**Tools Used:**

Kinesis DataStream and Firehose for streaming, Lambda for processing, Dynamo DB for define rules and application level configuration and S3 for storage

**Requirement:**

Kinesis used for processing high volume of data. Kinesis can continuously capture gigabytes of data per second from hundreds of thousands of sources such as website clickstreams, database event streams, financial transactions, social media feeds, IT logs, and location-tracking events.

Amazon S3 enables Storage (After applying filter from Lambda function) and data backup services for others.

**Why S3:**

1. Storing any files
2. Can easily extend your storage with Zero downtime.
3. High Availability
4. Low Network Latency.
5. Easy to Use
6. Quick data access and very Cheap

Technical overview:

Lambda service is apply user defined rules which defined in Dynamo DB to Firehose streaming data and store the data into two s3 buckets

1. Result – required result data which we applied filter condition in Lambda.
2. Datalake – All incoming data without filter – further data processing.

**Process Flow:**

Kinesis DataStream and Firehose -> Lambda -> S3 -> Reporting

**Lessons Learnt:**

Pros:

1. High data security.
2. Store and retrieve any amount of data , at any time, from anywhere on the web
3. Intentionally built with a minimal feature set that focuses on simplicity and robustness
4. Easily define public and private access rules
5. Highly scalable, reliable, fast, inexpensive data storage that Amazon uses to run its own global network of web sites.
6. Data consistency model
7. Version maintaining.
8. Rest API support.
9. Can easily Monitor S3 bucket
10. Storage metrics

**Cons:**

1. Bucket name should be unique across amazon s3 level.