

INTRODUCTION TO COMPUTER SECURITY

PHASE-3

INFECTION

Team members:

Pooja Peechara :11663837

Rachana Reddy Sunki :11709719

Mounika Nuchu :11653658

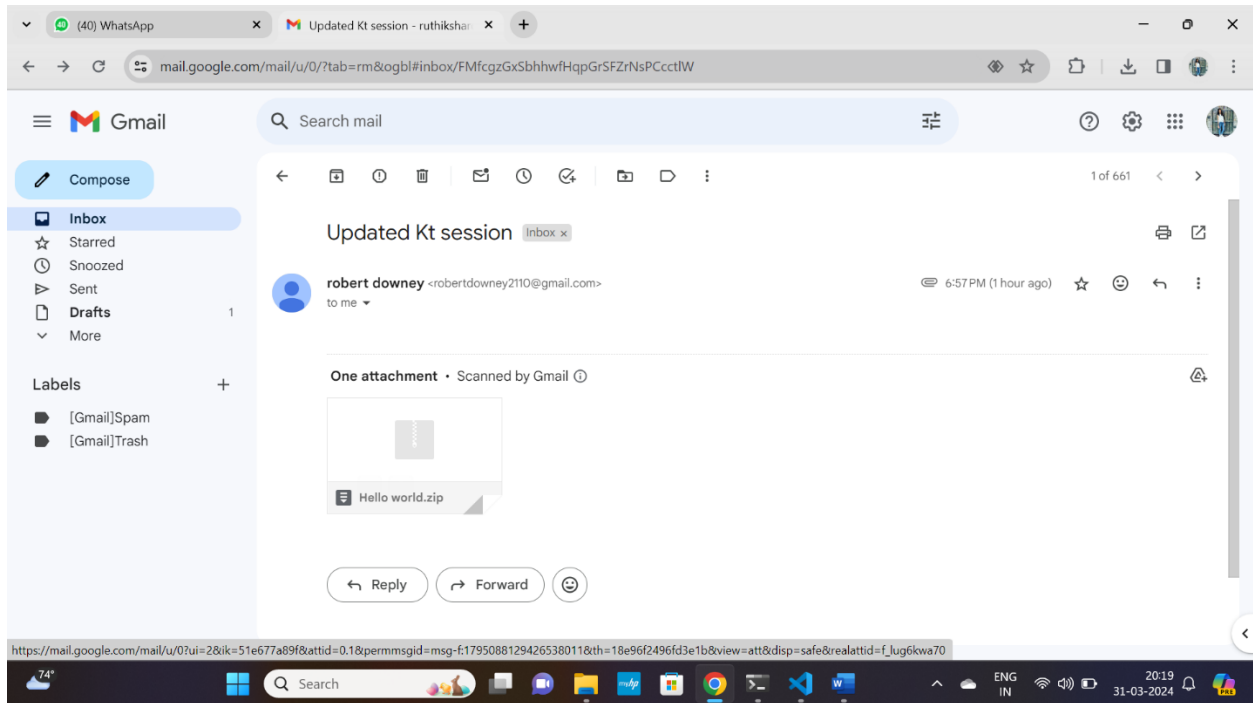
Lagadapati Kondarao :11661743

Yenugu Ruthiksha Reddy :11714976

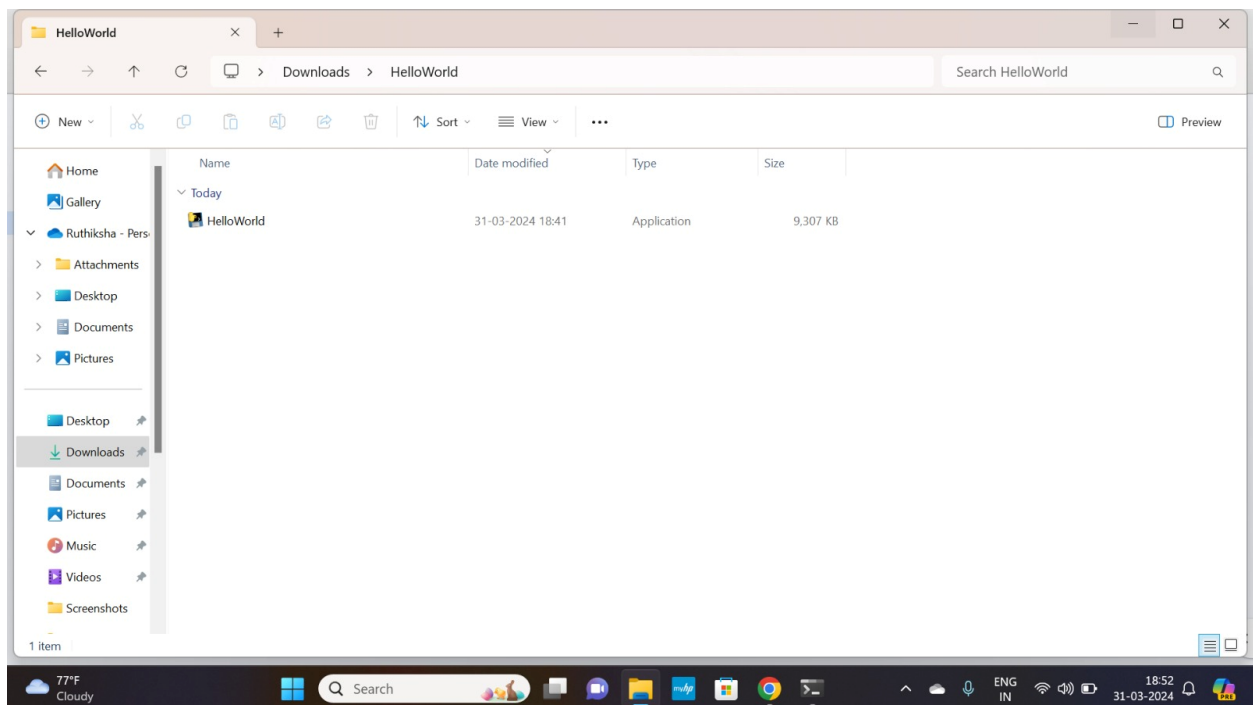
Introduction: In this phase we will be sending a Phishing email to our target person. In the previous phase we have seen the encryption and decryption. In addition to that we will be writing a script which the target user will download on their device. The .exe file will be run once the user clicks on it. We have given the path of the folder where the code runs and encrypts the entire folder.

Here, We have demonstrated the email phishing by giving our Gmail as the target email.

The below is the email from Robert Downey alias me. The Hello world.zip consists of the .exe file.

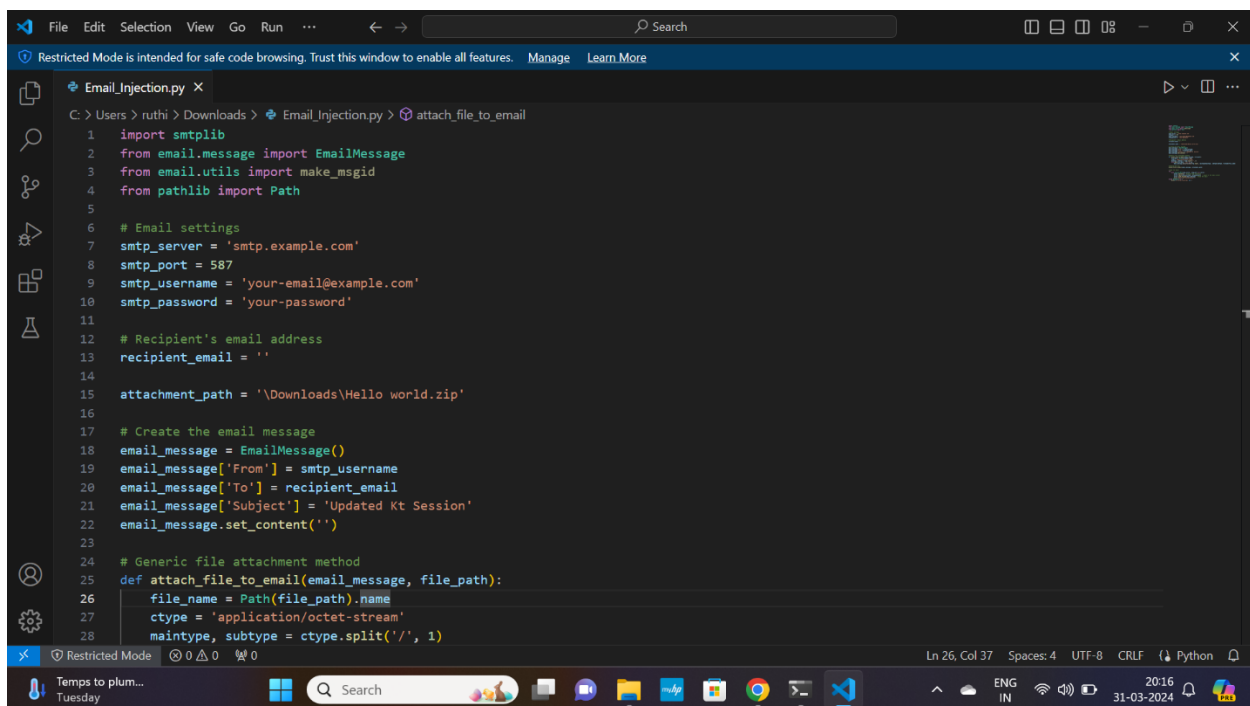


The below is the Helloworld application which will encrypt the entire folder as the path has been given in our code.



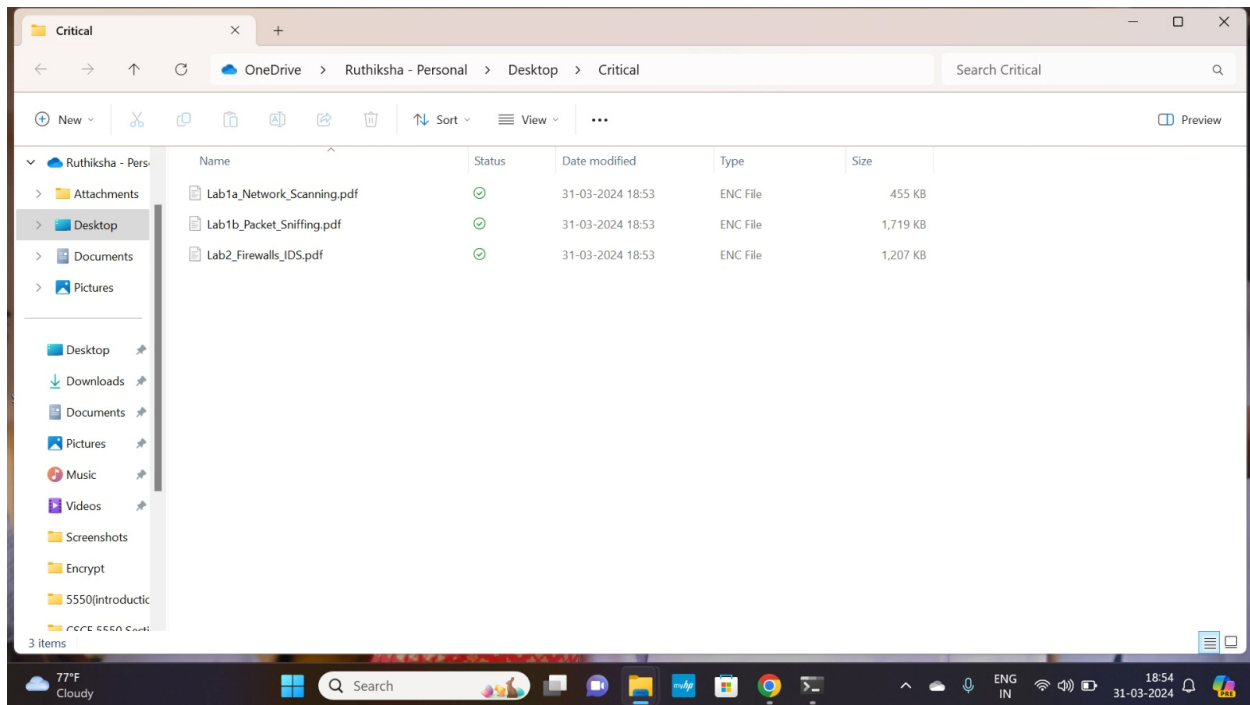
The below code is to send the Phishing email to our target. I have used smtplib library which is used to send emails programmatically. SMTP is called as simple mail transfer protocol. We can even send attachments by using this library by just giving the path of the folder and the attachment will be attached with the email.

You can see that the message in the code matches the above mail screenshot received from Robert Downey.

A screenshot of a code editor window showing a Python script named 'Email_Injection.py'. The script is designed to send an email with an attachment using the smtplib library. It includes settings for the SMTP server, port, username, and password. The recipient's email address is set to an empty string. The attachment path is set to 'Downloads\Hello world.zip'. The email message is created with a 'From' field, 'To' field, 'Subject' field, and 'Content' field. A generic file attachment method is defined, which takes an email message and a file path as input. The method extracts the file name from the path, sets the content type to 'application/octet-stream', and sets the main type and subtype to 'application/octet-stream'. The script is running in a restricted mode, as indicated by the 'Restricted Mode' label in the bottom left corner of the editor window. The status bar at the bottom shows the current line and column (Ln 26, Col 37), the number of spaces (4), the encoding (UTF-8), the line ending (CRLF), and the language (Python).

```
1 import smtplib
2 from email.message import EmailMessage
3 from email.utils import make_msgid
4 from pathlib import Path
5
6 # Email settings
7 smtp_server = 'smtp.example.com'
8 smtp_port = 587
9 smtp_username = 'your-email@example.com'
10 smtp_password = 'your-password'
11
12 # Recipient's email address
13 recipient_email = ''
14
15 attachment_path = 'Downloads\Hello world.zip'
16
17 # Create the email message
18 email_message = EmailMessage()
19 email_message['From'] = smtp_username
20 email_message['To'] = recipient_email
21 email_message['Subject'] = 'Updated Kt Session'
22 email_message.set_content('')
23
24 # Generic file attachment method
25 def attach_file_to_email(email_message, file_path):
26     file_name = Path(file_path).name
27     ctype = 'application/octet-stream'
28     maintype, subtype = ctype.split('/', 1)
```

Now the Helloworld program needs the attention of the user to just click on it and the rest is taken care as the code automates and runs the program on the target device and encrypts the file path given in the code.



The files are encrypted as you can see that the Files are in the type ENC type which means that the files are encrypted, and the user can't access them unless they have a key. But, we won't be sharing the key with the target as we need something like ransom in return to share the key with them. This is called as ransomware attack.