**Status Report**

**Individual tasks**

Kishan Byrosu -

* Worked on Data processing of incoming incidents from the Data Streamer and Transmitter.
* Implemented analytical and logical unit in which the conditions on the incidents received from the data streamer and perform data transformation and validation are executed using Flink and AWS Glue.
* Built the flink code, that includes filtering stream1 and storing in a separate stream 2.
* Worked on creating a pipeline to stream the real-world data to perform analytical and logical operations and obtain the timely insights using Amazon Kineses.

Bhuvana Prakash -

* Used data Streamer to create a new pipeline for the data that has been transformed and filtered by the data processor.
* Worked on building an event-based compute system using AWS Lambda to pass the inputs and transmit further to Dashboards, Logs and Alerting Service.
* Implemented the alerting service to send alerts about incident information received from Data handler by using Amazon Simple Notification Service (SNS) to build a robust alerting service which sends notifications two ways, A2A and A2P.
* Built the lambda function for records.

Shiva Sai Sampath Kumar –

* Implemented the rendering of incidents on dashboard with respect to their severity, location, and the timestamp.
* Worked extensively on building visualizations for the data by using AWS Grafana.
* Handled processing, filtering of incidents that had to be pushed into a logging service for auditing and analytical purposes.
* Worked on monitoring CloudWatch to collect and track metrics, which are variables to measure the resources and applications.

**Requirements**

AWS services

Hard Drive - Minimum of 100 GB disk space recommended.

Memory (RAM) - Minimum of 4 GB RAM required beyond the requirements of the operating system and running applications.

**Architecture**

A. Amazon S3:

Amazon S3 is an object storage service that provides manufacturing scalability, data availability,

security, and performance. Users may save and retrieve any quantity of data using Amazon S3 at

any time and from any location.

**Diagram

Description automatically generated**

B. AWS Cloud9- IDE:

Here, we use python in the cloud9 IDE, Python is a computer programming language often used to build websites and software, automate tasks, and conduct data analysis. Here, the data is converted to json payload and given to kinesis stream.

C. Amazon Kinesis:

Amazon Kinesis Data Streams is a real-time data collection and processing service from Amazon. Kinesis Data Streams apps are data-processing applications that may be created.

Kinesis Data Firehose is part of the Kinesis streaming data platform, which also includes Kinesis Data Streams, Kinesis Video Streams, and Amazon Kinesis Data Analytics.

When using Kinesis Data Firehose, the user does not need to develop apps or manage resources. Configure the data

producers to send data to Kinesis Data Firehose, and the data will be automatically transferred to the specified destination. Kinesis Data Firehose may also be used to load data into different AWS

data stores.

D. Apache Flink:

Flink is a scalable data analytics platform and distributed processing engine. Flink may be used to handle massive data streams and give real-time analytical insights about the processed data to your streaming application. Flink is built to work in a variety of cluster setups, with in-memory calculations of any size. For distributed computations over data streams, Flink also offers communication, fault tolerance, and data distribution. Flink applications use unbounded or bounded data sets to process streams of events. Unbounded streams have no fixed termination and are handled indefinitely. Bounded streams have a defined beginning and endpoint and may be handled in batches.

The streaming data is processed through the Flink and given to the lambda functions.

E. Lambda (Python) Functions:

Here based on the python code implementation, the lambda function picks the record with the severity score greater than 3 and passes it to the cloud watch.

F. CloudWatch:

CloudWatch enables users to collect and view monitoring data for AWS infrastructures in a single platform. CloudWatch includes specific features for data collection, monitoring, automated actions, analysis and compliance and security.

G. AWS Grafana- Data Visualization:

Grafana is a web application for interactive visualization and analytics that is open source and cross-platform. When linked to supported data sources, it displays charts, graphs, and alerts on the web for mainly time series data. Grafana is used for reporting and monitoring the alerts in dashboards. Grafana is well suited for working with time-series, application monitoring, and server monitoring.

**Implementation** **details**

This is a countrywide traffic accident dataset, which covers 49 states of the United States. The data is continuously being collected from February 2016, using several data providers, including multiple APIs that provide streaming traffic event data. These APIs broadcast traffic events captured by a variety of entities, such as the US and state departments of transportation, law enforcement agencies, traffic cameras, and traffic sensors within the road-networks. Currently, there are about 1.5 million accident records in this dataset. Check the below descriptions for more detailed information.

The Data Processor processes the incoming incidents, and this part acts as the analytical and logical unit where we execute conditions on the incidents received from the data streamer and perform data transformation and validation. Anytime a new data arrives it will trigger the transformation job using the python Apache link and further send the data to further processing.

Apache Flink is a streaming dataflow engine that you can use to run real-time stream processing on high-throughput data sources. Flink supports event time semantics for out-of-order events, exactly once semantics, backpressure control, and APIs optimized for writing both streaming and batch applications.

Additionally, Flink has connectors for third-party data sources, such as the following:

• Amazon Kinesis Data Streams

• Apache Kafka

Here, based on the flink code, we are filtering stream1 and storing in a separate stream with higher severity scores.

AWS Glue is a serverless data integration service that makes it easy for analytics users to discover, prepare, move, and integrate data from multiple sources. You can use it for analytics, machine learning, and application development. It also includes additional productivity and data ops tooling for authoring, running jobs, and implementing business workflows.

Data Streamer is further used to create a new pipeline for data that has been transformed and filtered by the data processor

Data Handler is a phase where we use the incoming data from the data streams and create an event trigger for each incident that is transmitted through the streamer on-demand.

We used AWS Lambda which is an event-based compute system to consume the inputs and transmit further to Dashboards, Logs and Alerting Service.

AWS Lambda is a compute service that lets you run code without provisioning or managing servers. Lambda runs your code on a high-availability compute infrastructure and performs all the administration of the compute resources, including server and operating system maintenance, capacity provisioning and automatic scaling, and logging. With Lambda, you can run code for virtually any type of application or backend service.

**Prototype**

A picture containing text

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, application

Description automatically generated

**Next Steps**

An Alerting Service should be created that sends the alerts about the incident information received from the Data Handler and is further sent to the subscribers as notifications simultaneously through multimodal communication to the end user.

We have to use Amazon Simple Notification Service (SNS) to build a robust alerting service which sends notifications two ways, A2A and A2P. A2A provides high-throughput, push-based, many-to-many messaging between distributed systems, microservices, and event-driven serverless applications..

We need to use Amazon CloudWatch to monitor your Amazon Web Services (AWS) resources and the applications you run on AWS in real time. You can use CloudWatch to collect and track metrics, which are variables you can measure for your resources and applications.

Data Dashboard must be created to render the incidents on the dashboard.

**Risks**

In real time, wrong calls can lead to greater risks.

Each and every call made will trigger all the units, which results in handling huge data that causes high risk.

Another risk is that it is vulnerable to ddos attack.

The data which is poorly handled can lead to data lost or misleads the system.