

# Securing Email Attachments

An industrial Use Case

Gaurav Mittal

DataScienceManager @ ThermoFisher



# Business Scenario

Emails sent as part of business with attachments to customers are prone to phishing email attacks.

## Potential Risks

- **Man-in-the-Middle (MitM) Attacks**

An attacker could intercept the email and its attachments during transmission.

- **Insecure Email Servers**

If the email servers are not properly secured, they could be vulnerable to hacking

- **Inadequate Encryption**

Easily accessed by attackers who gain unauthorized access to the transmission

GM

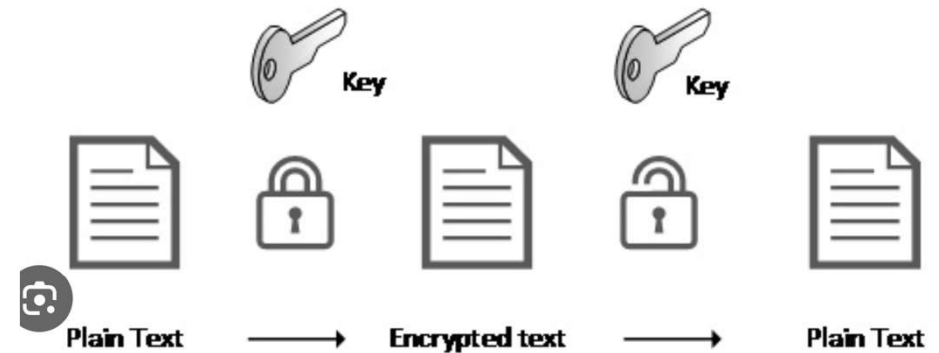
Avoiding **Man-in-the-Middle** Attacks



# Data Security

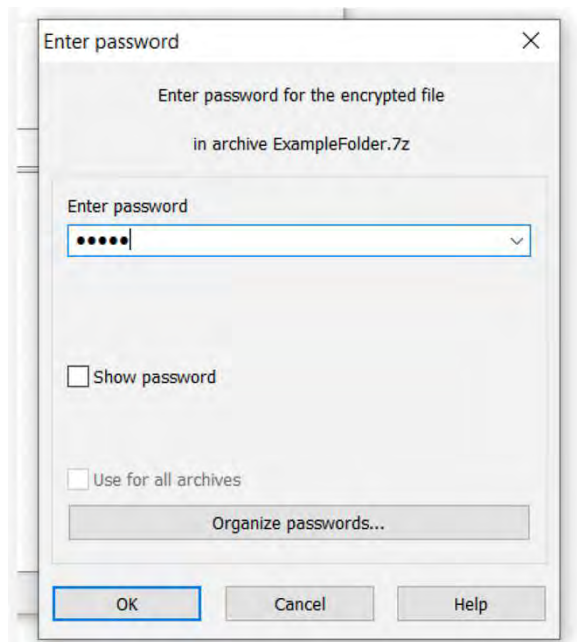
Safeguard data from unauthorized access or theft throughout its lifecycle.  
Key aspects of data security include:

- Data Encryption
- Authentication and Authorization
- Availability



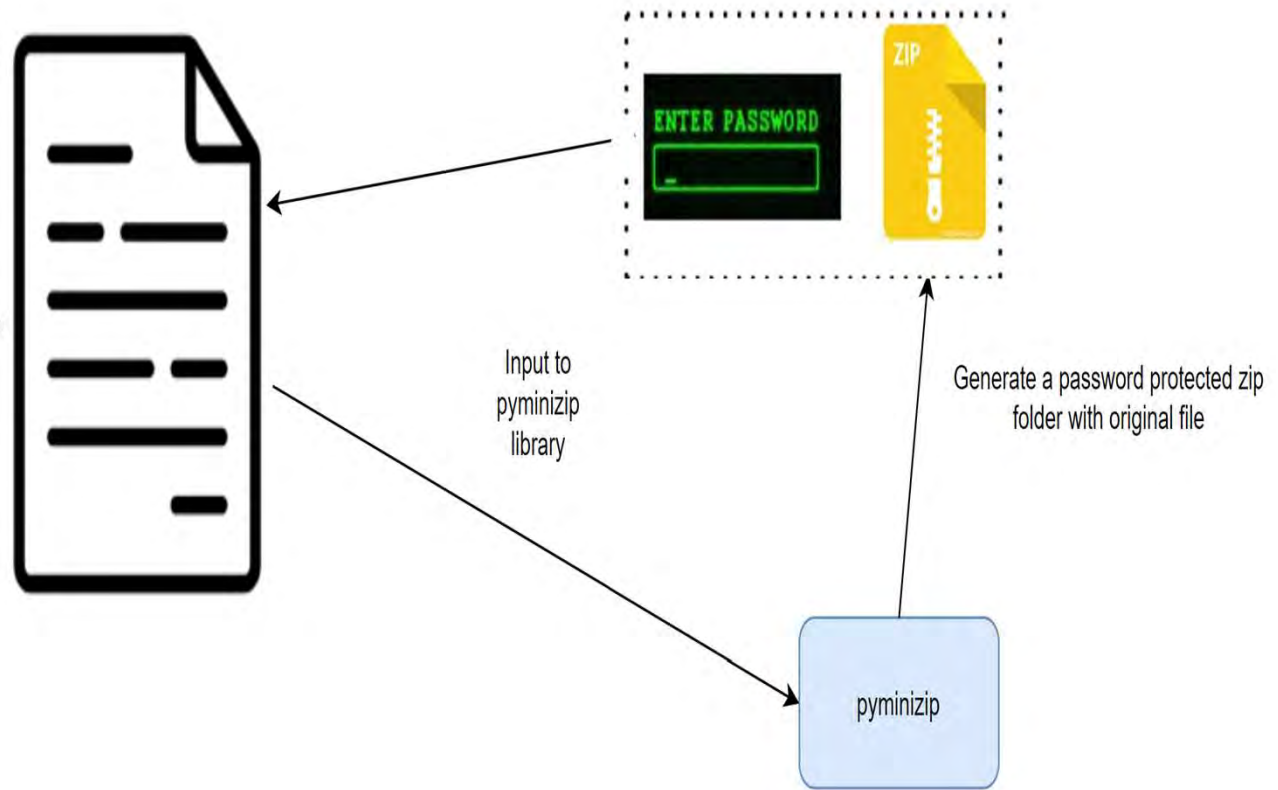
# How to secure email attachments?

- User Education & Awareness
- Antivirus Software
- Encryption
  - Password Protection
- Access Control
  - Signed URL



# Pyminizip Library

- The pyminizip module provides simple support for creating password-protected Zip archives.



`pyminizip.compress(src_file, dest_file, password, level)`

Create a password-protected zip archive from a single file. The *level* parameter determines the compression level, 0 is the default, other possible values are 1 (fastest) to 9 (most compression).

# Adding AWS Layer - pyminizip

## Lambda Layers

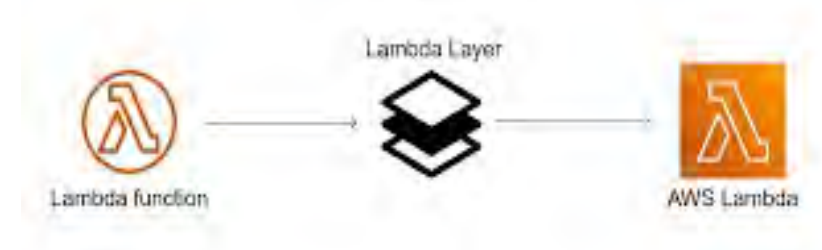
A .zip file archive that contains supplementary code or data.

## Steps

1. Install pyminizip in local

➤ `pip install pyminizip -t .`

2. Add Layer to Lambda Function



### Runtime settings [Info](#)

Runtime

Python 3.7

Handler [Info](#)

lambda\_function.lambda\_handler

▶ Runtime management configuration

### Layers [Info](#)

Merge order	Name	Layer version	Compatible runtimes	Compatible architectures
1	pyminizip	13	python3.7	x86_64



# S3 pre-signed URLs

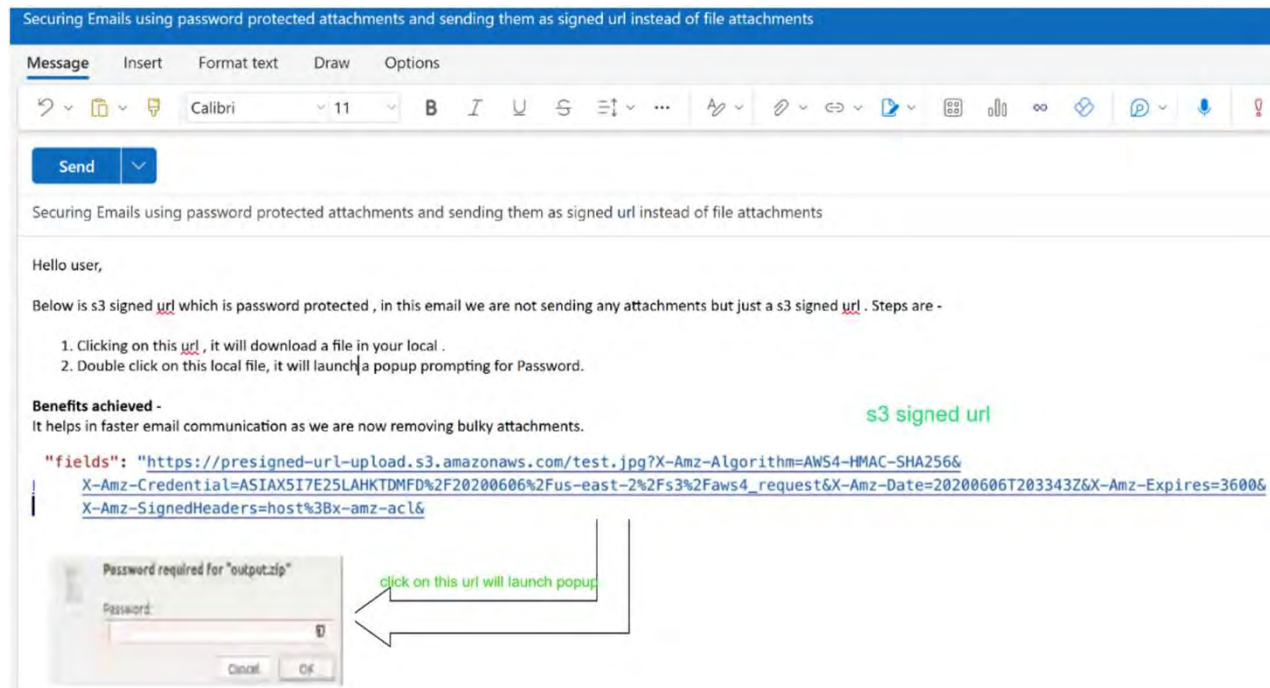


- Grant time-limited access to objects in S3 without updating your bucket policy
- Credentials that you can use to create a presigned URL:
  - **IAM instance profile** – Valid up to 6 hours.
  - **AWS Security Token Service** – Valid up to maximum 36 hours when signed with long-term security credentials or the duration of the temporary credential, whichever ends first.
  - **IAM user** – Valid up to 7 days when you're using AWS Signature Version 4.

PreSignedURL Benefits
Controlled Access - You decide whom to grant access
Security - No need to share AWS credentials
Flexibility - Can be used for upload/download objects

# Pre-Signed URL

- No Attachments
- Control duration for which this signed URL is valid



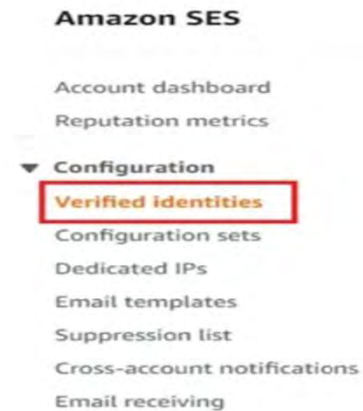


## AWS SES (SendEmailService)

---

- Create verified identity
- Sent email to verified, unverified entity
- 40 emails only per sec

Step 1: Open the SES service in AWS console. Click verified identities:



Step 2: Click “Create identity”



Step 3: Select “email address” from the identity details options, as we are creating an email identity. Enter the email address to verify, for example: here I want to verify my email address : ,

Click “create identity”.



# Final Code

```
def read_csv_from_s3():
```

```
    pyminizip.compress(f"/tmp.csv", 'user', f"/tmp.zip", password, 9)
```

compressing file and providing  
password, compress level

```
    url = s3.generate_presigned_url(  
        ClientMethod='get_object',  
        Params={  
            'Bucket': os.environ["s3Obj"],  
            'Key': f"archive/user.zip"  
        },  
        ExpiresIn = 604800
```

1 week timeframe

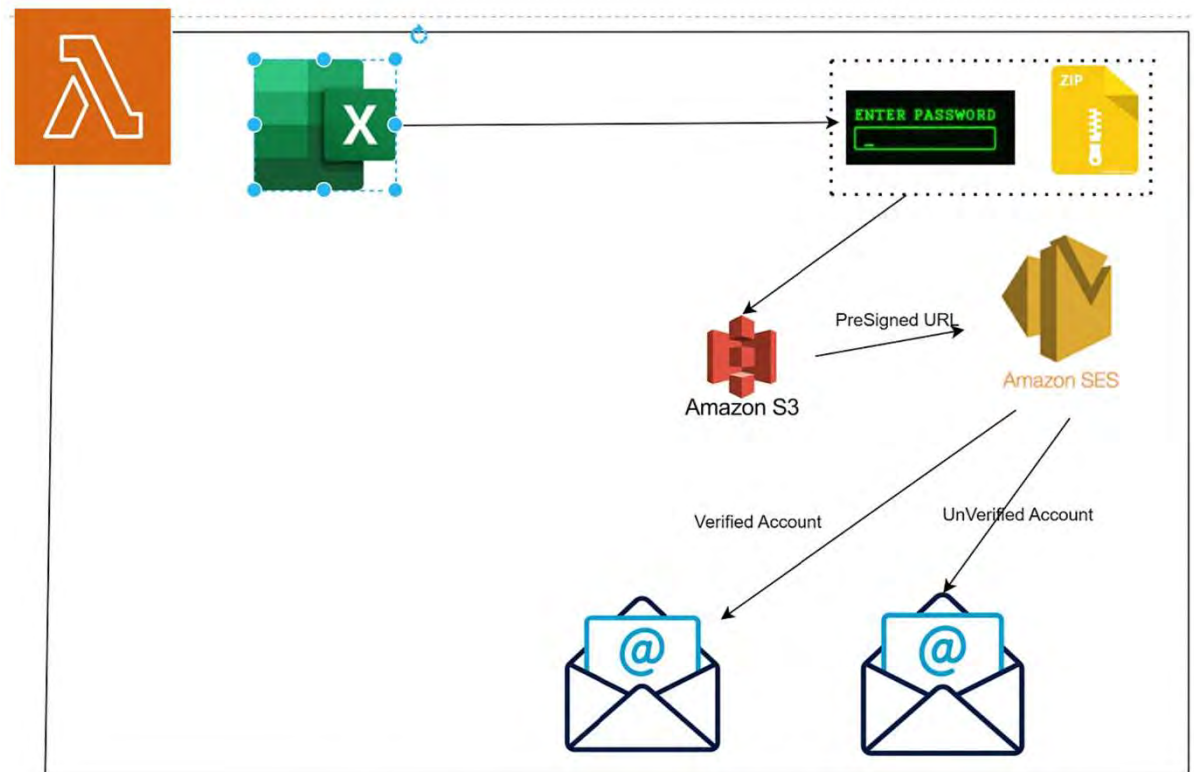
```
)  
    #sending email using SMTP relay to respective users  
    send_mail(username, url, recipient)  
    time.sleep(1)
```

SES time limit

# Complete Architecture

## Steps Performed –

1. Generating attachment
2. Encryption using Password protection libraries.
3. Preserve file in s3 bucket and generate s3 signed URL.
4. Using SendEmailService, send attachments in secure fashion.



Architecture for sending emails in secure fashion



Gaurav Mittal

Manager – ThermoFisher Scientific

 GauravM85

Article -

[Implementing Email Attachment Security](#)