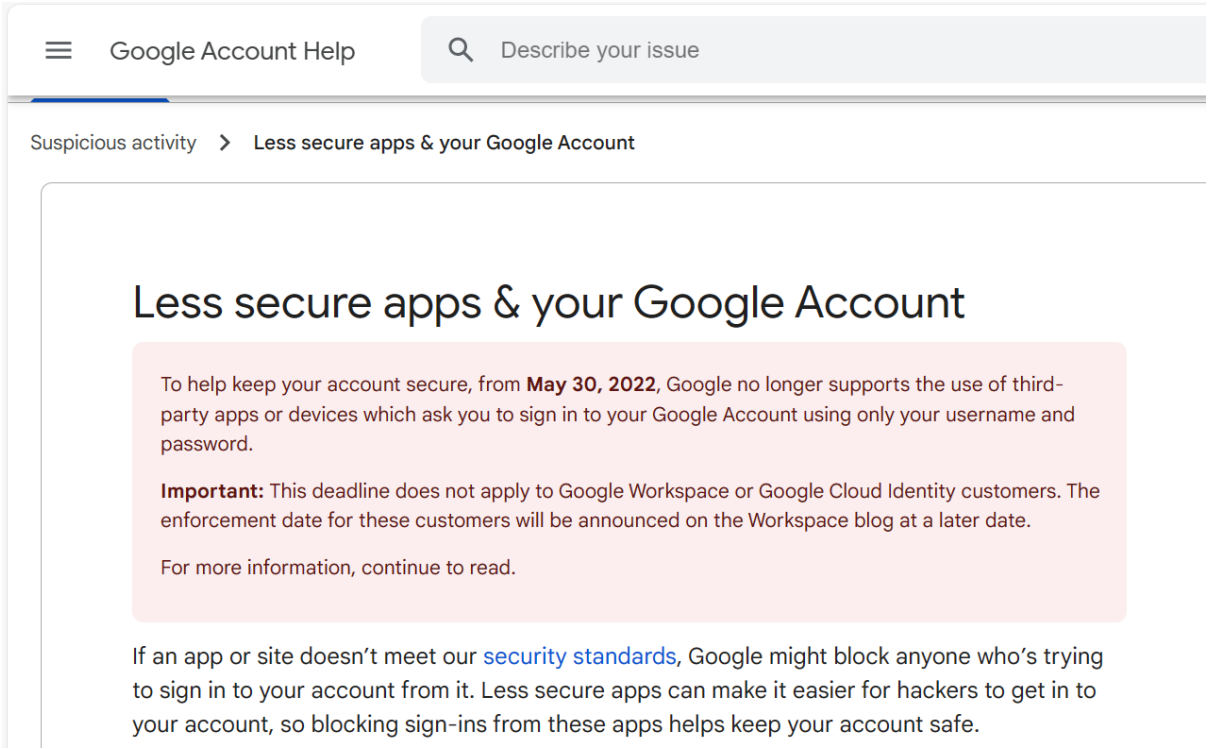
SAIC- Sysadmin test 2023

Challenge 5

For this challenge, choice of language is python. After googling, I learned about smtplib module to send mails via python (reference- realpython.com). Unfortunately, I was not able to use my Gmail account to send mails due to its updated security policies.



Therefore, switched to outlook account for sending mails.

Code-

saic_challenge5.py - D:\Python\saic_challenge5.py (3.11.4)

File Edit Format Run Options Window Help

```
#challenge 5
import getpass, smtplib
host="smtp-mail.outlook.com"
port=587
from_email="[redacted]outlook.com"
to_email="[redacted]gmail.com"  #to be taken via web scraping
pwd=getpass.getpass('Enter your password here :')

msg="""Subject: Reminder regarding unsubmitted assignment

Dear student

This is a reminder about your upcoming assignment which is due 6 hours from now.
Kindly submit the assignment on time

Regards
Well wisher
"""
smtp=smtplib.SMTP(host,port)

try:
    status_code,response=smtp.ehlo()
    print(f"[*] Echoing the server: {status_code},{response}")
    status_code,response=smtp.starttls()
    print(f"[*] Starting TLS connection: {status_code} {response}")

    status_code, repsonse = smtp.login(from_email,pwd)
    print(f"[*] Starting Logging in: {status_code} {response}")

    smtp.sendmail(from_email,to_email,msg)
    print('Mail sent successfully')

except smtplib.SMTPAuthenticationError as e:
     print(f"Error: SMTP Authentication failed. {e}")

smtp.quit()
```

Explanation

Importing getpass module to get outlook password and smtplib module to connect SMTP server and send emails using SMTP (Simple Mail Transfer Protocol)

ehlo method pings the server and checks if the server is running or not. It returns a tuple showing the status code of the server and the response

smtp.starttls() establishes TLS (Transport Layer Security) connection between system and server

TLS is a set of rules governing secure communications over the network (cryptographic protocol)

smtp.login(from_email,pwd) is used to authenticate and log in to SMTP server. Credentials are given by the user

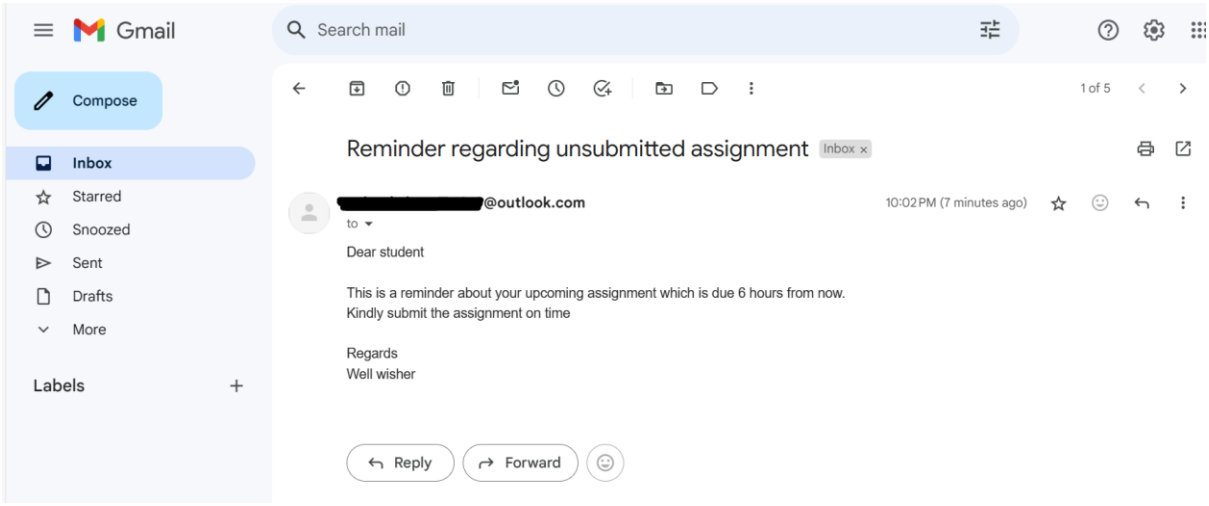
smtp.sendmail() is used to send the message

smtp.quit() is used to terminate the SMTP session

try()...except() is used to catch any errors and check if the connection is successfully established

```
========== RESTART: D:\Python\saic_challenge5.py ==========

Warning (from warnings module):
  File "D:\Python\Lib\getpass.py", line 100
    return fallback_getpass(prompt, stream)
GetPassWarning: Can not control echo on the terminal.
Warning: Password input may be echoed.
Enter your password here :[redacted]
[*] Echoing the server: 250,b'PN0PR01CA0032.outlook.office365.com Hello [2401:4900:1c5a:7222:14fa:a193:2ed7:8513]\nSIZE 157286400\nPIPELINING\nDSN\nENHANCEDSTATUSCODES\nSTARTTLS\n8BITMIME\nBINARYMIME\nCHUNKING\nSMTPUTF8'
[*] Starting TLS connection: 220 b'2.0.0 SMTP server ready'
[*] Starting Logging in: 235 b'2.0.0 SMTP server ready'
Mail sent successfully
>>>
```



This code caters to one aspect of the problem, that is sending automated mails. For specifying the date and time of sending the mail, we need data from the concerned website (lms.iitmandi.ac.in in this case)

After googling, I figured out two ways of doing so.

1) Using API – This is a more stable and sanctioned way to access data as compared to web scraping mentioned below
2) Web scraping – In python, web scraping is done by using modules- requests (Fetching HTML content) and BeautifulSoup (parsing the HTML and extracting data). However as per my understanding, we need prior permission from the website owners before scraping data from the concerned site

The idea was to extract data and time of unsubmitted assignments as well as email ids of students enrolled in the course. Data would be extracted from iitmandi lms by web scraping. Then, finally sending mails using this data.

However, unfortunately I was not able to figure out whether web scraping is allowed from the desired website