Real-Time Weather Data Pipeline for Weather Analytics

AWS Data Engineering Bootcamp Project #3

Objective

The goal of this project is to build a real-time weather data pipeline leveraging AWS serverless technologies. This pipeline is designed to ingest, process, store, and visualize weather data to enable:

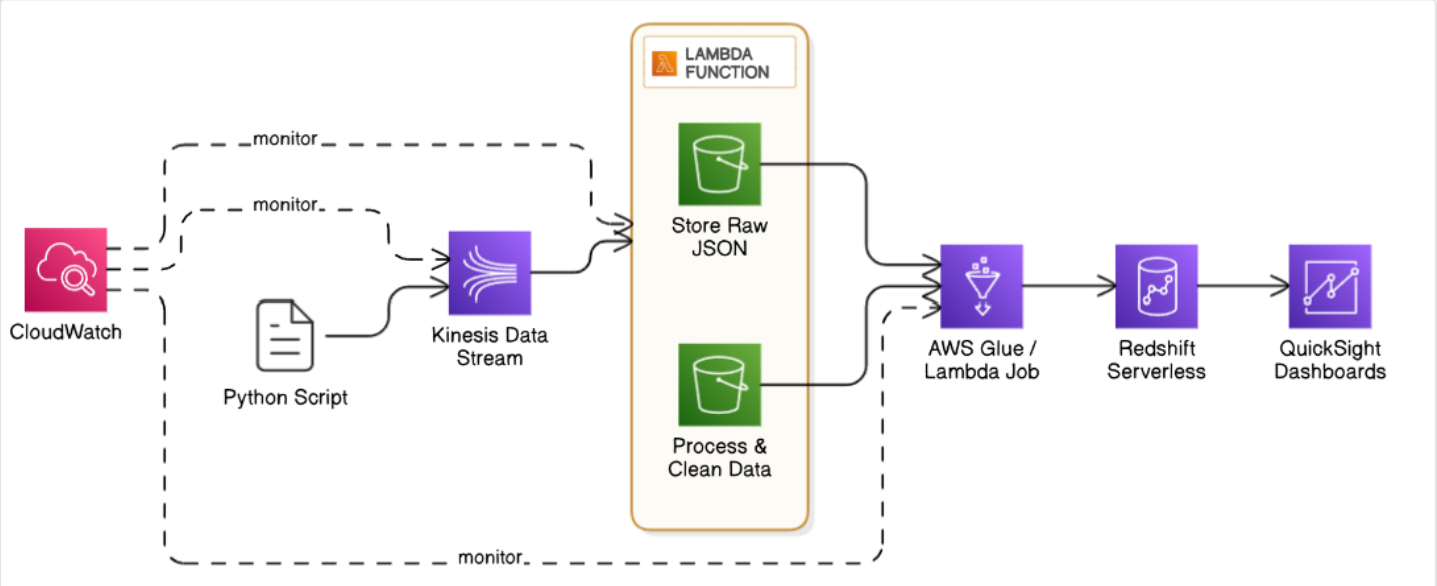
- Real-time monitoring of weather conditions
- Historical trend analysis for forecasting
- Automated data processing with low latency
- Data-driven insights through interactive dashboards

Key outcomes:

- Achieved a 90% reduction in manual data processing time
- Designed a scalable and cost-efficient architecture using AWS Kinesis, Lambda, Redshift, and QuickSight

---

System Architecture



Prerequisites

- AWS account with access to Kinesis, Lambda, Redshift, S3, and IAM
- Python 3.x for Lambda functions and data simulation
- OpenWeatherMap API key for weather data simulation
- Optional: Terraform or other Infrastructure as Code tools for automation

---

Component Breakdown

A. Data Ingestion (Kinesis)

- Kinesis Data Stream named `weather-stream` configured with on-demand capacity
- Python data producer script (`weather_stream-project-3.py`) sending weather data every 60 seconds

B. Data Processing (Lambda)

- Lambda function (`weatherStreamFunction`) triggered by Kinesis events
- Performs data validation, converts temperatures from Kelvin to Celsius, flattens JSON, and outputs CSV
- Stores raw data (Bronze layer) and cleaned data (Silver layer) in Amazon S3

C. Data Warehouse (Redshift)

- Serverless Redshift configured with appropriate VPC and security groups
- Tables created to store cleaned weather data for analytics

D. Analytics (QuickSight)

- QuickSight used to create dashboards and visualizations querying Redshift data

---

Design Decisions

Service	Rationale
Kinesis	On-demand mode supports unpredictable spikes without shard management
Lambda	Serverless, cost-effective ETL for batch processing
S3	Cost-efficient storage for raw and processed data with lifecycle management
Redshift Serverless	Auto-scaling compute capacity for analytical queries with no cluster maintenance
QuickSight	Native AWS integration for real-time BI dashboards

---

Data Flow

1. **Ingestion:** Python script streams weather data into Kinesis (`weather-stream`) and stores raw JSON in S3 Bronze layer.
2. **Processing:** Lambda processes Kinesis events, cleans and transforms data, stores CSV files in S3 Silver layer.
3. **Warehousing:** Lambda or Glue loads cleaned data into Redshift for analytical queries.
4. **Visualization:** QuickSight queries Redshift to generate interactive dashboards.

Security & Compliance

- **Encryption:** Kinesis streams encrypted with KMS, S3 buckets use server-side encryption (SSE-S3)
- **IAM Roles:** Least-privilege roles assigned to Lambda functions (e.g., `LambdaRoleProject3`)
- **Network Isolation:** Redshift deployed within private VPC subnets secured by security groups

---

Monitoring & Data Quality

- **CloudWatch:** Centralized logging and monitoring for Lambda and Kinesis errors

- **Data Validation:** Lambda functions check for empty or invalid JSON records and verify unit conversions
- **Redshift Query Monitoring:** Performance tracked via Redshift console to optimize queries