

# Configuration Manual

MSc Research Project  
MSc Cloud Computing

Ganyashree Suvarna  
Student ID: x22242864

School of Computing  
National College of Ireland

Supervisor: Sudarshan Deshmukh

**StudentName:** Ganyashree Sadashiv Suvarna

**Student ID:** x22242864

**Programme:** MSc in Cloud Computing

**Year:** 2023-2024

**Module:** MSc Research Project

**Supervisor:** Sudarshan Deshmukh

**Submission**

**Due Date:** 12-Aug-2024

**Project Title:** Secure And Verifiable Cloud Based Data Sharing For Law Enforcement with Sensitive Information And Encryption.

**Word Count:** 700

**Page Count:** 6

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

**Signature:** Ganyashree Suvarna

**Date:** 12-Aug-2024

**PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST**

|                                                                                                                                                                                           |                          |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------|
| Attach a completed copy of this sheet to each project (including multiple copies)                                                                                                         | <input type="checkbox"/> |
| Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).                                                                         | <input type="checkbox"/> |
| You must ensure that you retain a <b>HARD COPY</b> of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer. | <input type="checkbox"/> |

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

|                                  |  |
|----------------------------------|--|
| <b>Office Use Only</b>           |  |
| Signature:                       |  |
| Date:                            |  |
| Penalty Applied (if applicable): |  |

# Configuration Manual

Ganyashree Sadashiv Suvarna

Student ID: x22242864

## 1. Introduction:

This manual provides a detailed overview of the tools, technologies, and configurations required to implement the research model described in the project. It is organized into several sections: Section 2 covers the environment setup, Section 3 details the tools and software used, and Section 4 explains the implementation steps.

## 2. Environmental Setup:

Below mentioned configuration was used to implement the model.

**Processor:** Intel i5 or higher

**Memory:** 8GB RAM

**Operating System:** Ubuntu 20.04 LTS or Windows 10

**Programming Language:** Python 3.x

**Python Environment:** Jupyter Notebook, Anaconda Navigator

**Cloud Environment:** AWS Cloud Services (S3, Lambda, SageMaker)

## 3. Tools and Software used:

This section outlines the tools and software required for the project, along with instructions for installation.

### 3.1 AWS CLI Installation

#### 1. Download and Install AWS CLI:

- o Go to the [AWS CLI official documentation](#) and follow the instructions to install the AWS Command Line Interface on your system.

#### 2. Configure AWS CLI:

- o After installation, open a terminal or command prompt and run the command `aws configure`.

## 4. Implementation of the model:

This section provides step-by-step instructions on how to implement the project.

### 4.1 Setting Up S3 Buckets

#### 1. Create S3 Buckets:

- o Log in to your AWS account and navigate to the S3 service.

- Create two buckets:
  - Project Files Bucket:** This will store the Python scripts, encryption keys, and processed datasets.
  - Dataset Bucket:** This will store the raw datasets for processing.

## 4.2 Configuring AWS Lambda

### 1. Create a Lambda Function:

- In the AWS Management Console, go to the Lambda service and click 'Create Function'.
- Choose 'Author from scratch', name your function, and select Python 3.x as the runtime.

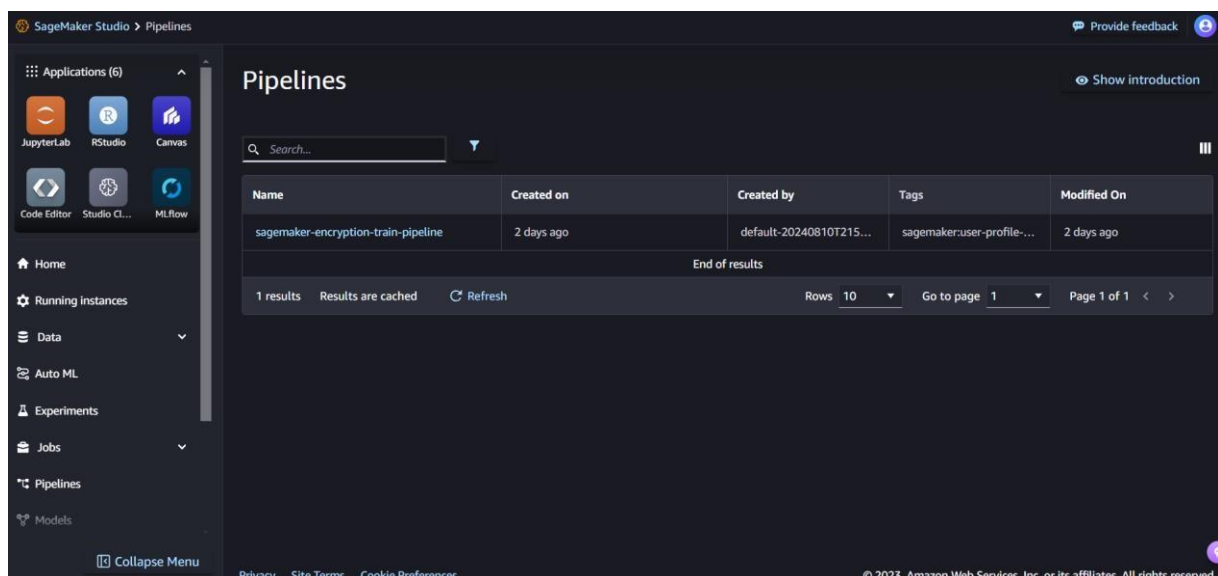
### 2. Set Up Trigger for S3:

- Under the 'Function code' section, add the necessary Python script (provided in the project) to handle dataset processing.
- Set the S3 bucket as the trigger for this Lambda function to automatically invoke it when a new file is uploaded.

## 4.3 Setting Up SageMaker

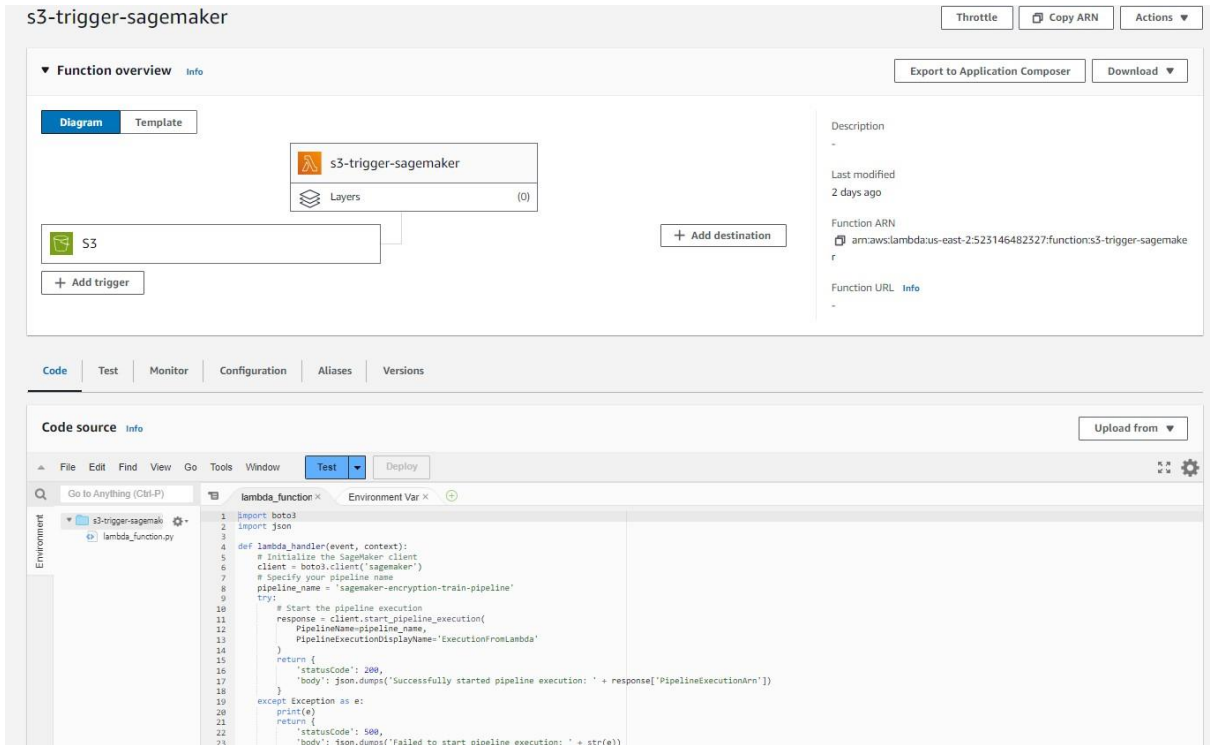
### 1. Create a SageMaker Pipeline:

- Use the SageMaker console or SDK to define a pipeline that includes the steps for PII detection, encryption, and decryption.
- Integrate a custom Docker image stored in ECR that contains all dependencies required for the processing script.
- Click on Domains and create the user and launch the studio, click on Jupyter Space and write code in the Notebook.



## 2. Link SageMaker with Lambda:

- Ensure the Lambda function triggers the SageMaker pipeline when a new dataset is uploaded.
- Add S3 trigger in Lambda, insert dataset in S3 and trigger the pipeline.



## 4.4 Data Processing and Encryption

### 1. PII Detection and Encryption:

- The Python script (ric.py) will handle the detection of PII within the datasets using predefined regular expressions.
- Detected PII will be encrypted using the Fernet encryption algorithm, with the encrypted data stored securely back in the S3 bucket.

### 2. Decryption Process:

- Authorized entities can trigger the decryption process using another Lambda function linked to the same SageMaker pipeline.

## 5. Security and Compliance

### 1. Access Control:

- o Implement strict IAM roles to control access to the S3 buckets and AWS services used in the project.

## 2. Logging and Monitoring:

- o Utilize AWS CloudWatch to monitor and log all actions within the system, ensuring compliance with data protection regulations.

**CloudWatch** ×

CloudWatch > Log groups > /aws/sagemaker/ProcessingJobs > pipelines-kahggf68h5jt-EncryptionTraining-tXq0095JdY/algo-1-1725326852

**Log events**

You can use the filter bar below to search for and match terms, phrases, or values in your log events. [Learn more about filter patterns](#)

**Message**

No older events at this moment. [Retry](#)

```
<class 'pandas.core.frame.DataFrame'>
```

RangeIndex: 1000 entries, 0 to 999

Data columns (total 22 columns):

| #  | Column                              | Non-Null Count | Dtype  |
|----|-------------------------------------|----------------|--------|
| 0  | Name                                | 1000 non-null  | object |
| 1  | Age                                 | 1000 non-null  | int64  |
| 2  | Crime Committed                     | 1000 non-null  | object |
| 3  | Date of Crime                       | 1000 non-null  | object |
| 4  | Location                            | 1000 non-null  | object |
| 5  | Sentence Length                     | 1000 non-null  | int64  |
| 6  | Gender                              | 1000 non-null  | object |
| 7  | Occupation                          | 1000 non-null  | object |
| 8  | Email                               | 1000 non-null  | object |
| 9  | Phone Number                        | 1000 non-null  | object |
| 10 | STATISTIC                           | 1000 non-null  | object |
| 11 | Statistic Label                     | 1000 non-null  | object |
| 12 | TLIST(AI)                           | 1000 non-null  | object |
| 13 | Year reported                       | 1000 non-null  | int64  |
| 14 | CO2488V8                            | 1000 non-null  | int64  |
| 15 | Crime Cat                           | 1000 non-null  | int64  |
| 16 | CB2876V8                            | 1000 non-null  | int64  |
| 17 | Age of victim                       | 1000 non-null  | object |
| 18 | CB4825V8                            | 1000 non-null  | int64  |
| 19 | Nature of relationship with suspect | 1000 non-null  | object |
| 20 | UNIT                                | 1000 non-null  | object |
| 21 | VALUE                               | 1000 non-null  | int64  |

dtypes: int64(8), object(14)

memory usage: 172.0+ KB

File <cryptography.fernet.Fernet object at 0x7f99c3ac0fa0> uploaded to projectfiles222428641aw

| File | Name                                               | Age | ... | UNIT | VALUE |
|------|----------------------------------------------------|-----|-----|------|-------|
| 0    | gAAAAABnt-GjthxQ18PiwntsRn1zAcqMc7dh8ysQKp5LPK...  | 23  | ... | %    | 6     |
| 1    | gAAAAABnt-Gjprgri4ihex5nPkskj1967b5gZrrvUjasm0S... | 37  | ... | %    | 21    |
| 2    | gAAAAABnt-GjXv8et2nuML6ooGAABECZqTjAPc2bwbk1y...   | 64  | ... | %    | 20    |
| 3    | gAAAAABnt-Gjte3EBC3npv1gVGZV2J2esahh32wPtl-5Qd...  | 62  | ... | %    | 21    |
| 4    | gAAAAABnt-GjK8to5KnRk1rbh5VXxEnd8UEhonZLcc1k12B... | 68  | ... | %    | 20    |
| ...  | ...                                                | ... | ... | ...  | ...   |
| 995  | gAAAAABnt-Gje3sMY88q_8iLR_35Fw7fE3BxGNAN5MM71K...  | 52  | ... | %    | 17    |
| 996  | gAAAAABnt-GjqlBJKzKzMT8AzHpVgeLUho7ixXhqfIDcD7...  | 51  | ... | %    | 18    |
| 997  | gAAAAABnt-Gj8Gujds8g19YjK9apfXtN3pwkult5DX5_fg...  | 70  | ... | %    | 8     |
| 998  | gAAAAABnt-G5Kt41adzhTfVWf5f4Wren7m7GCTfH10n...     | 61  | ... | %    | 16    |

## 6. Testing and Validation

### 1. Unit Testing:

- o Test each component of the system individually to ensure it functions correctly.

### 2. Integration Testing:

- o Perform end-to-end testing of the entire pipeline, from data upload to encrypted data storage, to validate the system's performance and security.

## **7. Deployment and Maintenance**

### **1. Deployment:**

- o Deploy the system in the production environment using the configurations tested and validated during development.

### **2. Maintenance:**

- o Continuously monitor the system for performance and security, applying updates as necessary.