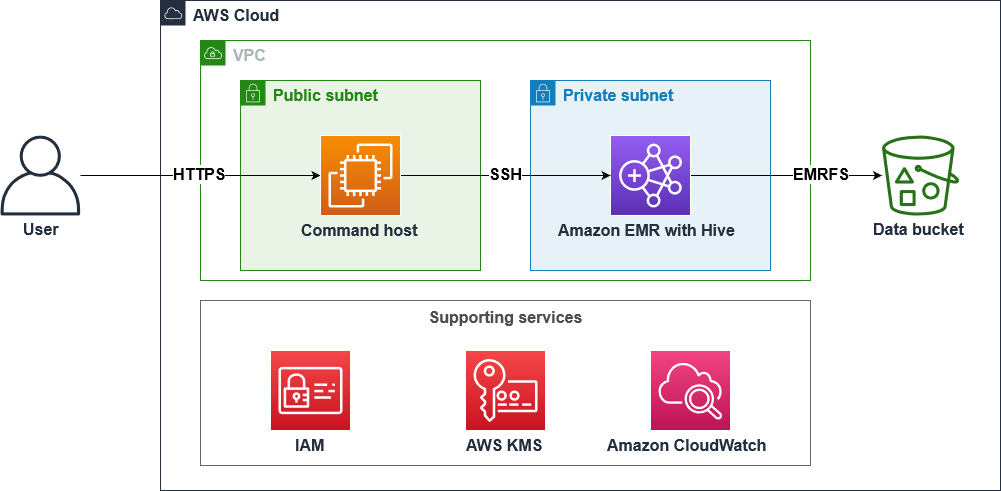
# Lab 2 - Batch Data Processing using Amazon EMR with Hive

creating a table using Apache Hive, and running queries on this data using HiveQL.

1. Connect to an EMR cluster with SSH
2. Create a table using Apache Hive and load batch data from Amazon S3
3. Run queries using HiveQL



1. Amazon EMR: A service that simplifies the deployment and management of big data frameworks such as Hive on scalable clusters of Amazon EC2 instances.
2. Apache Hive: A data warehouse infrastructure for providing data summarization, query, and analysis capabilities

Task 1 - precreated -

1. S3 file
2. EMR cluster

**Task 2: Connect to the EMR leader node using Session Manager**

2.1 use CLI link to connect to EMR cluster

2.2 connect to cluster

*# Get EMR cluster ID and export to the Environment.*

*export ID=$(aws emr list-clusters | jq '.Clusters[0].Id' | tr -d '"')*

*# Use the ID to get the PublicDNS name of the EMR cluster*

*# and export to the Environment.*

*export HOST=$(aws emr describe-cluster --cluster-id $ID | jq '.Cluster.MasterPublicDnsName' | tr -d '"')*

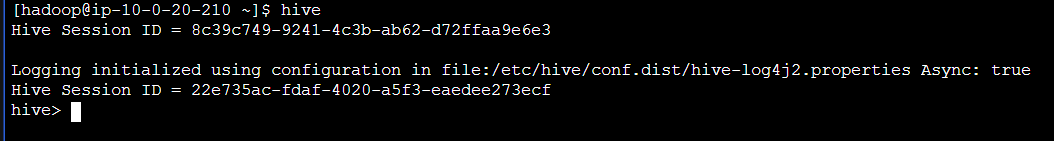
*# SSH to the EMR cluster*

*ssh -i ~/EMRKey.pem hadoop@$HOST*

**Task 3: Access your Amazon S3 data using Amazon S3 Select with Hive**

3.1 Connect to hive using

hive



3.2

Amazon S3 Select is a feature that allows you to retrieve a subset of data from an S3 object using SQL-like queries, reducing the amount of data transferred and improving performance for data processing tasks.

3.3 create table

*CREATE TABLE stockprice (*

*`Trade\_Date` string,*

*`Ticker` string,*

*`High` double,*

*`Low` double,*

*`Open` double,*

*`Close` double,*

*`Volume` double,*

*`Adj\_Close` double*

*)*

*ROW FORMAT DELIMITED FIELDS TERMINATED BY ','*

*STORED AS*

*INPUTFORMAT*

*'com.amazonaws.emr.s3select.hive.S3SelectableTextInputFormat'*

*OUTPUTFORMAT*

*'org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat'*

*LOCATION 's3://<dataBucket>/data/'*

*TBLPROPERTIES (*

*"s3select.format" = "csv",*

*"s3select.headerInfo" = "ignore",*

*"skip.header.line.count"="1"*

*);*



- ignore the header column

Here, we created table from the csv file in S3

3.4 check working

*SET s3select.filter=true;*

*SELECT \* FROM stockprice WHERE `Trade\_Date` LIKE '2020-01-03' ORDER BY `Ticker`;*

