Design and Implementation of a Real-Time Data Streaming Pipeline Using AWS

Author: Joshua Paul Muppidi

Date: May 20, 2025

Table of Contents

1. Abstract

This project presents a cloud-native, real-time data streaming architecture using Amazon Web Services (AWS). It focuses on capturing live event data via API Gateway, streaming it using Amazon Kinesis, processing with AWS Lambda, storing structured records in DynamoDB, and archiving raw data in Amazon S3. The objective is to build a scalable, event-driven pipeline that is serverless, cost-effective, and production-ready. The report includes architecture, implementation, logs, screenshots, and results.

2. Introduction

Real-time data processing is essential for responsive systems in modern applications. This project leverages AWS managed services to implement a pipeline that captures, processes, and stores JSON payloads sent through HTTP POST requests. By using serverless components, the architecture ensures low operational overhead and horizontal scalability.

3. Problem Statement & Objectives

- Create an end-to-end real-time event tracking pipeline
- Use AWS native tools like Lambda, Kinesis, DynamoDB, and S3
- Ensure each component logs data and is testable
- Provide working screenshots as validation

4. Tools and Technologies

The following AWS services and tools were used in the implementation:
- Amazon API Gateway
- AWS Lambda (Producer & Consumer)
- Amazon Kinesis Data Stream
- Amazon DynamoDB
- Amazon S3
- AWS CloudWatch
- IAM Roles and Policies
- cURL (for testing API Gateway endpoint)

5. System Architecture

The system consists of the following components:

1. API Gateway: Accepts external POST requests
2. Lambda Producer: Sends event data to Kinesis
3. Kinesis Data Stream: Buffers and routes the data
4. Lambda Consumer: Consumes data from Kinesis and processes it
5. DynamoDB: Stores structured parsed events
6. S3: Archives raw event payloads
7. CloudWatch: Logs Lambda events and errors

(See GitHub repository for architecture diagram and screenshots.)