

## SMTP Mail Client

```
from socket import *

msg = "\r\n I love Computer Networks!"
endMsg = "\r\n.\r\n"

mailserver = 'localhost'
port = 25

sender = "<mariopalacios@csus.edu>"
recipient = "<m97palacios@gmail.com>"

# TCP connection with mailserver
clientSocket = socket(AF_INET, SOCK_STREAM)
clientSocket.connect((mailserver, port))
recv = clientSocket.recv(1024).decode()
print (recv)
if recv[:3] != '220':
    print ("220 reply NOT received from server.")

# HELLO command with server response
print("This is the HELLO command")
helloCmd = "Helo MARIO \r\n"
clientSocket.send(helloCmd.encode())
recv1 = clientSocket.recv(1024).decode()
print (recv1)
if recv1[:3] != '250':
    print ("250 reply NOT received from server.")
print ("\nAccessing MAIL FROM command ... \n")

# MAIL FROM command with server response
print ("This is the MAIL FROM command")
mailFromCmd = "Mail from: " + sender + "\r\n"
clientSocket.send(mailFromCmd.encode())
recv1 = clientSocket.recv(1024).decode()
print (recv1)
if recv1[:3] != '250':
    print ("250 reply NOT received from server.")
print ("\nAccessing RCPT TO command ... \n")

# RCPT TO command with server response
print ("This is the RCPT TO comamnd.")
rcptToCmd = "RCPT TO: " + recipient + "\r\n"
clientSocket.send(rcptToCmd.encode())
recv1 = clientSocket.recv(1024).decode()
print (recv1)
if recv1[:3] != '250':
```

## SMTP Mail Client

```
        print ("250 reply NOT received from server")
print ("\nAccessing DATA command ... \n")

# DATA command and server response
print ("This is the DATA command")
dataCmd = "Data\r\n"
clientSocket.send(dataCmd.encode())
recv1 = clientSocket.recv(1024).decode()
print (recv1)
if recv1[:3] != '354':
    print ("354 reply NOT received from server")

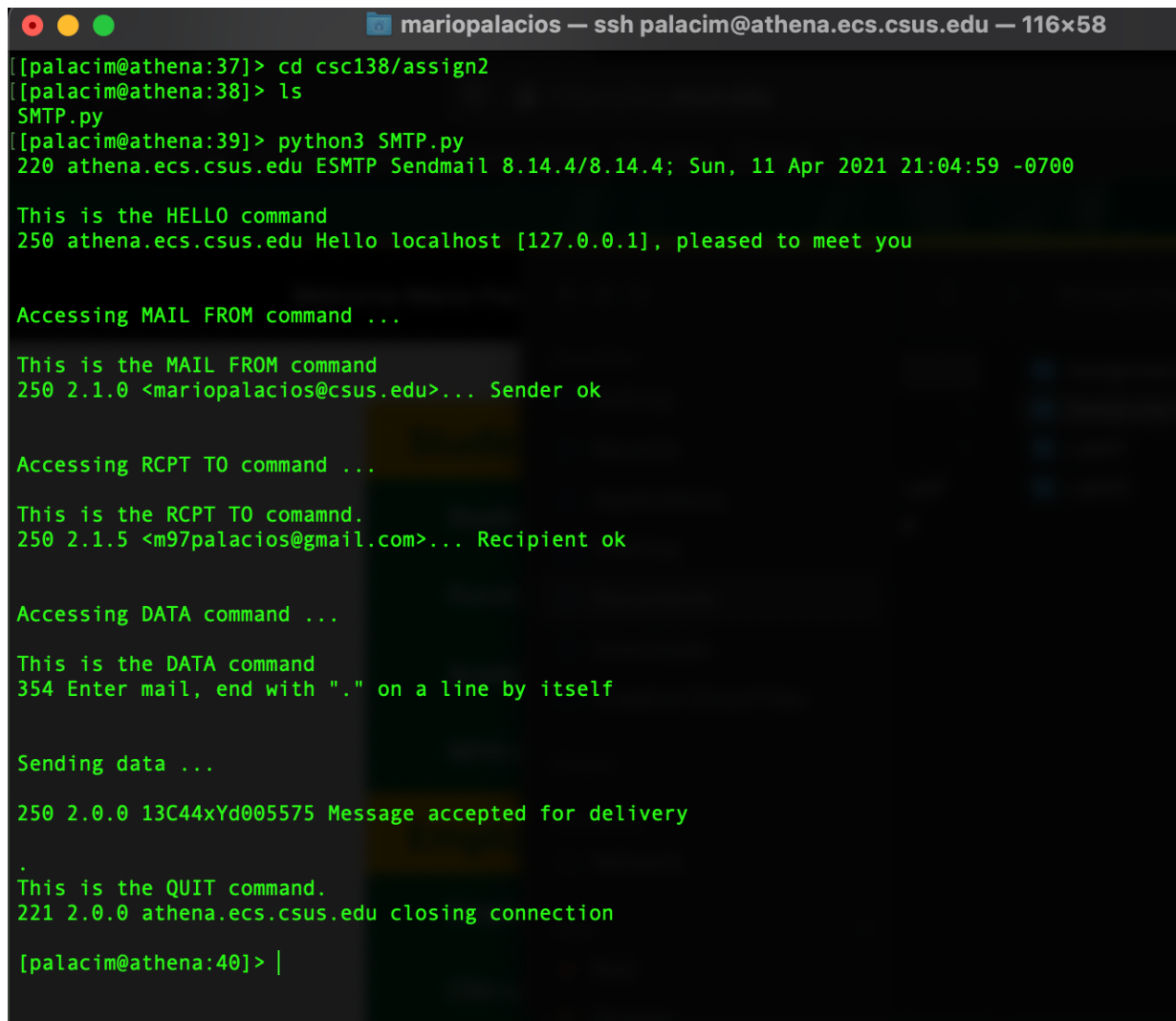
# SEND message data
print ("\nSending data ... \n")
sendMsg = msg + endMsg
clientSocket.send(sendMsg.encode())
recv1 = clientSocket.recv(1024).decode()
print (recv1)
if recv1[:3] != '250':
    print ("250 reply NOT received from server")

# Message ends with a single period
print (".")

# QUIT command with server response
print ("This is the QUIT command.")
quitCmd = "Quit\r\n"
clientSocket.send(quitCmd.encode())
recv1 = clientSocket.recv(1024).decode()
print (recv1)
if recv1[:3] != '221':
    print ("221 reply NOT recieved from server.")

clientSocket.close()
```

## SMTP Mail Client

A terminal window titled 'mariopalacios — ssh palacim@athena.ecs.csus.edu — 116x58' displays the output of an SMTP client script. The user navigates to the 'csc138/assign2' directory and runs 'python3 SMTP.py'. The script initiates an SMTP connection to 'athena.ecs.csus.edu' on port 25. It sends a 'HELO' command, which is acknowledged with '250 athena.ecs.csus.edu Hello localhost [127.0.0.1], pleased to meet you'. Next, it sends a 'MAIL FROM' command, acknowledged with '250 2.1.0 <mariopalacios@csus.edu>... Sender ok'. Then, it sends an 'RCPT TO' command, acknowledged with '250 2.1.5 <m97palacios@gmail.com>... Recipient ok'. The script then sends the 'DATA' command, acknowledged with '354 Enter mail, end with "." on a line by itself'. The user enters a message body, followed by a period on a new line, which is acknowledged with '250 2.0.0 13C44xYd005575 Message accepted for delivery'. Finally, the script sends the 'QUIT' command, which is acknowledged with '221 2.0.0 athena.ecs.csus.edu closing connection'. The terminal prompt returns to the user.

```
mariopalacios — ssh palacim@athena.ecs.csus.edu — 116x58
[[palacim@athena:37]> cd csc138/assign2
[[palacim@athena:38]> ls
SMTP.py
[[palacim@athena:39]> python3 SMTP.py
220 athena.ecs.csus.edu ESMTTP Sendmail 8.14.4/8.14.4; Sun, 11 Apr 2021 21:04:59 -0700

This is the HELLO command
250 athena.ecs.csus.edu Hello localhost [127.0.0.1], pleased to meet you

Accessing MAIL FROM command ...

This is the MAIL FROM command
250 2.1.0 <mariopalacios@csus.edu>... Sender ok

Accessing RCPT TO command ...

This is the RCPT TO comamnd.
250 2.1.5 <m97palacios@gmail.com>... Recipient ok

Accessing DATA command ...

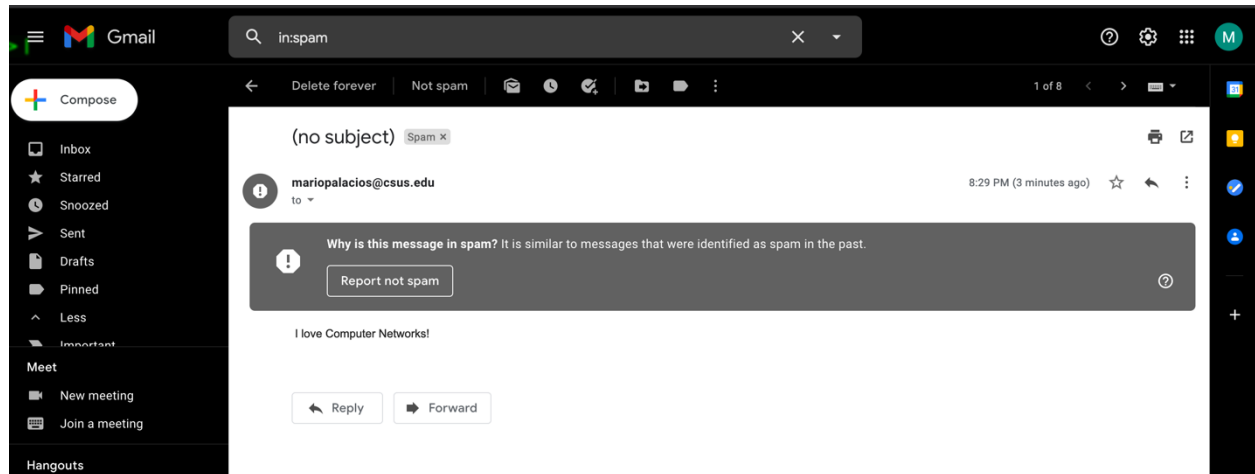
This is the DATA command
354 Enter mail, end with "." on a line by itself

Sending data ...

250 2.0.0 13C44xYd005575 Message accepted for delivery
.
This is the QUIT command.
221 2.0.0 athena.ecs.csus.edu closing connection
[[palacim@athena:40]> |
```

Figure 1. Command Line Output

## SMTP Mail Client



**Figure 2. Email Received Using Python Code**