AWS Kinesis

1. Kinesis

- Kinesis is a platform for collecting, processing and delivering streaming data on AWS.
- Using Kinesis, we can build custom streaming applications for specific purpose also.
- Supports data sources to produce streaming data and deliver data records simultaneously in small size (usually in KBs)

1.1. Streaming Data :

- Data produced by thousands of data sources continuously.
- Eg:
    - Log files generated from web servers,
    - ecommerce purchases,
    - in-game player activity,
    - information from social networks,
    - financial trading floors, etc.

1.2. Services :

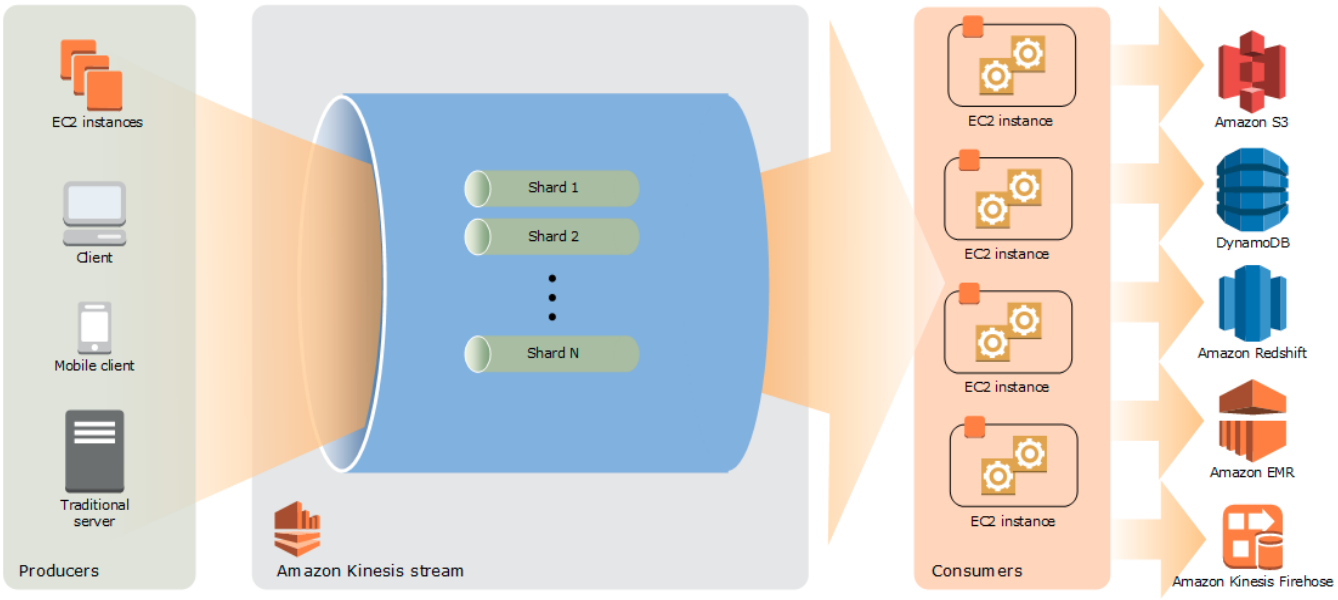
- Kinesis provides three services :
    I. Streams : Streams collect and process large streams of data records using providers and consumers in real-time.
    II. Firehose : Firehose directly delivers real-time streaming data to other AWS services.
    III. Analytics : Analytics process and analyze real-time streaming data with standard SQL.

2. Kinesis Streams

- It collect and process large streams of data records in real time.
- Support rapid and continuous data intake and aggregation.
- Kinesis applications are data-processing applications or consumers
    - They read data from Kinesis stream as data records.
    - These applications can use the Kinesis Client Library (KCL) and they can run on Amazon EC2 instance.
- Use cases :
    - Accelerated log and data feed intake and processing
    - Real-time metrics and reporting
    - Real-time data analytics
    - Complex stream processing
- Provides durability and elasticity :
    - Put-to-get delay is typically less than 1 second.
    - Enables scaling the stream up or down.
- Multiple applications can consume data from a stream.

3. Kinesis Data Streams High-level architecture

- It collect and process large streams of data records in real time.



3.1. Shards

- Streams are made of shards, and used as base throughput unit of a stream.
- Write Operation (Producer) : Each shard support 1000 records/sec or up to maximum rate of 1MB/sec.
- Read Operation (Consumer) : Each shard support up to 5 transactions/sec or up to maximum read rate of 2MB/sec.
- PUT data call will be rejected with ProvisionThroughputExceeded exception when throughput limits are exceeded.

3.2. Retention Period

- By default, records of a stream are accessible upto 24 hours from the time they are added to a stream.
- You can extend it upto 7 days.

3.3. Records

- A record is a unit of data stored in a stream.
- Stream is an ordered sequence of data records.
- Record is composed of a sequence no., partition key and data blob.

3.3.1 : Data Blob

- Data blob is the original data from a producer with maximum size of 1MB.

3.3.2 : Partition key

- Partition key helps to identify and route records to different shards.

3.3.3 : Sequence Number

- A sequence no. is a unique identifier for each record.
- It is like a primary key (from Database context) for each record.