Professional Report: Real-Time Data Ingestion Using Kinesis Firehose and Redshift

# 1. Introduction

This report outlines the implementation of a real-time data ingestion pipeline using AWS Kinesis Firehose, AWS Lambda, Amazon Redshift, and Amazon S3. The objective is to capture and store live clickstream data from a website or mobile application for real-time analytics and decision-making.

# 2. Architecture Overview

The system architecture consists of multiple AWS services working in coordination to ensure efficient real-time data ingestion. Below is the flow of data through the pipeline:

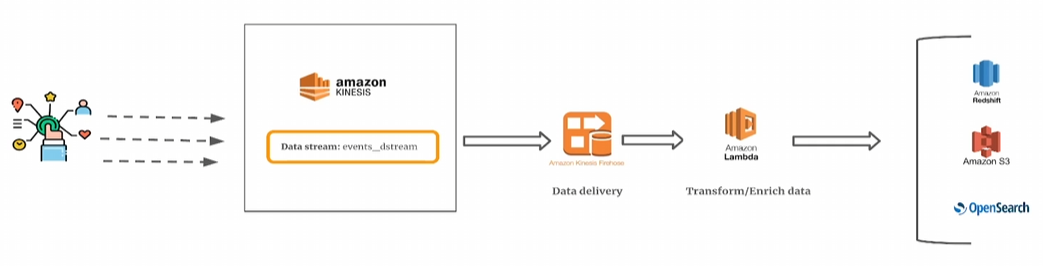
1. \*\*Clickstream Data Generation\*\*: User interactions (clicks, views, etc.) are streamed into an AWS Kinesis Data Stream.

2. \*\*Kinesis Firehose\*\*: Firehose delivers the incoming raw data in batches to a designated destination such as Redshift, S3, or OpenSearch.

3. \*\*AWS Lambda Transformation\*\*: The streaming data is processed and enriched using an AWS Lambda function before being stored.

4. \*\*Amazon Redshift & S3\*\*: The transformed data is finally ingested into Redshift for analysis and stored in S3 for backup and further processing.

### Architecture Diagram:



# 3. Data Processing Pipeline

The data processing pipeline follows these steps:

1. A Python script simulates real-time data generation from website user interactions.

2. The script pushes data to the Kinesis Data Stream for processing.

3. Kinesis Firehose buffers the data and processes it before sending it to Redshift.

4. A Lambda function enriches the data with additional attributes before ingestion.

5. The final transformed data is stored in Redshift for query execution and analysis.

# 4. Lambda Function for Data Transformation

The AWS Lambda function plays a crucial role in cleaning and transforming data before ingestion. It performs the following operations:

1. Identifies if a user is browsing from a \*\*mobile or desktop\*\*.

2. Categorizes the traffic source (e.g., Social Media, Search Engine, Direct, Email).

3. Formats data into a Redshift-compatible CSV format.

# 5. Querying Data in Amazon Redshift

Once the data is ingested into Redshift, it can be queried using SQL. Below is an example SQL query to fetch clickstream data:

```sql  
SELECT user\_id, session\_id, source, is\_mobile FROM prod\_schema.events LIMIT 10;  
```

# 6. Conclusion

This real-time data ingestion pipeline enables businesses to process high-volume clickstream data efficiently. The integration of AWS services like Kinesis Firehose, Lambda, Redshift, and S3 ensures scalability, reliability, and real-time analytics for better business insights.