

Email Automation

The purpose of this document is to explain the process of determining the top classification of emails in order for the system to send an automated email response. It also explains how the email will be sent back to the user using Amazon SES.

Overview.

Once a workmail account is set up on amazon to receive emails from various users, the emails will be stored in buckets as txt files as they go through various stages of the whole email automation process. For this stage to begin, the high_score lambda function will be triggered by an event on the Lambda_unzip bucket. The Lambda_unzip bucket gets its data from a function named my_unzip_lambda which unzips a file from the job classification batch. From the job classification batch, the zip file is stored in the S3 bucket named [comprehend-custom-classifier-de-ids3bucketecb34074-q5ilzz2skafm](#). Using Amazon SES, the system then automatically sends back emails to users based on what they asked on their emails.

1. Extraction Lambda:

- Purpose: Extracts a JSONL file from a ZIP file created by a batch job.
- Action:
 - Receives a ZIP file from a job classification batch containing a JSONL file. From the job classification batch, the zip file is stored in the S3 bucket named [comprehend-custom-classifier-de-ids3bucketecb34074-q5ilzz2skafm](#).
 - An S3 event triggers the my_unzip_lambda to start extracting the jsonl file from the zip file.
 - The lambda function then unzips the file and stores the unzipped version of the file in the Lambda Unzip S3 bucket.

2. Lambda Unzip Bucket:

- Purpose: Triggers the High Score Lambda function upon file upload.
- Action:
 - Monitors for new JSONL files uploaded to the bucket.
 - Triggers the High Score Lambda function when a new file is detected.

3. High Score Lambda:

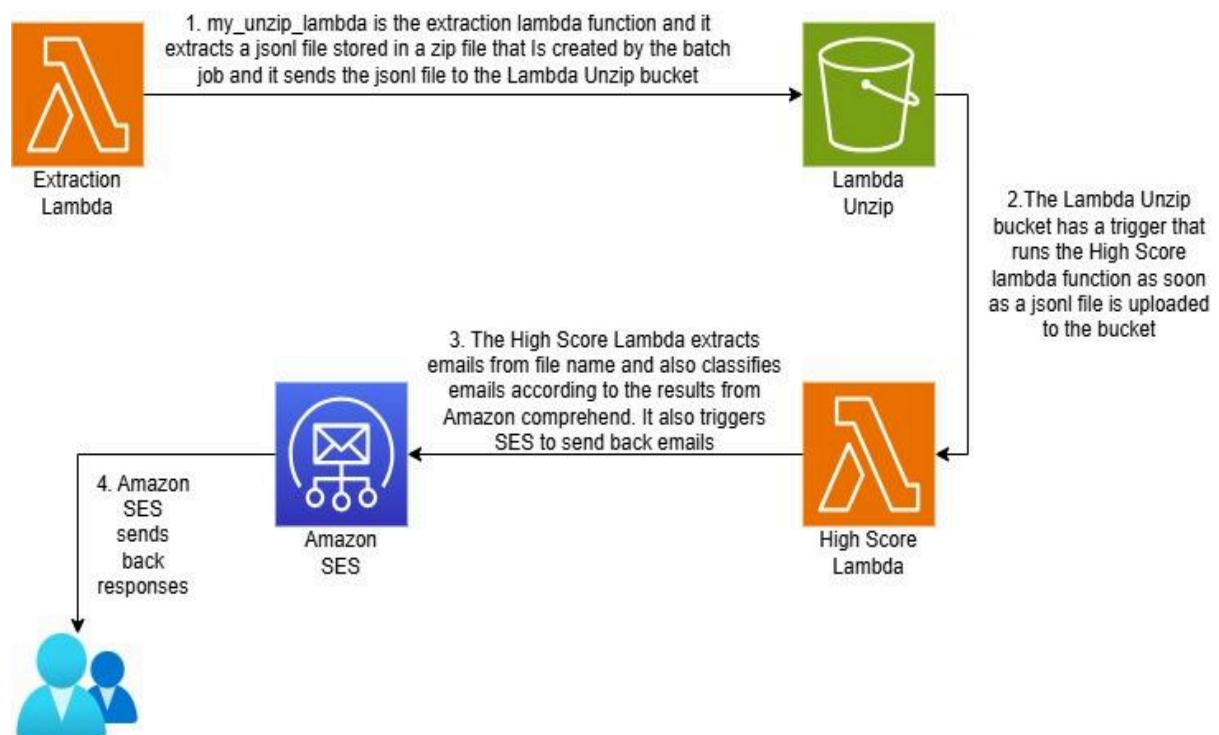
- **Purpose:** Extracts emails from file names, classifies emails using Amazon Comprehend results, and triggers Amazon SES to send back emails.
- **Action:**
 - Receives the JSONL file from the Lambda Unzip bucket.
 - The Lambda function is triggered by an upload event to an S3 bucket. It retrieves the uploaded file from S3

- A function `get_email_address` is used to extract the email address from the parsed JSON data. The system is designed to save the sender emails on the file name between "<>" symbols and this is where the function will extract email addresses.
- Another function `get_highest_score_classification` identifies the classification with the highest score within the JSON data.
- Based on the classification type, the function `send_notification_email` sends an email notification to the extracted email address. Different email templates are used based on the classification type.

4. Amazon SES:

- **Purpose:** To send back responses to user emails.
- **Limitation:** In a sandbox environment, you can use all of the features offered by Amazon SES; however, certain sending limits and restrictions apply. There is a daily limit of 200 emails that you are allowed to send while using a sandbox environment. If your account is still in the Amazon SES sandbox, you also need to verify any email addresses which you plan on sending email to.
- **Action:**
 - Receives email sending requests from the "High Score" Lambda function.
 - Sends back responses to the email addresses extracted from the file names.
 - There are three templates of emails stored in SES that can be sent back as responses depending on which classification the High Score Lambda function determines as the top one. In our test cases, the Top classification is either MONEYTRANSFER, PROMOCODE and PASSRESET. This means that the stored email templates in SES are for those three classifications.

High_Score_function



The code is designed to be triggered by an S3 event. This function is triggered by the S3 PutObject event. It retrieves the JSON file from the Lambda Unzip bucket then goes through the JSON data. The JSON data should contain a 'File' field with the email address and a 'Classes' field with an array of classification objects.

- For each JSON record, it extracts the "File" field which contains an email address that will be used when sending back an email.
- It uses the `get_highest_score_classification` function to determine the top classification based on the record's "Classes" field. This function identifies the classification with the highest 'Score' value
- It sends a notification email using the `send_notification_email` function based on the classification.

Send_notification_email helper function

- The purpose of this helper function is to send a notification email to the specified email address based on the classification type.
- The `classification['TopClassification']` is used to select the appropriate email template between `MONEYTRANSFER` for money transfer notifications, `PROMOCODE` for promo code notifications and `PASSRESET` for password reset requests.

- The template data is prepared based on the selected template and the classification information.
- The `ses_client.send_templated_email` method is used to send the email. The Source, Destination, Template, and TemplateData parameters are specified accordingly.
- The function includes error handling for various exceptions that might occur during the email sending process, such as message rejection, domain verification issues, configuration set issues, and general exceptions.
- The function returns a JSON response with the processing results or an error message if encountered. S3 Access Errors if there's an issue accessing or processing the S3 object, an error message is logged and returned. SES Sending Errors if an error occurs while sending the email, the specific error message is logged, and an appropriate response is returned.
- If an email is not verified on SES, the function will skip that email and move on to the next verified email then send back the appropriate email based on the highest score.