# Email Spoofing

## Contents

Time required: 90 minutes

Spoofing an email address happens all the time. Just because a message says it is from someone, doesn't mean is.

## Tutorial 1: Setup Brevo Account

https://www.brevo.com/ is an email marketing platform. They offer a Free plan with a daily sending limit of 300 emails. That is plenty for our email spoofing campaign.

1. Go to https://www.brevo.com/ and create a free account.

2. Go to your Account Logon → Click your **Account name** → **SMTP and API**

3. Record your **SMTP Server**, **Port**, **Login** and **SMTP KEY VALUE**.
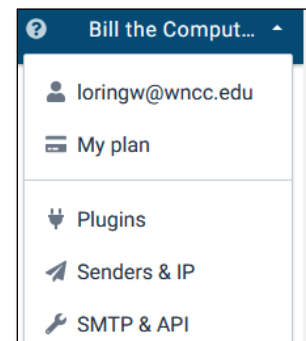
## Tutorial 2: Send Spoofed Email in Python

Sending an email with the Brevo SMTP approach adds some code to the message to show if someone has opened the message or not.

Create a Python program named: **brevo_credentials.py** We will import this information into our email program.

```python
1  #!/usr/bin/env python3
2  """
3      Name: brevo_credentials.py
4      Author:
5      Created:
6      Purpose: brevo email credentials
7  """
8  SMTP_SERVER = "smtp-relay.brevo.com"
9  PORT = 587
10
11  # ---------- REPLACE WITH YOUR INFORMATION -----------#
12  LOGIN = ""
13  PASSWORD = ""
14  # ---------------------------------------------------#
```

Create a Python program named: **brevo.py**

```python
1  #!/usr/bin/env python3
2  """
3      Name: python.py
4      Author:
5      Created:
6      Purpose: Send email through Python with Brevo
7      300 emails per day
8  """
9  from email.message import EmailMessage  # Create email message
10 import smtplib  # Send email through an smtp server
11 import ssl  # Create secure context to send mail
12 import brevo_credentials  # Import credentials
```

This code imports the libraries needed to send a secure email. Everything is encrypted in transit.

```
21    # -------------- CREATE MESSAGE --------------------------------------#
      CodiumAI: Options | Test this function
22    def create_message():
23        """Create an email message with specific content and recipients."""
24
25        # ------------- REPLACE BELOW WITH YOUR INFORMATION ------------------#
26        # A list containing one or more email addresses
27        email_from = "William Loring <williamloring@hotmail.com>"
28        email_dst = [
29            "williamaloring@gmail.com",
30            "williamloring@hotmail.com"
31        ]
32        content = """\
33        This test message is sent from Python Bill."""
34        subject = "Email from Ethical Hacking Class from Brevo"
35        # ------------- REPLACE ABOVE WITH YOUR INFORMATION ------------------#
36
37        msg = EmailMessage()
38        msg["From"] = email_from
39        msg.set_content(content)
40        msg["Subject"] = subject
41        msg["To"] = email_dst
42
43        return msg
```

The **create_message** function creates the email message object we are going to send. creates an email message with specific content and recipients. We return that object to the main function.

- Replace the **email_from**, **email_dst**, and **content** variables with your information.

- Edit the subject as needed.

```python
46    # -------------- SEND MESSAGE -------------------------------------------#
      CodiumAI: Options | Test this function
47    def send_message(msg):
48        """Send the prepared email message."""
49        # Set email server object
50        server = smtplib.SMTP(
51            brevo_credentials.SMTP_SERVER,
52            brevo_credentials.PORT
53        )
54
55        # Show all communication with the server
56        server.set_debuglevel(True)
57
58        # Create a secure SSL context
59        context = ssl.create_default_context()
60
61        # -------------------- SEND EMAIL -----------------------------------#
62        try:
63            # Start an encrypted TLS session
64            # This will encrypt our password and message
65            server.starttls(context=context)
66
67            # Login to the mail server
68            server.login(brevo_credentials.LOGIN, brevo_credentials.PASSWORD)
69
70            # Send the email message
71            server.send_message(msg)
72
73            print("****************************************")
74            print("*    Email message successfully sent.   *")
75            print("****************************************")
76        except Exception as e:
77            print(e)
78        finally:
79            # Quit from server
80            print(server.quit())
81
82
83    # Call the main function
84    if __name__ == "__main__":
```

The **send_message(msg)**: function sends a prepared email message.

- It establishes a connection with the Brevo SMTP server using smtplib.

- It creates a secure TLS connection using ssl for secure communication.

- It logs in to the server with your Brevo credentials.

- It sends the email message and prints a confirmation message.

Send an email to one of your email accounts.

Read the message.

Go back to [www.Brevo.com](www.Brevo.com) Go to **Transactional** tab → **Statistics**.

Scroll down to recent activity. You can see the messages that are sent and opened. An html tag is inserted in the message to track whether the end user received the email message or not.

Take a screenshot of your successfully sent email message to be submitted with this assignment.


## Tutorial 3: Send Spoofed Email with sendemail with Kali

sendemail with Kali allows us to better spoof a target.

1. Set the Kali Linux virtual machine to a Bridged Adapter.

2. At a terminal prompt type: **sendemail** Look through the help information.

3. At a terminal prompt → **geany send_email.sh**

**NOTE: Put everything on one line as this is a shell script.** The example below is separated onto separate lines for easier reading.

4. Replace the generic information with your Brevo information.

5. **CTRL-S** will save the file. **CTRL-X** will exit nano.

```
sendemail -o tls=no
-xu brevo_login
-xp brevo_password
-s smtp-relay.brevo.com:587
-f "from_email_address"
-t "to_email address"
-u "A test of this stuff"
-m "This is an email test of send in blue"
-o message-header="From: Spoofed Name <Spoofed Email address>"
-v
```

6. At the terminal → **chmod 744 send_email.sh**

7. At the terminal → **./send_email.sh**

8. The email should be sent.

Explanation of program and switches.

| sendemail | Send email program in Kali |
|-----------|----------------------------|
| -xp | Username for SMTP authentication |
| -s | SMTP mail server (relay) |
| -f | From (sender) email address |
| -t | To (recipient) email address |
| -u | Message subject |
| -m | Message body |
| -o | Advanced option, create a spoofed header |
| -v | Verbose mode for debugging |

# Assignment

1. Send yourself a spoofed email using each method.

2. Take a screenshot of the received email.

3. Compare the two email messages.

## Assignment Submission

Attach the complete program, script, and screenshot of each spoofed email to the assignment in Blackboard.