Email App
- Instruction:** Please use the form below to send an email
- Form:** A label "mail id" followed by a text input field containing "loali10189@gmail.com".
- Buttons:** Three buttons labeled "Send", "Reset", and "Attachments" are arranged horizontally.
- Message:** A green text message at the bottom states "Email has been sent successfully".



الاستنتاجات :

بذلك نكون شرحنا أحد تطبيقات بايثون وهو كيفية استخدام البايثون في ارسال بريد إلكتروني مع كافة الخيارات المتاحة لارفاق ملف مع البريد

References

- [1] Gordon, M. (2005). An introduction to network programming the python way [book review]. *IEEE Distributed Systems Online*, 6(10)..
- [2] Gordon, M. (2005). An introduction to network programming the python way [book review]. *IEEE Distributed Systems Online*, 6(10).
- [3] Tzerefos, P., Smythe, C., Stergiou, I., & Cvetkovic, S. (1997, November). A comparative study of simple mail transfer protocol (SMTP), post office protocol (POP) and X. 400 electronic mail protocols. In *Proceedings of 22nd Annual Conference on Local Computer Networks* (pp. 545-554). IEEE