

## **Scripts Execution**

### **Explanation of the solution to the batch layer problem**

- In this project our goal is to detect the credit card fraud in the real time streaming transaction.
- This whole project is divided into two part. In this document we will focus on first half of the project i.e
  - Task 1: Load the transactions history data (card\_transactions.csv) in a NoSQL database.
  - Task 2: Ingest the relevant data from AWS RDS to Hadoop.
  - Task 3: Create a look-up table with columns specified

earlier in the problem statement.

- Task 4: After creating the table, you need to load the relevant data in the lookup table.
- In order to perform the aforementioned tasks, I have created a single node EMR cluster with the below configuration.

Clone

Terminate

AWS CLI export

Cluster: Capstone   Terminated   Terminated by user request

Summary

Application user interfaces

Monitoring

Hardware

Configurations

Events

Steps

Bootstrap actions

Summary

ID: j-2KNCH7WWGHZXX

Creation date: 2022-05-13 23:33 (UTC+5:30)

End date: 2022-05-14 01:09 (UTC+5:30)

Elapsed time: 1 hour, 36 minutes

After last step completes: Cluster waits

Termination protection: Off

Tags: --

Master public DNS: ec2-54-167-14-71.compute-1.amazonaws.com [🔗](#)

[Connect to the Master Node Using SSH](#)

Configuration details

Release label: emr-5.30.1

Hadoop distribution: Amazon 2.8.5

Applications: Spark 2.4.5, Zeppelin 0.8.2, Hive 2.3.6, Hue 4.6.0, HBase 1.4.13, Sqoop 1.4.7, Livy 0.7.0

Log URI: --

EMRFS consistent view: Disabled

Custom AMI ID: --

Application user interfaces

Persistent user interfaces [🔗](#): [Spark history server](#), [YARN timeline server](#), [Tez UI](#)

On-cluster user -- interfaces [🔗](#):

Network and hardware

Availability zone: us-east-1a

Subnet ID: [subnet-83d32ecf](#) [🔗](#)

Master: Terminated 1 m4.xlarge

Core: --

Task: --

Cluster scaling: Not enabled

Auto-termination: Not enabled

Security and access

Key name: aditya\_key\_pair

EC2 instance profile: EMR\_EC2\_DefaultRole

EMR role: EMR\_DefaultRole

Visible to all users: All [Change](#)

Security groups for Master: [sg-0b899d3af4ea60b89](#) [🔗](#) (ElasticMapReduce-

- List of documents submitted in the zip: • Load NoSQL.pdf - Task 1
- Sqoop Data Ingestion.pdf - Task 2 • Create NoSQL.pdf - Task 3
- PreAnalysis.pdf - Task 4

- **TASK 1:**

- To load the data from csv file to Hbase (NoSQL database). I have uploaded the csv file to S3 bucket. Loc: s3://capstone-aditya/input/card\_transactions.csv
- Then from S3, I have imported the file into hdfs using distcp command.
- After that I have opened the hive shell and created a table to hold the data from csv file and then created hive-hbase integrated table so that all data must be reflected in Hbase also.
- After importing data to hive-hbase integrated table, I used random UUID for

row key creation.

- Finally verified all the records imported successfully into table.
- All the commands and screenshot attached in the **Load NoSQL.pdf**.

- **TASK 2:**

- In order to ingest data from AWS RDS to hdfs, I have used sqoop to import the data.
- After importing the data into hdfs, I used hive to create and load the imported data into hive table.
- Finally verified all the records imported successfully into

table.

- All the commands and screenshot attached in the  
**Sqoop Data Ingestion.pdf.**
  - **TASK 3:**
    - In order to create Lookup table I have used hive-base integrated table.
    - All the commands and screenshot attached in the  
**Create NoSQL.pdf.**
    - **TASK 4:**
- To load the data into Lookup table first we need to calculate some fields.
- I have created two intermediate table in hive, first one to hold the last 10 transactions and second one to calculate the UCL of last 10 transactions.

- Finally load the data into lookup table.
- Finally verified all the records imported successfully into table.
- All the commands and screenshot attached in the  
**PreAnalysis.pdf.**

- All the command and screenshot are consolidated in

### **Scripts Execution.pdf.**

- This is the logic behind my way to complete task (till task 4).