**Birla Institute of Technology & Science, Pilani**

**Work Integrated Learning Programmes Division**

**First Semester 2021-2022**

**Assignment 1**

Course No. : DSECL ZC556

Course Title : Stream Processing and Analytics

Nature of Exam : Take Home

Weightage : 10%

No. of Pages = 2

# No. of Questions = 5

Duration : 10 days

**Streaming data solutions on AWS with Amazon Kinesis**

Amazon Kinesis makes it easy to collect, process, and analyze real-time, streaming data so you can get timely insights and react quickly to new information. Amazon Kinesis offers key capabilities to cost-effectively process streaming data at any scale, along with the flexibility to choose the tools that best suit the requirements of your application. With Amazon Kinesis, you can ingest real-time data such as video, audio, application logs, website clickstreams, and IoT telemetry data for machine learning, analytics, and other applications. Amazon Kinesis enables you to process and analyze data as it arrives and respond instantly instead of having to wait until all your data is collected before the processing can begin.

Your group has to read and understand the various streaming data solutions possible with Amazon Kinesis streaming with the help of attached white paper published by Amazon. Based on your grasping of this white paper, address the following questions



Q1. For the given scenario descriptions in the paper, identify and briefly explain the components/services used for - [2.5]

a) Data ingestion and data flow management:

Amazon Kinesis Data Firehose is used for Data Ingestion and Data Flow Management.

• Kinesis Data Firehose enables near-real-time analytics with existing business intelligence tools and dashboards.

• It’s a fully managed serverless service that automatically scales to match the throughput of your data and requires no ongoing administration.

• Kinesis Data Firehose can batch, compress, and encrypt the data before loading, minimizing the amount of storage used at the destination and increasing security.

• It can also transform the source data using AWS Lambda and deliver the transformed data to destinations.

• You configure your data producers to send data to Kinesis Data Firehose, which automatically delivers the data to the destination that you specify.

b) Data enrichment:

AWS Lambda and AWS Glue help in Data enrichment.

1. AWS Lambda: -

• You can subscribe Lambda functions to automatically read batches of records off your Kinesis stream and process them if records are detected on the stream.

• AWS Lambda periodically polls the stream (once per second) for new records and when it detects new records, it invokes the Lambda function passing the new records as parameters.

• The Lambda function is only run when new records are detected.

• You can map a Lambda function to a shared-throughput consumer (standard iterator)

2. AWS Glue: -

• AWS Glue is a fully managed ETL service that you can use to catalog your data, clean it, enrich it, and move it reliably between data stores.

• With AWS Glue, you can significantly reduce the cost, complexity, and time spent creating ETL jobs.

• AWS Glue is serverless, so there is no infrastructure to set up or manage.

• You pay only for the resources consumed while your jobs are running.

• Utilizing AWS Glue, you can create a consumer application with an AWS Glue streaming ETL job.

• This enables you to utilize Apache Spark and other Spark-based modules writing to consume and process your event data.

b) Data processing:

Amazon Kinesis Data Streams is used for Data processing

• Amazon Kinesis Data Streams enables you to build custom, real-time applications using popular stream processing frameworks and load streaming data into many different data stores.

• A Kinesis stream can be configured to continuously receive events from hundreds of thousands of data producers delivered from sources like website clickstreams, IoT sensors, social media feeds and application logs.

• Within milliseconds, data is available to be read and processed by your application.

• When implementing a solution with Kinesis Data Streams, you create custom data processing applications known as Kinesis Data Streams applications.

• A typical Kinesis Data Streams application reads data from a Kinesis stream as data records.

• Data put into Kinesis Data Streams is ensured to be highly available and elastic, and is available in milliseconds.

• You can continuously add various types of data such as clickstreams, application logs, and social media to a Kinesis stream from hundreds of thousands of sources.

• Amazon Kinesis Data Streams is a fully managed streaming data service.

• It manages the infrastructure, storage, networking, and configuration.

c) Data storage :

1.Amazon S3

• Amazon S3 is object storage with a simple web service interface to store and retrieve any amount of data from anywhere on the web.

• It’s designed to deliver 99.999999999% durability, and scale past trillions of objects worldwide.

2.Amazon Redshift

• Amazon Redshift is a fast, fully managed data warehouse that makes it simple and cost-effective to analyze all your data using standard SQL and your existing BI tools.

• It allows you to run complex analytic queries against petabytes of structured data using sophisticated query optimization, columnar storage on high-performance local disks, and massively parallel query running.

3.Amazon DynamoDB

• Amazon DynamoDB is a key-value and document database that delivers single-digit millisecond performance at any scale.

• It's a fully managed, multi-Region, multi-active, durable database with built-in security, backup and restore, and in-memory caching for internet-scale applications.

d) Processed data delivery:

Kinesis Data Firehose helps in Processed data delivery

• As a near-real-time delivery stream, Kinesis Data Firehose buffers incoming data.

• After your delivery stream’s buffering thresholds have been reached, your data is delivered to the destination you’ve configured.

There are some differences in how Kinesis Data Firehose delivers data to each destination: -

1. For data delivery to S3, Kinesis Data Firehose concatenates multiple incoming records based on the buffering configuration of your delivery stream, and then delivers them to Amazon S3 as an S3 object.

2. For data delivery to Amazon Redshift, Kinesis Data Firehose first delivers incoming data to your S3 bucket in the format described earlier. Kinesis Data Firehose then issues an Amazon Redshift COPY command to load the data from your S3 bucket to your Amazon Redshift cluster.

3. For data delivery to Amazon ES, Kinesis Data Firehose buffers incoming records based on the buffering configuration of your delivery stream, and then generates an Elasticsearch bulk request to index multiple records to your Elasticsearch cluster.

Q2. What are the different ways by which data can be written into and read from AWS Kinesis Streams? [1.75]

**Writing**: There are several ways to send data to Kinesis Data Streams, providing flexibility in the designs of your solutions.

• You can write code utilizing one of the AWS SDKs that are supported by multiple popular languages.

• You can use the Amazon Kinesis Agent, a tool for sending data to Kinesis Data Streams.

The Amazon Kinesis Producer Library (KPL) simplifies the producer application development by enabling developers to achieve high write throughput to one or more Kinesis data streams.

The KPL is an easy to use, highly configurable library that you install on your hosts. It acts as an intermediary between your producer application code and the Kinesis Streams API actions.

There are two different operations in the Kinesis Streams API that add data to a stream: PutRecords and PutRecord. The PutRecords operation sends multiple records to your stream per HTTP request while, PutRecord submits one record per HTTP request.

**Reading**: To read and process data from Kinesis streams, you need to create a consumer application. There are varied ways to create consumers for Kinesis Data Streams.

Some of these approaches include using Amazon Kinesis Data Analytics to analyze streaming data using KCL, using AWS Lambda, AWS Glue streaming ETL jobs, and using the Kinesis Data Streams API directly.

• Consumer applications for Kinesis Streams can be developed using the KCL, which helps you consume and process data from Kinesis Streams. The KCL takes care of many of the complex tasks associated with distributed computing such as load

balancing across multiple instances, responding to instance failures, checkpointing processed records, and reacting to resharding. The KCL enables you to focus on the writing record-processing logic.

• You can subscribe Lambda functions to automatically read batches of records off your Kinesis stream and process them if records are detected on the stream. AWS Lambda periodically polls the stream (once per second) for new records and when it detects new records, it invokes the Lambda function passing the new records as parameters. The

Lambda function is only run when new records are detected. You can map a Lambda function to a shared-throughput consumer (standard iterator).You can build a consumer that uses a feature called enhanced fan-out when you require dedicated throughput that you do not want to contend with other consumers that are receiving data from the stream.

This feature enables consumers to receive records from a stream with throughput of up to two MB of data per second per shard.

• For most cases, using Kinesis Data Analytics, KCL, AWS Glue, or AWS Lambda should be used to process data from a stream. However, if you prefer, you can create a consumer application from scratch using the Kinesis Data Streams API. The Kinesis

Data Streams API provides the GetShardIterator and GetRecords methods to retrieve data from a stream.

In this pull model, your code extracts data directly from the shards of the stream.

• AWS Glue processes and writes out data in 100-second windows, by default. This allows data to be processed efficiently, and permits aggregations to be performed on data arriving later than expected. You can configure the window size by adjusting it to accommodate the speed in response vs the accuracy of your aggregation.

Q3. Briefly explain in your understanding of the situation (excluding given scenarios) where you will consider using Kinesis Streams over Firehose. [2]

Kinesis Streams can help the business with streaming data several services will require access to read and process the same stream concurrently. Kinesis data streams is highly customizable and best suited for developers building custom applications or streaming data for specialized needs.

However, requires manual scaling and provisioning. Data typically is made available in a stream for 24 hours, but for an additional cost, users can gain data availability for up to seven days.

Kinesis Streams can be used for application Monitoring on a Massive Scale.

• Netflix uses Amazon Web Services (AWS) for nearly all its computing and storage needs, including databases, analytics, recommendation engines, video transcoding, and more—hundreds of functions that in total use more than 100,000 server instances on AWS.

• This results in an extremely complex and dynamic networking environment where applications are constantly communicating inside AWS and across the Internet. Monitoring and optimizing its network is critical for Netflix to continue improving customer experience, increasing efficiency, and reducing costs.

• In particular, Netflix needed a solution for ingesting, augmenting, and analyzing the multiple terabytes of data its network generates daily in the form of virtual private cloud (VPC) flow logs.

• This would enable Netflix to identify performance-improvement opportunities, such as identifying apps that are communicating across regions and collocating them.

• The company would also be able to increase uptime by quickly detecting and mitigating application downtime.

• Each log record carries information about the communications between two IP addresses. However, in a dynamic environment like the one at Netflix, where an IP address can float between applications from day to day or even minute to minute, IP addresses alone don’t have much meaning.

• “The data sources we had before we took on this initiative were one sided,” says John Bennett, senior software engineer at Netflix. “We’d know an application was connecting to others, but we didn’t know both sides of the conversation and how to optimize those communications or the placement of the applications on the network.”

Netflix set out to establish a new data source that could give it more insight into communication among applications and regions by combining VPC flow logs with application metadata.

Q4. What are the ways by which streaming data can be queried using SQL? [0.75]

Amazon Redshift is a fast, fully managed data warehouse that makes it simple and cost-effective to analyze all your data using standard SQL and your existing BI tools. It allows you to run complex analytic queries against petabytes of structured data using sophisticated query optimization, columnar storage on high-performance local disks, and massively parallel query running.

2. Amazon Kinesis Data Analytics enables you to transform and analyze streaming data and respond to anomalies in real time. It is a serverless service on AWS, which means Kinesis Data Analytics takes care of provisioning, and elastically scales the infrastructure to handle any data throughput. This takes away all the undifferentiated heavy lifting of setting up and managing the streaming infrastructure and enables you to spend more time on writing steaming applications. With Amazon Kinesis Data Analytics, you can interactively query streaming data using multiple options, including Standard SQL, Apache Flink applications in Java, Python and Scala, and build Apache Beam applications using Java to analyze data streams. These options provide you with flexibility of using a specific approach depending on the complexity level of streaming application and source/target support. The following section discusses Kinesis Data Analytics for Flink Applications option.

3. Kinesis Data Analytics Studio is available for customers to interactively query data streams in real time, and easily build and run stream processing applications using SQL, Python, and Scala. Studio notebooks are powered by Apache Zeppelin. Using Studio notebook, you have the ability to develop your Flink Application code in a notebook environment, view results of your code in real time, and visualize it within your notebook. You can create a Studio Notebook powered by Apache Zeppelin and Apache Flink with a single click from Kinesis Data Streams and Amazon MSK console or launch it from Kinesis Data Analytics Console.

Q5. Design and draw a streaming data pipeline using AWS components satisfying the following requirements - [3]

- Ingested streaming data to be stored for 7 days

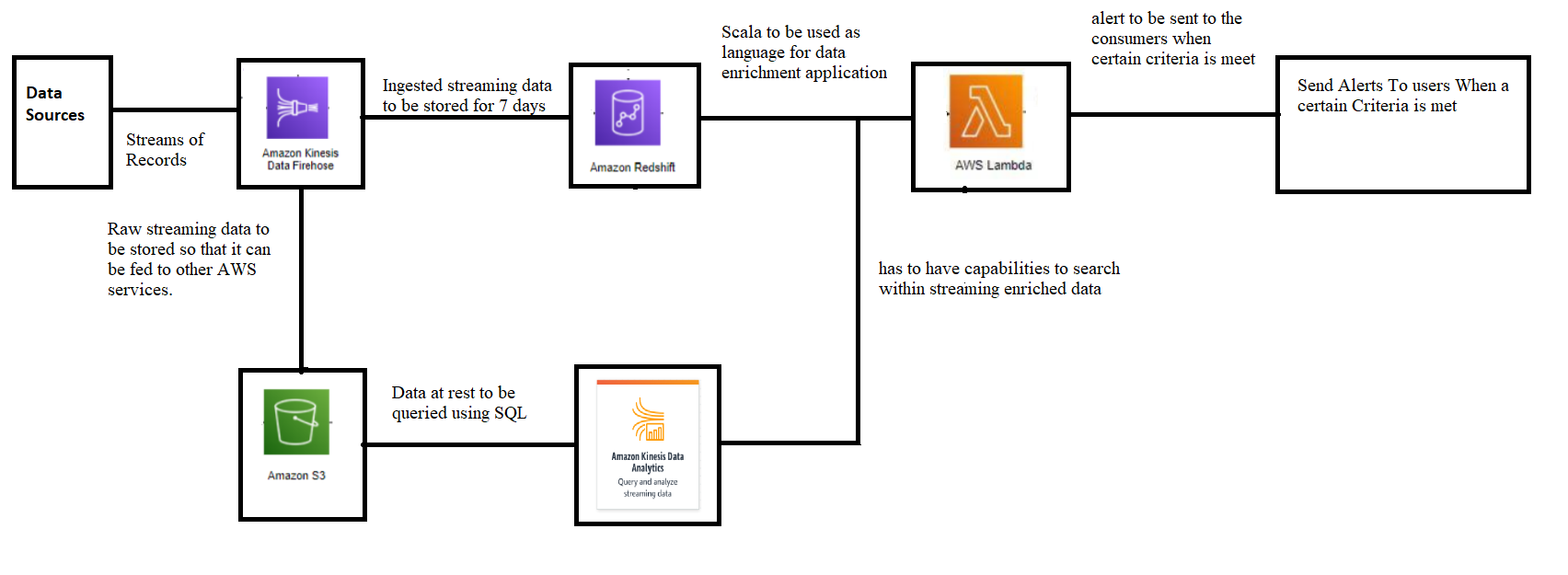
- Raw streaming data to be stored so that it can be fed to other AWS services

- Scala to be used as language for data enrichment application

- Data at rest to be queried using SQL

- has to have capabilities to search within streaming enriched data

- alert to be sent to the consumers when certain criteria is meet



\*\*\*\*\*\*\*\*\*\*\*