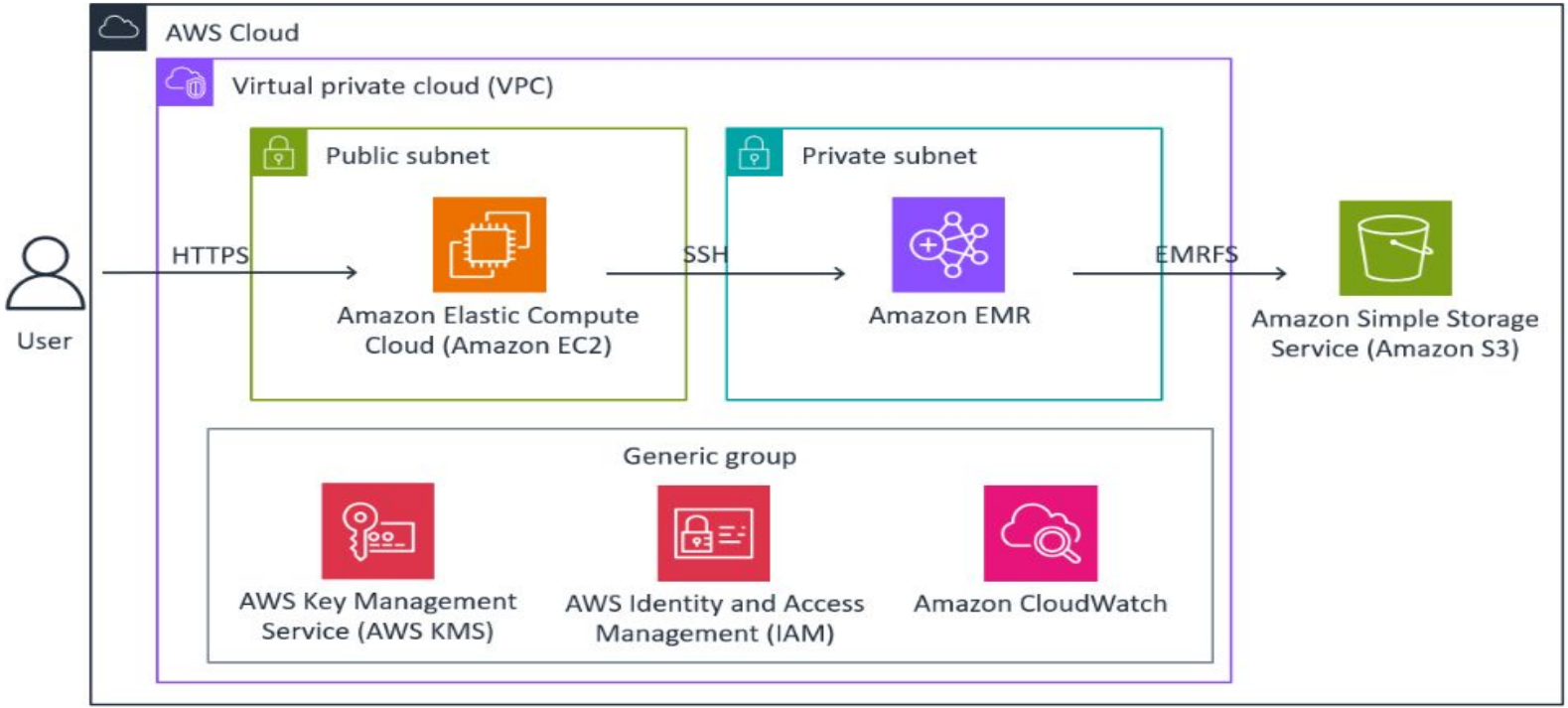
Load the sample data in Amazon Simple Storage Service (Amazon S3) and connect to the Amazon EMR cluster. Next, you create an Apache Hive table, load the data from Amazon S3, and run queries using HiveQL



Objectives

By the end of this lab, you should be able to do the following:

- Review how Amazon EMR and Apache Hive can be used together to ingest and query data
- Identify key components of an EMR cluster.
- Connect to an EMR cluster with SSH.
- Create a table using Apache Hive and load batch data from Amazon S3.
- Run queries using HiveQL

Trade_Date	Ticker	High	Low	Open	Close	Volume	Adj_Close
2020-01-02	aapl	75.1500015258789	73.7975006103515 6	74.0599975585937 5	75.0875015258789	135480400. 0	74.207466125488 8
2020-01-02	sq	64.0500030517578 1	62.9500007629394 5	62.9900016784668	63.8300018310546 9	5264700	63.830001831054 9
2020-01-02	amzn	1898.01000976562 5	1864.15002441406 25	1875.0	1898.01000976562 5	4029000	1898.0100097656 5
2020-01-02	ge	11.9600000381469 73	11.2299995422363 28	11.2299995422363 28	11.9300003051757 81	87421800.0	11.861019134521 84
2020-01-02	m	17.2700004577636 72	16.3899993896484 38	17.1800003051757 8	16.5200004577636 72	26388100.0	15.861986160278 2
2020-01-02	tsla	86.1399993896484 4	84.3420028686523 4	84.9000015258789	86.052001953125	47660500.0	86.052001953125
2020-01-02	msft	160.729995727539 06	158.330001831054 7	158.779998779296 88	160.619995117187 5	22622100.0	158.205764770507 8

Connect to the EMR leader node using Session Manager

use Session Manager, a capability of AWS Systems Manager, to connect to your EMR leader node

```
# 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
```

Access your Amazon S3 data with Hive

In this task, you start an interactive Hive session with the leader node. Then, you create the Hive table from the CSV file on S3.

Note: For the purpose of this lab, you use the Open CSV SerDe hive driver to read the CSV files on Amazon S3 and create a table in Amazon EMR.

10. Command: To create a logging directory that is used by Hive, copy and paste the following commands into the SSH window:

```
sudo chown hadoop -R /var/log/hive

  mkdir /var/log/hive/user/hadoop
```

The *hive.log* file is stored in this directory, which contains logs related to Hive.

11. Command: To connect to the Hive CLI, paste the following command into the SSH window:

```
  Hive
```

You should be presented with a *hive*> prompt. It might take about 10 seconds to appear.

To create a table, copy and paste the following Hive statement in a text editor:

Warning: Replace *<dataBucket>* with the dataBucket value shown to the left of these instructions.

```
CREATE TABLE stockprice  (

`Trade_Date` string,

`Ticker` string,

`High` double,

`Low` double,

`Open` double,

`Close` double,

`Volume` double,

`Adj_Close` double

)ROW FORMAT SERDE 'org.apache.hadoop.hive.serde2.OpenCSVSerdeWITH SERDEPROPERTIES (

  "separatorChar" = ",",

  "quoteChar"     = "`",

  "escapeChar"    = "\\"

)

STORED AS TEXTFILE

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

    TBLPROPERTIES ("skip.header.line.count"="1");
```

```
CREATE TABLE movies  (
`year` int,
`title` string,
`directors_0` string,
`rating` string,
`genres_0` string,
`genres_1` string,
`rank` string,
`running_time_secs` string,
`actors_0` string,
`actors_1` string,
`actors_2` string,
`directors_1` string,
`directors_2` string
)
ROW FORMAT SERDE 'org.apache.hadoop.hive.serde2.OpenCSVSerde'
WITH SERDEPROPERTIES (
  "separatorChar" = ",",
  "quoteChar"     = "`",
  "escapeChar"    = "\\"
```