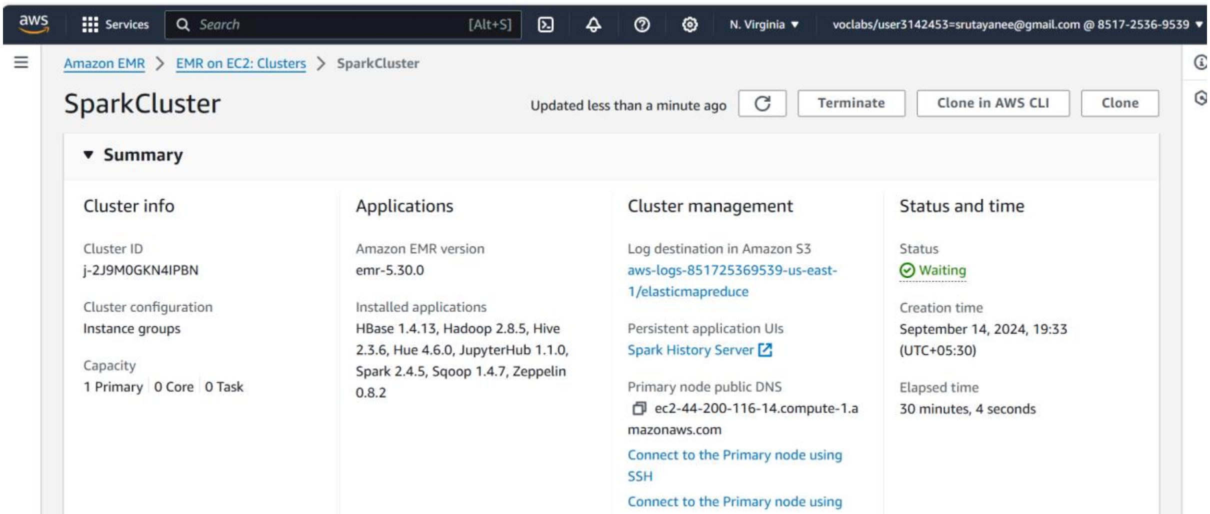
Final Submission

(Explanation of the solution to the streaming layer problem in detail should be provided properly in a document)

Step 1: In order to achieve the requirements for final submission, we have used an EMR cluster with Hadoop, Sqoop, Hive, HBase, Spark and RDS with EBS volume size as 20 GB. We have also updated the YARN configurations as per expectation.


aws
Services
Search [Alt+S]
N. Virginia
voclabs/user3142453=srutayanee@gmail.com @ 8517-2536-9539
Amazon EMR > EMR on EC2: Clusters > SparkCluster
Updated less than a minute ago
Terminate
Clone in AWS CLI
Clone

SparkCluster

▼ Summary

Cluster info

Cluster ID
j-2J9M0GKN4IPBN

Cluster configuration
Instance groups

Capacity
1 Primary | 0 Core | 0 Task

Applications

Amazon EMR version
emr-5.30.0

Installed applications
HBase 1.4.13, Hadoop 2.8.5, Hive 2.3.6, Hue 4.6.0, JupyterHub 1.1.0, Spark 2.4.5, Sqoop 1.4.7, Zeppelin 0.8.2

Cluster management

Log destination in Amazon S3
aws-logs-851725369539-us-east-1/elasticmapreduce

Persistent application UIs
Spark History Server

Primary node public DNS
ec2-44-200-116-14.compute-1.amazonaws.com

Connect to the Primary node using SSH

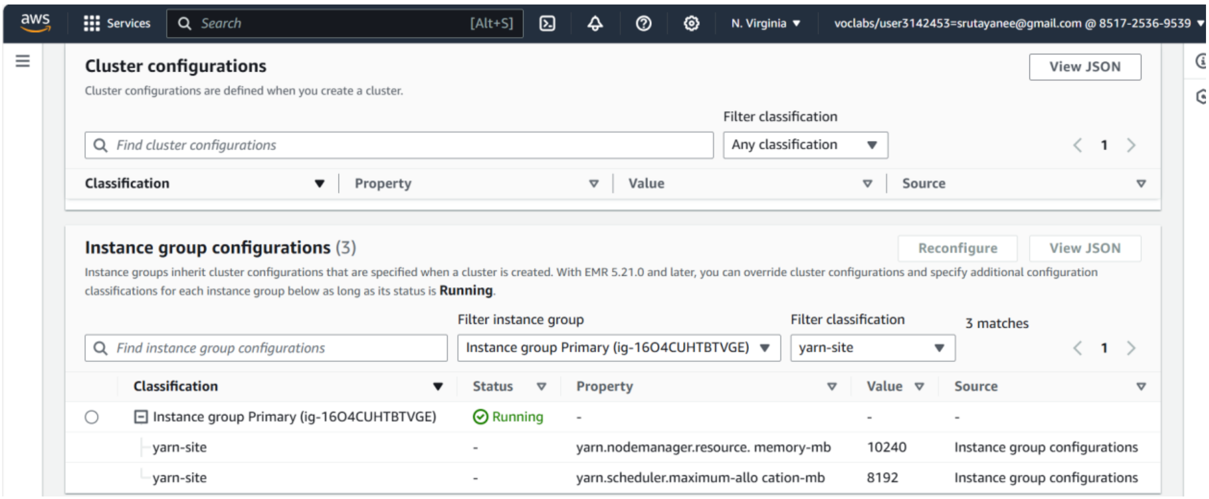
Connect to the Primary node using

Status and time

Status
Waiting

Creation time
September 14, 2024, 19:33 (UTC+05:30)

Elapsed time
30 minutes, 4 seconds


aws
Services
Search [Alt+S]
N. Virginia
voclabs/user3142453=srutayanee@gmail.com @ 8517-2536-9539
View JSON
Find cluster configurations
Reconfigure
View JSON
Find instance group configurations

Cluster configurations

Cluster configurations are defined when you create a cluster.

Filter classification: Any classification

Classification	Property	Value	Source

Instance group configurations (3)

Instance groups inherit cluster configurations that are specified when a cluster is created. With EMR 5.21.0 and later, you can override cluster configurations and specify additional configuration classifications for each instance group below as long as its status is **Running**.

Filter instance group: Instance group Primary (ig-16O4CUHTBTVGE)

Filter classification: yarn-site

3 matches

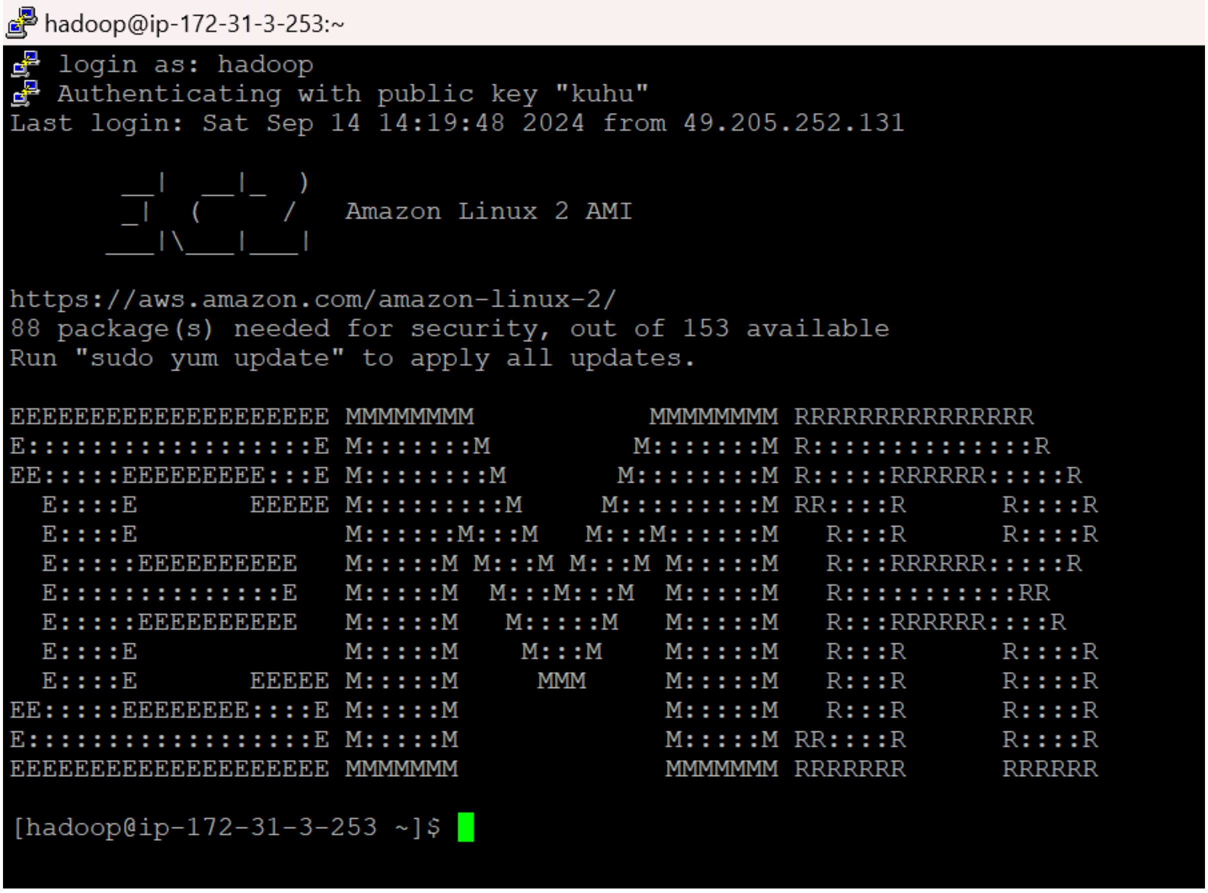
Classification	Status	Property	Value	Source
Instance group Primary (ig-16O4CUHTBTVGE)	Running	-	-	-
yarn-site	-	yarn.nodemanager.resource. memory-mb	10240	Instance group configurations
yarn-site	-	yarn.scheduler.maximum-allo cation-mb	8192	Instance group configurations

Step 2: Below are the tasks we are supposed to perform as per final submission:

- **Task 5**: Create a streaming data processing framework that ingests real-time POS transaction data from Kafka. The transaction data is then validated based on the three rules' parameters (stored in the NoSQL database) discussed previously.
- **Task 6**: Update the transactions data along with the status (fraud/genuine) in the card_transactions table.

- **Task 7**: Store the ‘postcode’ and ‘transaction_dt’ of the current transaction in the look-up table in the NoSQL database if the transaction was classified as genuine.

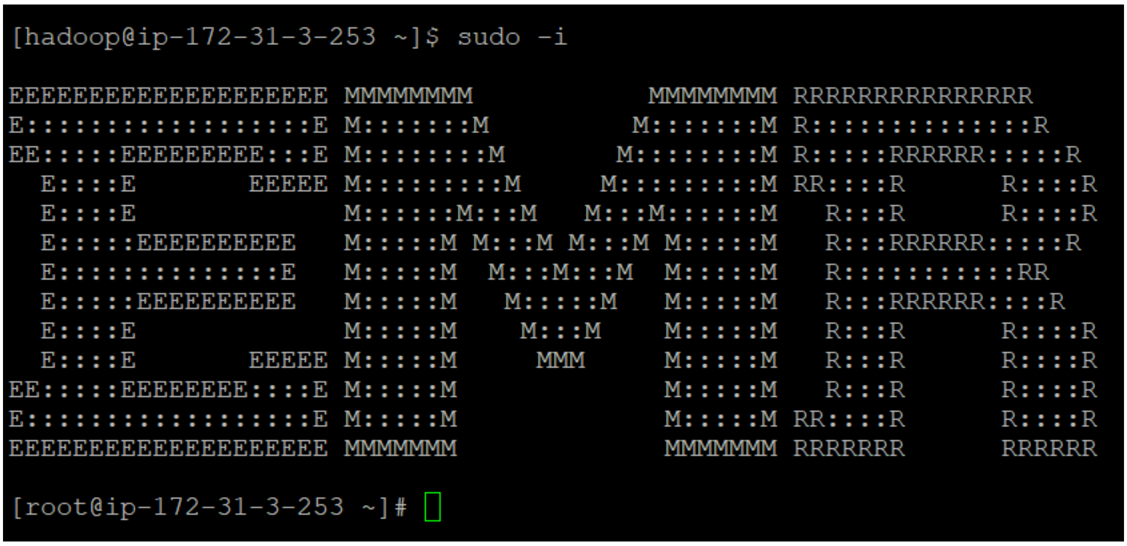
Step 3: We will enter into our EMR cluster as a Hadoop user for which we will give **hadoop** in login as.



hadoop@ip-172-31-3-253:~

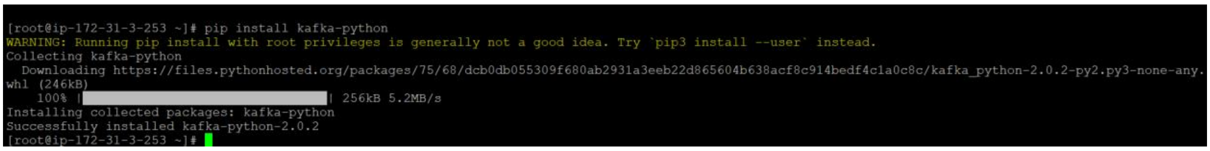
```
login as: hadoop
Authenticating with public key "kuhu"
Last login: Sat Sep 14 14:19:48 2024 from 49.205.252.131

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
88 package(s) needed for security, out of 153 available
Run "sudo yum update" to apply all updates.

EEEEEEEEEEEEEEEEEEEE MMMMMMMM           MMMMMMMM RRRRRRRRRRRRRRR
E::::::::::::::::::E M:::::::M         M:::::::M R::::::::::::::R
EE:::::EEEEEEEEE:::E M::::::::M       M::::::::M R:::::RRRRRR:::::R
  E::::E       EEEEE M:::::::::M     M:::::::::M RR::::R      R::::R
  E::::E             M::::::M:::M   M:::M::::::M   R:::R      R::::R
  E:::::EEEEEEEEEE   M:::::M M:::M M:::M M:::::M   R:::RRRRRR:::::R
  E::::::::::::::E   M:::::M  M:::M:::M  M:::::M   R:::::::::::RR
  E:::::EEEEEEEEEE   M:::::M   M:::::M   M:::::M   R:::RRRRRR::::R
  E::::E             M:::::M    M:::M    M:::::M   R:::R      R::::R
  E::::E       EEEEE M:::::M     MMM     M:::::M   R:::R      R::::R
EE:::::EEEEEEEE::::E M:::::M             M:::::M   R:::R      R::::R
E::::::::::::::::::E M:::::M             M:::::M RR::::R      R::::R
EEEEEEEEEEEEEEEEEEEE MMMMMMM             MMMMMMM RRRRRRR      RRRRRR

[hadoop@ip-172-31-3-253 ~]$
```

Step 4: Next we will switch to root user for which we will enter the command **sudo -i**.



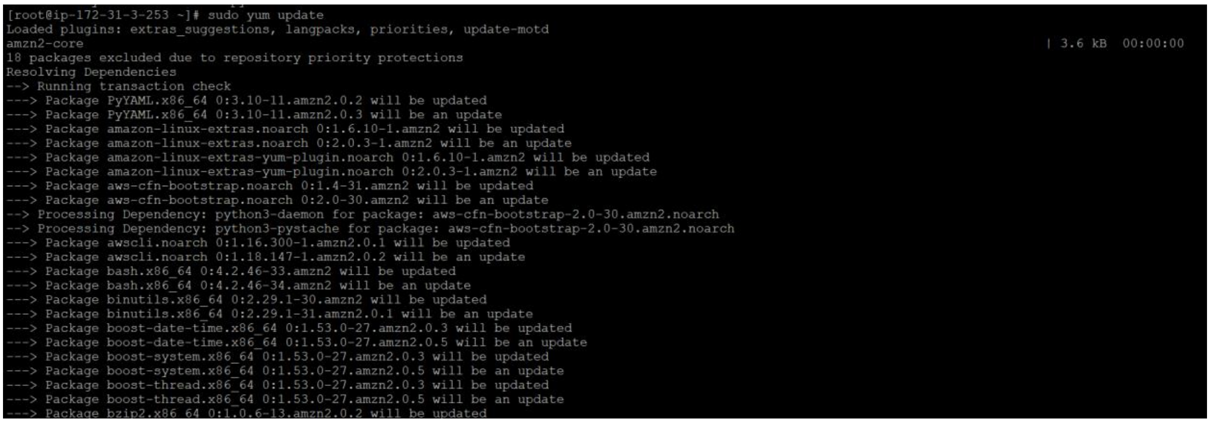
```
[hadoop@ip-172-31-3-253 ~]$ sudo -i

EEEEEEEEEEEEEEEEEEEE MMMMMMMM           MMMMMMMM RRRRRRRRRRRRRRR
E::::::::::::::::::E M:::::::M         M:::::::M R::::::::::::::R
EE:::::EEEEEEEEE:::E M::::::::M       M::::::::M R:::::RRRRRR:::::R
  E::::E       EEEEE M:::::::::M     M:::::::::M RR::::R      R::::R
  E::::E             M::::::M:::M   M:::M::::::M   R:::R      R::::R
  E:::::EEEEEEEEEE   M:::::M M:::M M:::M M:::::M   R:::RRRRRR:::::R
  E::::::::::::::E   M:::::M  M:::M:::M  M:::::M   R:::::::::::RR
  E:::::EEEEEEEEEE   M:::::M   M:::::M   M:::::M   R:::RRRRRR::::R
  E::::E             M:::::M    M:::M    M:::::M   R:::R      R::::R
  E::::E       EEEEE M:::::M     MMM     M:::::M   R:::R      R::::R
EE:::::EEEEEEEE::::E M:::::M             M:::::M   R:::R      R::::R
E::::::::::::::::::E M:::::M             M:::::M RR::::R      R::::R
EEEEEEEEEEEEEEEEEEEE MMMMMMM             MMMMMMM RRRRRRR      RRRRRR

[root@ip-172-31-3-253 ~]#
```

Step 5: Next we will run the command **pip install kafka-python**



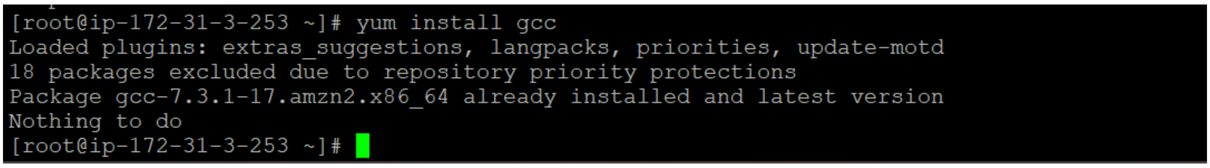
```
[root@ip-172-31-3-253 ~]# pip install kafka-python
WARNING: Running pip install with root privileges is generally not a good idea. Try `pip3 install --user` instead.
Collecting kafka-python
  Downloading https://files.pythonhosted.org/packages/75/68/dcb0db055309f680ab2931a3eeb22d865604b638acf8c914bedf4c1a0c8c/kafka_python-2.0.2-py2.py3-none-any.
whl (246kB)
    100% |████████████████████████████████| 256kB 5.2MB/s
Installing collected packages: kafka-python
Successfully installed kafka-python-2.0.2
[root@ip-172-31-3-253 ~]# █
```

Step 6: Next we ran the command **sudo yum update**



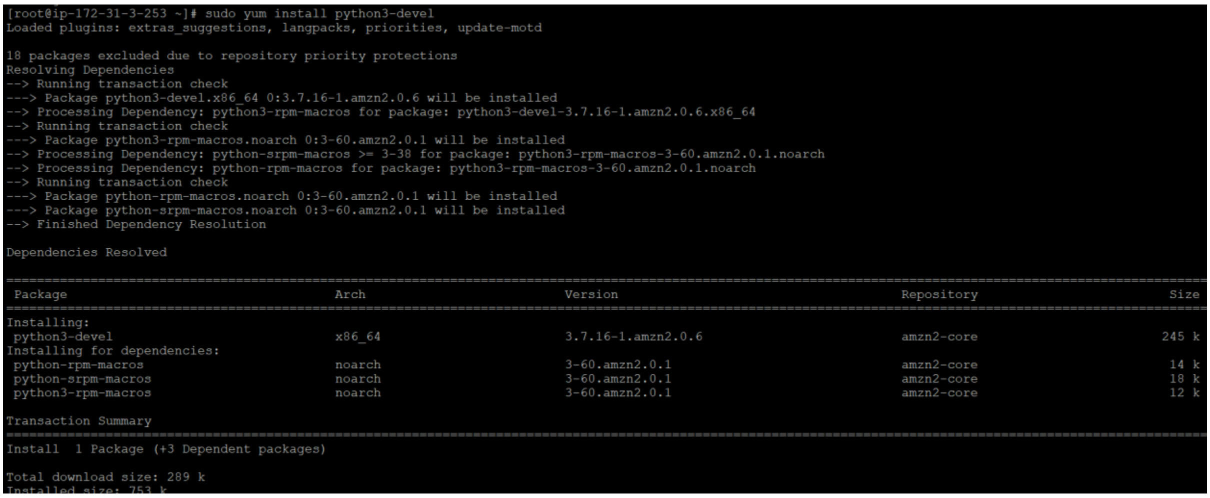
```
[root@ip-172-31-3-253 ~]# sudo yum update
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core                                                                                                          | 3.6 kB  00:00:00
18 packages excluded due to repository priority protections
Resolving Dependencies
--> Running transaction check
---> Package PyYAML.x86_64 0:3.10-11.amzn2.0.2 will be updated
---> Package PyYAML.x86_64 0:3.10-11.amzn2.0.3 will be an update
---> Package amazon-linux-extras.noarch 0:1.6.10-1.amzn2 will be updated
---> Package amazon-linux-extras.noarch 0:2.0.3-1.amzn2 will be an update
---> Package amazon-linux-extras-yum-plugin.noarch 0:1.6.10-1.amzn2 will be updated
---> Package amazon-linux-extras-yum-plugin.noarch 0:2.0.3-1.amzn2 will be an update
---> Package aws-cfn-bootstrap.noarch 0:1.4-31.amzn2 will be updated
---> Package aws-cfn-bootstrap.noarch 0:2.0-30.amzn2 will be an update
--> Processing Dependency: python3-daemon for package: aws-cfn-bootstrap-2.0-30.amzn2.noarch
--> Processing Dependency: python3-pystache for package: aws-cfn-bootstrap-2.0-30.amzn2.noarch
---> Package awscli.noarch 0:1.16.300-1.amzn2.0.1 will be updated
---> Package awscli.noarch 0:1.18.147-1.amzn2.0.2 will be an update
---> Package bash.x86_64 0:4.2.46-33.amzn2 will be updated
---> Package bash.x86_64 0:4.2.46-34.amzn2 will be an update
---> Package binutils.x86_64 0:2.29.1-30.amzn2 will be updated
---> Package binutils.x86_64 0:2.29.1-31.amzn2.0.1 will be an update
---> Package boost-date-time.x86_64 0:1.53.0-27.amzn2.0.3 will be updated
---> Package boost-date-time.x86_64 0:1.53.0-27.amzn2.0.5 will be an update
---> Package boost-system.x86_64 0:1.53.0-27.amzn2.0.3 will be updated
---> Package boost-system.x86_64 0:1.53.0-27.amzn2.0.5 will be an update
---> Package boost-thread.x86_64 0:1.53.0-27.amzn2.0.3 will be updated
---> Package boost-thread.x86_64 0:1.53.0-27.amzn2.0.5 will be an update
---> Package bzip2.x86_64 0:1.0.6-13.amzn2.0.2 will be updated
```

Step 7: Next we ran the command **yum install gcc**



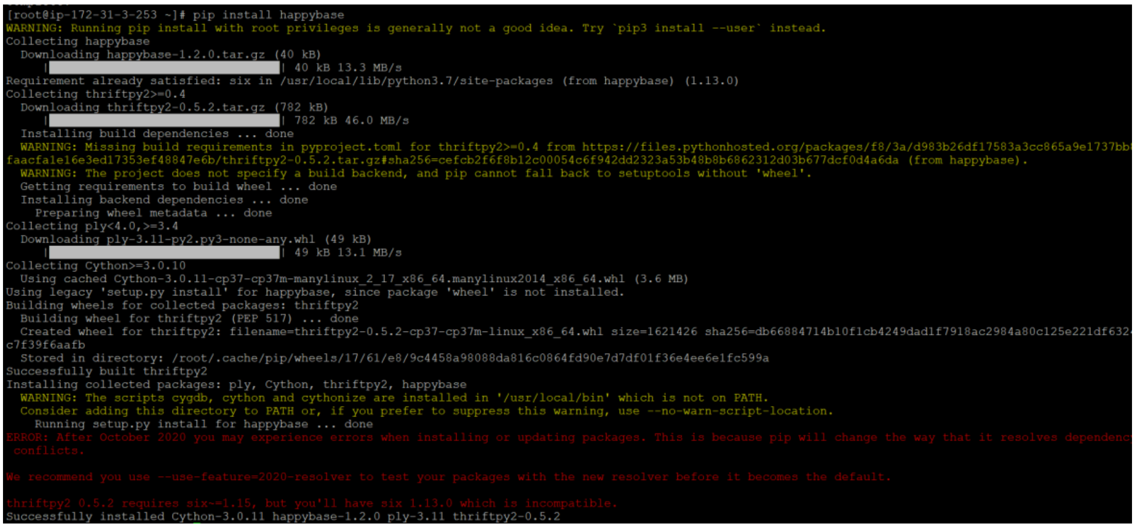
```
[root@ip-172-31-3-253 ~]# yum install gcc
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
18 packages excluded due to repository priority protections
Package gcc-7.3.1-17.amzn2.x86_64 already installed and latest version
Nothing to do
[root@ip-172-31-3-253 ~]# █
```

Step 8: Next we ran the command **sudo yum install python3-devel**



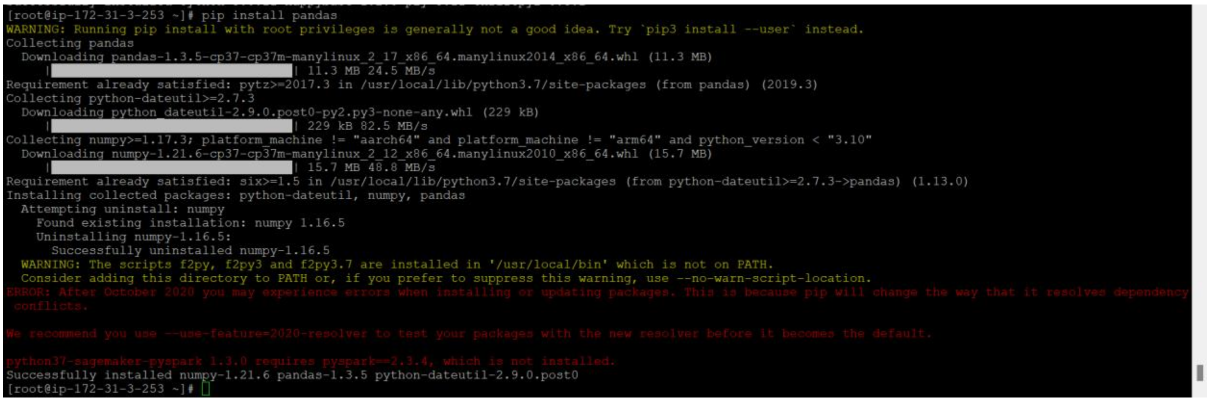
```
[root@ip-172-31-3-253 ~]# sudo yum install python3-devel
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd

18 packages excluded due to repository priority protections
Resolving Dependencies
--> Running transaction check
---> Package python3-devel.x86_64 0:3.7.16-1.amzn2.0.6 will be installed
--> Processing Dependency: python3-rpm-macros for package: python3-devel-3.7.16-1.amzn2.0.6.x86_64
--> Running transaction check
---> Package python3-rpm-macros.noarch 0:3-60.amzn2.0.1 will be installed
--> Processing Dependency: python-srpm-macros >= 3-38 for package: python3-rpm-macros-3-60.amzn2.0.1.noarch
--> Processing Dependency: python-rpm-macros for package: python3-rpm-macros-3-60.amzn2.0.1.noarch
--> Running transaction check
---> Package python-rpm-macros.noarch 0:3-60.amzn2.0.1 will be installed
---> Package python-srpm-macros.noarch 0:3-60.amzn2.0.1 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

==========================================================================================================================================
 Package                                   Arch                          Version                                   Repository          Size
==========================================================================================================================================
Installing:
 python3-devel                             x86_64                        3.7.16-1.amzn2.0.6                        amzn2-core         245 k
Installing for dependencies:
 python-rpm-macros                         noarch                        3-60.amzn2.0.1                            amzn2-core          14 k
 python-srpm-macros                        noarch                        3-60.amzn2.0.1                            amzn2-core          18 k
 python3-rpm-macros                        noarch                        3-60.amzn2.0.1                            amzn2-core          12 k

Transaction Summary
==========================================================================================================================================
Install  1 Package (+3 Dependent packages)

Total download size: 289 k
Installed size: 753 k
```

Step 9: Next for installing happybase, we ran the command **pip install happybase**



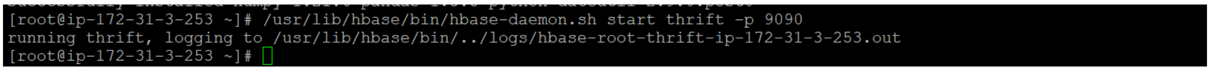
```
[root@ip-172-31-3-253 ~]# pip install happybase
WARNING: Running pip install with root privileges is generally not a good idea. Try `pip3 install --user` instead.
Collecting happybase
  Downloading happybase-1.2.0.tar.gz (40 kB)
     |████████████████████████████████| 40 kB 13.3 MB/s
Requirement already satisfied: six in /usr/local/lib/python3.7/site-packages (from happybase) (1.13.0)
Collecting thriftpy2>=0.4
  Downloading thriftpy2-0.5.2.tar.gz (782 kB)
     |████████████████████████████████| 782 kB 46.0 MB/s
  Installing build dependencies ... done
  WARNING: Missing build requirements in pyproject.toml for thriftpy2>=0.4 from https://files.pythonhosted.org/packages/f8/3a/d983b26df17583a3cc865a9e1737bb
faacfa1e16e3ed17353ef48847e6b/thriftpy2-0.5.2.tar.gz#sha256=cefcb2f6f8b12c00054c6f942dd2323a53b48b8b6862312d03b677dcf0d4a6da (from happybase).
  WARNING: The project does not specify a build backend, and pip cannot fall back to setuptools without 'wheel'.
  Getting requirements to build wheel ... done
  Installing backend dependencies ... done
    Preparing wheel metadata ... done
Collecting ply<4.0,>=3.4
  Downloading ply-3.11-py2.py3-none-any.whl (49 kB)
     |████████████████████████████████| 49 kB 13.1 MB/s
Collecting Cython>=3.0.10
  Using cached Cython-3.0.11-cp37-cp37m-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (3.6 MB)
Using legacy 'setup.py install' for happybase, since package 'wheel' is not installed.
Building wheels for collected packages: thriftpy2
  Building wheel for thriftpy2 (PEP 517) ... done
  Created wheel for thriftpy2: filename=thriftpy2-0.5.2-cp37-cp37m-linux_x86_64.whl size=1621426 sha256=db66884714b10f1cb4249dad1f7918ac2984a80c125e221df632
c7f39f6aafb
  Stored in directory: /root/.cache/pip/wheels/17/61/e8/9c4458a98088da816c0864fd90e7d7df01f36e4ee6e1fc599a
Successfully built thriftpy2
Installing collected packages: ply, Cython, thriftpy2, happybase
  WARNING: The scripts cygdb, cython and cythonize are installed in '/usr/local/bin' which is not on PATH.
  Consider adding this directory to PATH or, if you prefer to suppress this warning, use --no-warn-script-location.
    Running setup.py install for happybase ... done
ERROR: After October 2020 you may experience errors when installing or updating packages. This is because pip will change the way that it resolves dependenc
 conflicts.

We recommend you use --use-feature=2020-resolver to test your packages with the new resolver before it becomes the default.

thriftpy2 0.5.2 requires six~=1.15, but you'll have six 1.13.0 which is incompatible.
Successfully installed Cython-3.0.11 happybase-1.2.0 ply-3.11 thriftpy2-0.5.2
```

Step 10: Next for installing pandas, we ran the command **pip install pandas**



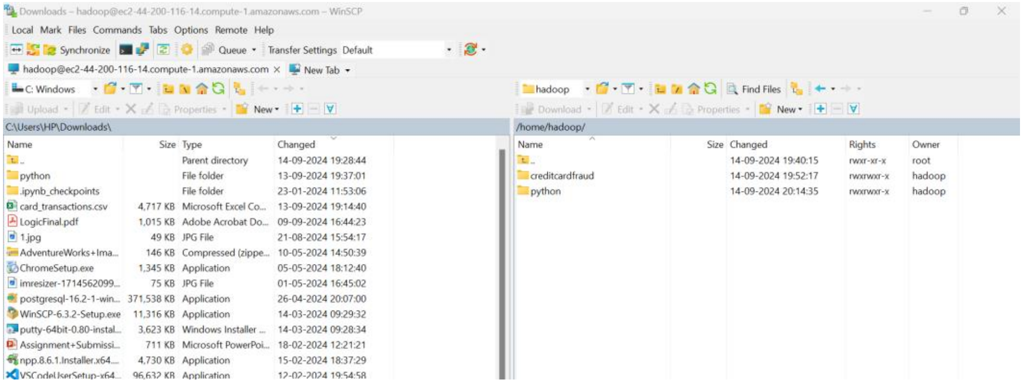
```
[root@ip-172-31-3-253 ~]# pip install pandas
WARNING: Running pip install with root privileges is generally not a good idea. Try `pip3 install --user` instead.
Collecting pandas
  Downloading pandas-1.3.5-cp37-cp37m-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (11.3 MB)
     |████████████████████████████████| 11.3 MB 24.5 MB/s
Requirement already satisfied: pytz>=2017.3 in /usr/local/lib/python3.7/site-packages (from pandas) (2019.3)
Collecting python-dateutil>=2.7.3
  Downloading python_dateutil-2.9.0.post0-py2.py3-none-any.whl (229 kB)
     |████████████████████████████████| 229 kB 82.5 MB/s
Collecting numpy>=1.17.3; platform_machine != "aarch64" and platform_machine != "arm64" and python_version < "3.10"
  Downloading numpy-1.21.6-cp37-cp37m-manylinux_2_12_x86_64.manylinux2010_x86_64.whl (15.7 MB)
     |████████████████████████████████| 15.7 MB 48.8 MB/s
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.7/site-packages (from python-dateutil>=2.7.3->pandas) (1.13.0)
Installing collected packages: python-dateutil, numpy, pandas
  Attempting uninstall: numpy
    Found existing installation: numpy 1.16.5
    Uninstalling numpy-1.16.5:
      Successfully uninstalled numpy-1.16.5
  WARNING: The scripts f2py, f2py3 and f2py3.7 are installed in '/usr/local/bin' which is not on PATH.
  Consider adding this directory to PATH or, if you prefer to suppress this warning, use --no-warn-script-location.
ERROR: After October 2020 you may experience errors when installing or updating packages. This is because pip will change the way that it resolves dependency
 conflicts.

We recommend you use --use-feature=2020-resolver to test your packages with the new resolver before it becomes the default.

python37-sagemaker-pyspark 1.3.0 requires pyspark==2.3.4, which is not installed.
Successfully installed numpy-1.21.6 pandas-1.3.5 python-dateutil-2.9.0.post0
[root@ip-172-31-3-253 ~]#
```

Step 11: Next for kickstarting the thrift server, we ran the command **/usr/lib/hbase/bin/hbase-daemon.sh start thrift -p 9090**



```
[root@ip-172-31-3-253 ~]# /usr/lib/hbase/bin/hbase-daemon.sh start thrift -p 9090
running thrift, logging to /usr/lib/hbase/bin/../logs/hbase-root-thrift-ip-172-31-3-253.out
[root@ip-172-31-3-253 ~]#
```

Step 12: Next we used winscp tool to move the python folder to /home/hadoop path



Downloads – hadoop@ec2-44-200-116-14.compute-1.amazonaws.com – WinSCP

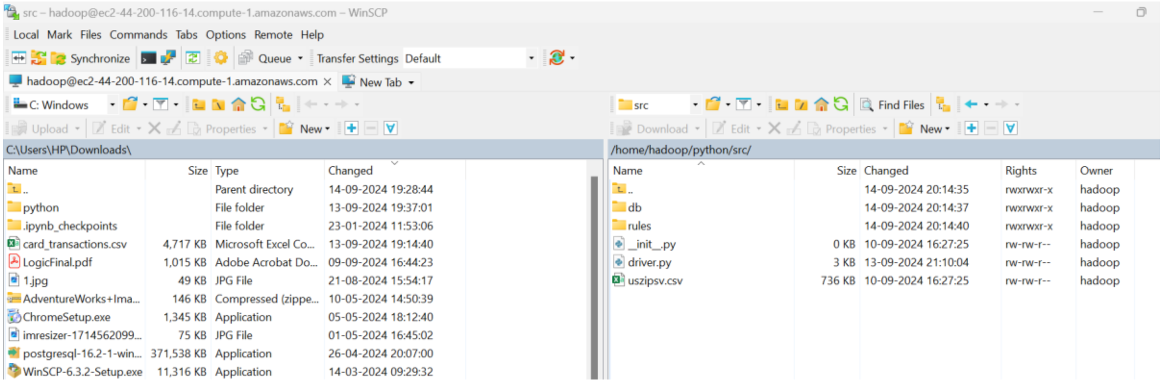
hadoop@ec2-44-200-116-14.compute-1.amazonaws.com

C:\Users\HP\Downloads\

Name	Size	Type	Changed
..		Parent directory	14-09-2024 19:28:44
python		File folder	13-09-2024 19:37:01
.ipynb_checkpoints		File folder	23-01-2024 11:53:06
card_transactions.csv	4,717 KB	Microsoft Excel Co...	13-09-2024 19:14:40
LogicFinal.pdf	1,015 KB	Adobe Acrobat Do...	09-09-2024 16:44:23
1.jpg	49 KB	JPG File	21-08-2024 15:54:17
AdventureWorks+Ima...	146 KB	Compressed (zippe...	10-05-2024 14:50:39
ChromeSetup.exe	1,345 KB	Application	05-05-2024 18:12:40
imresizer-1714562099...	75 KB	JPG File	01-05-2024 16:45:02
postgresql-16.2-1-win...	371,538 KB	Application	26-04-2024 20:07:00
WinSCP-6.3.2-Setup.exe	11,316 KB	Application	14-03-2024 09:29:32
putty-64bit-0.80-instal...	3,623 KB	Windows Installer ...	14-03-2024 09:28:34
Assignment+Submissi...	711 KB	Microsoft PowerPoi...	18-02-2024 12:21:21
npp.8.6.1.Installer.x64....	4,730 KB	Application	15-02-2024 18:37:29
VSCodeUserSetup-x64...	96,632 KB	Application	12-02-2024 19:54:58

/home/hadoop/

Name	Size	Changed	Rights	Owner
..		14-09-2024 19:40:15	rwxr-xr-x	root
creditcardfraud		14-09-2024 19:52:17	rwxrwxr-x	hadoop
python		14-09-2024 20:14:35	rwxrwxr-x	hadoop



src – hadoop@ec2-44-200-116-14.compute-1.amazonaws.com – WinSCP

hadoop@ec2-44-200-116-14.compute-1.amazonaws.com

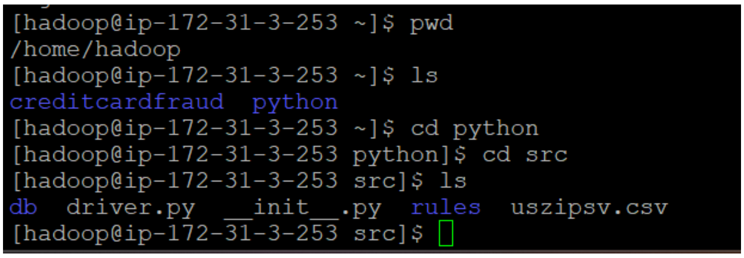
C:\Users\HP\Downloads\

Name	Size	Type	Changed
..		Parent directory	14-09-2024 19:28:44
python		File folder	13-09-2024 19:37:01
.ipynb_checkpoints		File folder	23-01-2024 11:53:06
card_transactions.csv	4,717 KB	Microsoft Excel Co...	13-09-2024 19:14:40
LogicFinal.pdf	1,015 KB	Adobe Acrobat Do...	09-09-2024 16:44:23
1.jpg	49 KB	JPG File	21-08-2024 15:54:17
AdventureWorks+Ima...	146 KB	Compressed (zippe...	10-05-2024 14:50:39
ChromeSetup.exe	1,345 KB	Application	05-05-2024 18:12:40
imresizer-1714562099...	75 KB	JPG File	01-05-2024 16:45:02
postgresql-16.2-1-win...	371,538 KB	Application	26-04-2024 20:07:00
WinSCP-6.3.2-Setup.exe	11,316 KB	Application	14-03-2024 09:29:32

/home/hadoop/python/src/

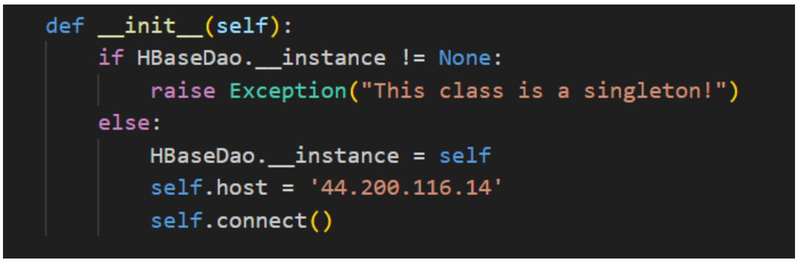
Name	Size	Changed	Rights	Owner
..		14-09-2024 20:14:35	rwxrwxr-x	hadoop
db		14-09-2024 20:14:37	rwxrwxr-x	hadoop
rules		14-09-2024 20:14:40	rwxrwxr-x	hadoop
__init__.py	0 KB	10-09-2024 16:27:25	rw-rw-r--	hadoop
driver.py	3 KB	13-09-2024 21:10:04	rw-rw-r--	hadoop
uszipsv.csv	736 KB	10-09-2024 16:27:25	rw-rw-r--	hadoop

Step 13: Next we checked in EMR end if the python folder and its contents got loaded correctly or not.



```
[hadoop@ip-172-31-3-253 ~]$ pwd
/home/hadoop
[hadoop@ip-172-31-3-253 ~]$ ls
creditcardfraud  python
[hadoop@ip-172-31-3-253 ~]$ cd python
[hadoop@ip-172-31-3-253 python]$ cd src
[hadoop@ip-172-31-3-253 src]$ ls
db  driver.py  __init__.py  rules  uszipsv.csv
[hadoop@ip-172-31-3-253 src]$
```

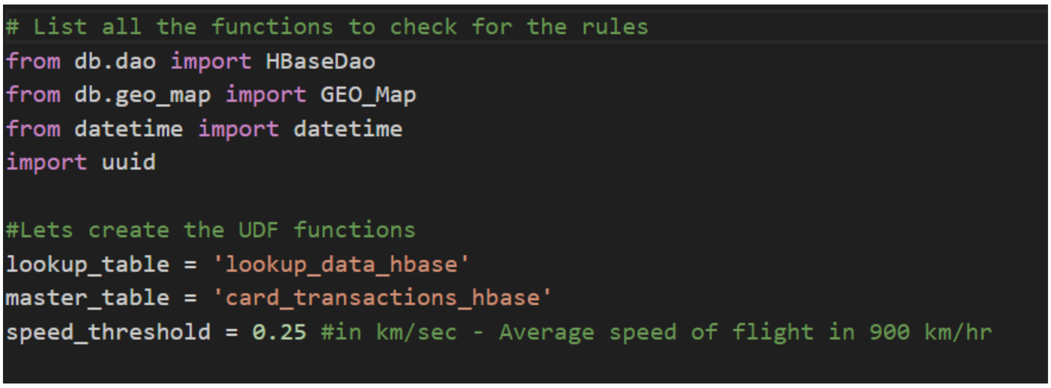
Step 14: Next in dao.py file -> __init__ function -> in self.host, we kept the public IP address of our EMR cluster.



```
def __init__(self):
    if HBaseDao.__instance != None:
        raise Exception("This class is a singleton!")
    else:
        HBaseDao.__instance = self
        self.host = '44.200.116.14'
        self.connect()
```

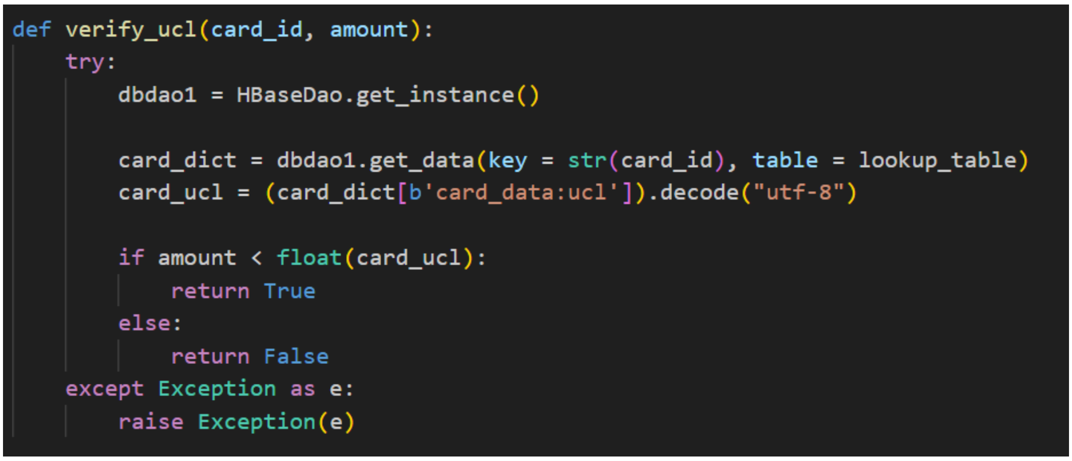
Step 15: Next in rules.py file, we updated the below parameters:

- lookup_table = ‘lookup_data_hbase’
- master_table = ‘card_transactions_hbase’



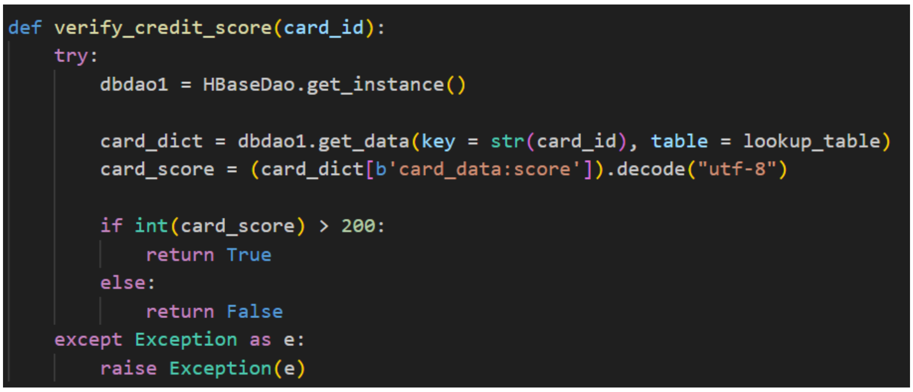
```
# List all the functions to check for the rules
from db.dao import HBaseDao
from db.geo_map import GEO_Map
from datetime import datetime
import uuid

#Lets create the UDF functions
lookup_table = 'lookup_data_hbase'
master_table = 'card_transactions_hbase'
speed_threshold = 0.25 #in km/sec - Average speed of flight in 900 km/hr
```

Step 16: Next we created a UDF for verifying the UCL (Upper control limit) – the transaction amount should be less than the UCL.



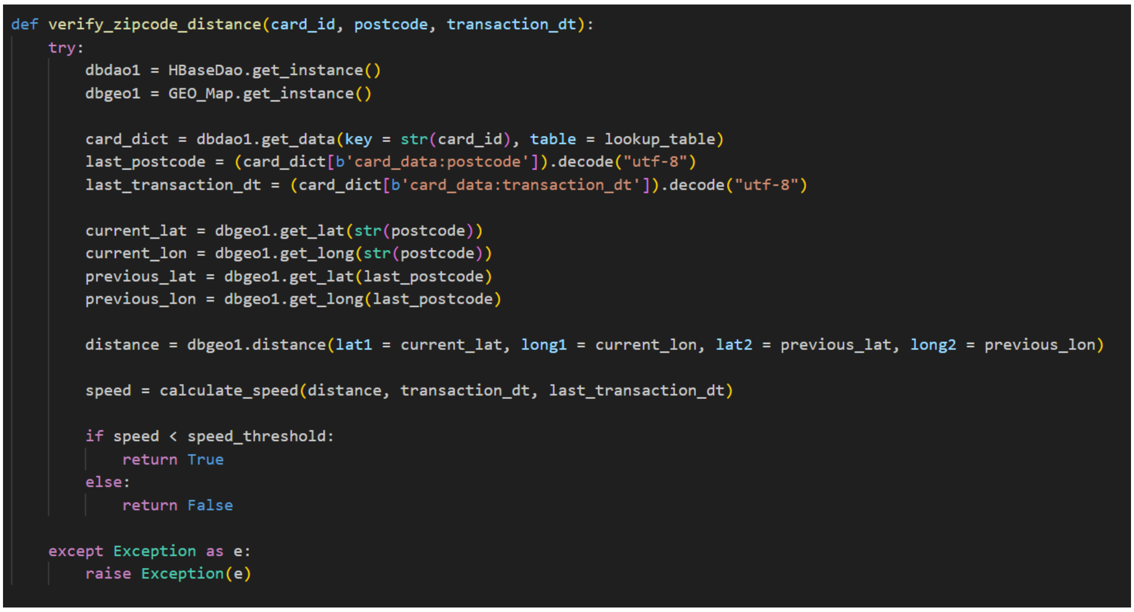
```
def verify_ucl(card_id, amount):
    try:
        dbdao1 = HBaseDao.get_instance()

        card_dict = dbdao1.get_data(key = str(card_id), table = lookup_table)
        card_ucl = (card_dict[b'card_data:ucl']).decode("utf-8")

        if amount < float(card_ucl):
            return True
        else:
            return False
    except Exception as e:
        raise Exception(e)
```

Step 17: Next we created a UDF for verifying the credit score i.e. credit score of each member should be greater than 200.



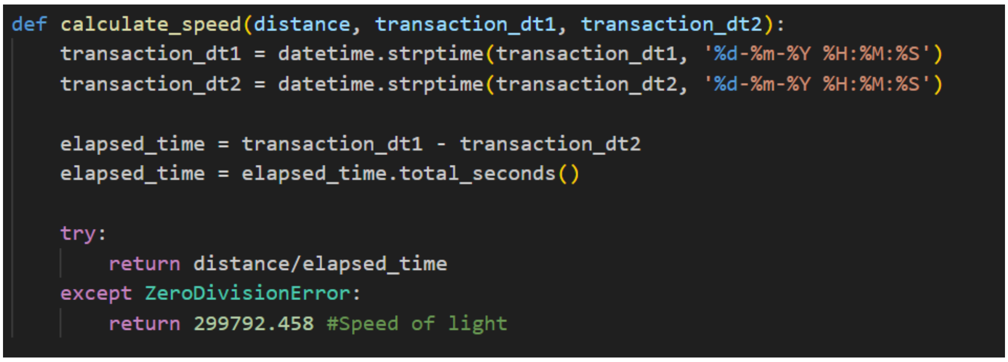
```
def verify_credit_score(card_id):
    try:
        dbdao1 = HBaseDao.get_instance()

        card_dict = dbdao1.get_data(key = str(card_id), table = lookup_table)
        card_score = (card_dict[b'card_data:score']).decode("utf-8")

        if int(card_score) > 200:
            return True
        else:
            return False
    except Exception as e:
        raise Exception(e)
```

Step 18: Next we create a UDF for verifying the zipcode distance.



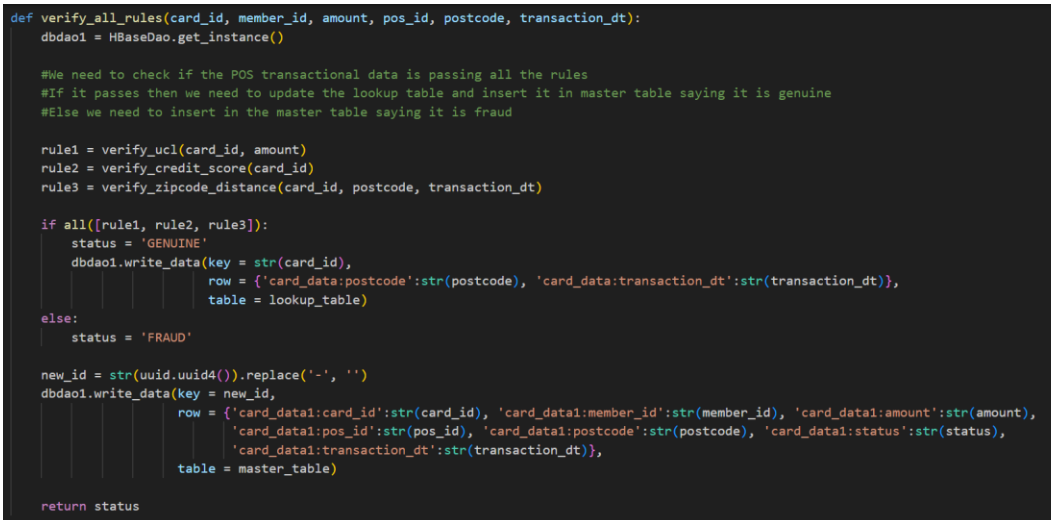
```
def verify_zipcode_distance(card_id, postcode, transaction_dt):
    try:
        dbdao1 = HBaseDao.get_instance()
        dbgeo1 = GEO_Map.get_instance()

        card_dict = dbdao1.get_data(key = str(card_id), table = lookup_table)
        last_postcode = (card_dict[b'card_data:postcode']).decode("utf-8")
        last_transaction_dt = (card_dict[b'card_data:transaction_dt']).decode("utf-8")

        current_lat = dbgeo1.get_lat(str(postcode))
        current_lon = dbgeo1.get_long(str(postcode))
        previous_lat = dbgeo1.get_lat(last_postcode)
        previous_lon = dbgeo1.get_long(last_postcode)

        distance = dbgeo1.distance(lat1 = current_lat, long1 = current_lon, lat2 = previous_lat, long2 = previous_lon)

        speed = calculate_speed(distance, transaction_dt, last_transaction_dt)

        if speed < speed_threshold:
            return True
        else:
            return False

    except Exception as e:
        raise Exception(e)
```

Step 19: Next we created a UDF for calculating the speed from distance and transaction timestamp differentials.



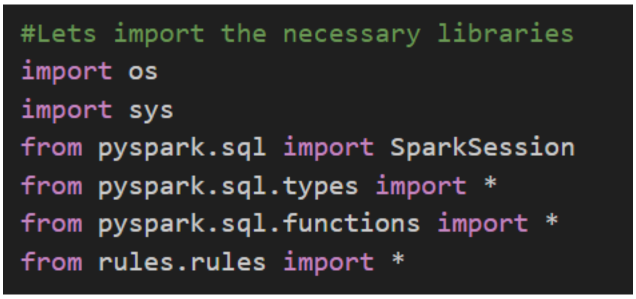
```
def calculate_speed(distance, transaction_dt1, transaction_dt2):
    transaction_dt1 = datetime.strptime(transaction_dt1, '%d-%m-%Y %H:%M:%S')
    transaction_dt2 = datetime.strptime(transaction_dt2, '%d-%m-%Y %H:%M:%S')

    elapsed_time = transaction_dt1 - transaction_dt2
    elapsed_time = elapsed_time.total_seconds()

    try:
        return distance/elapsed_time
    except ZeroDivisionError:
        return 299792.458 #Speed of light
```

Step 20: Next we wrote a function to verify all the 3 rules – UCL, credit score and zipcode distance.



```
def verify_all_rules(card_id, member_id, amount, pos_id, postcode, transaction_dt):
    dbdao1 = HBaseDao.get_instance()

    #We need to check if the POS transactional data is passing all the rules
    #If it passes then we need to update the lookup table and insert it in master table saying it is genuine
    #Else we need to insert in the master table saying it is fraud

    rule1 = verify_ucl(card_id, amount)
    rule2 = verify_credit_score(card_id)
    rule3 = verify_zipcode_distance(card_id, postcode, transaction_dt)

    if all([rule1, rule2, rule3]):
        status = 'GENUINE'
        dbdao1.write_data(key = str(card_id),
                          row = {'card_data:postcode':str(postcode), 'card_data:transaction_dt':str(transaction_dt)},
                          table = lookup_table)
    else:
        status = 'FRAUD'

    new_id = str(uuid.uuid4()).replace('-', '')
    dbdao1.write_data(key = new_id,
                      row = {'card_data1:card_id':str(card_id), 'card_data1:member_id':str(member_id), 'card_data1:amount':str(amount),
                             'card_data1:pos_id':str(pos_id), 'card_data1:postcode':str(postcode), 'card_data1:status':str(status),
                             'card_data1:transaction_dt':str(transaction_dt)},
                      table = master_table)

    return status
```

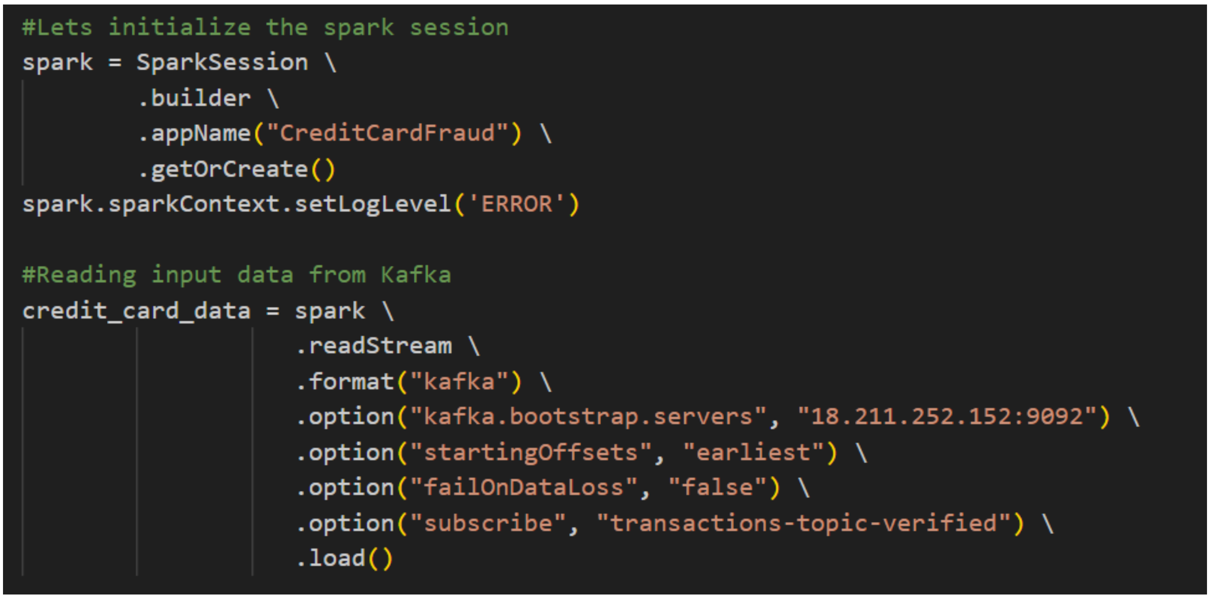
Step 21: Next in the driver.py file we need to make sure to set up the system dependencies and importing necessary libraries and modules.



```
#Lets import the necessary libraries
import os
import sys
from pyspark.sql import SparkSession
from pyspark.sql.types import *
from pyspark.sql.functions import *
from rules.rules import *
```

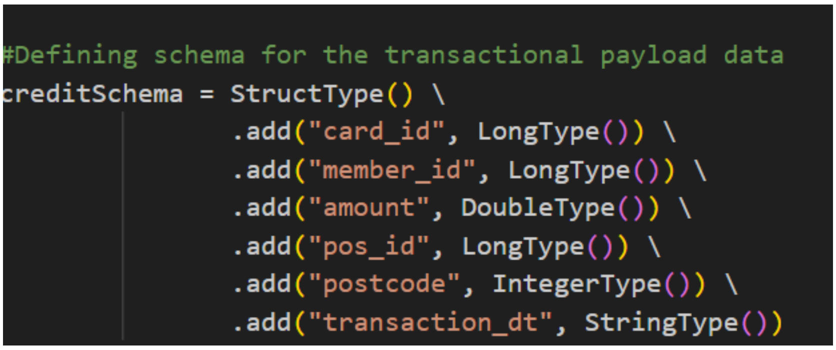
Step 22: Next we initialized the Spark session and read input data from Kafka by mentioning the details of the Kafka broker, such as bootstrap server, port and topic name.

- Bootstrap-server: 18.211.252.152
- Port Number: 9092
- Topic: transactions-topic-verified



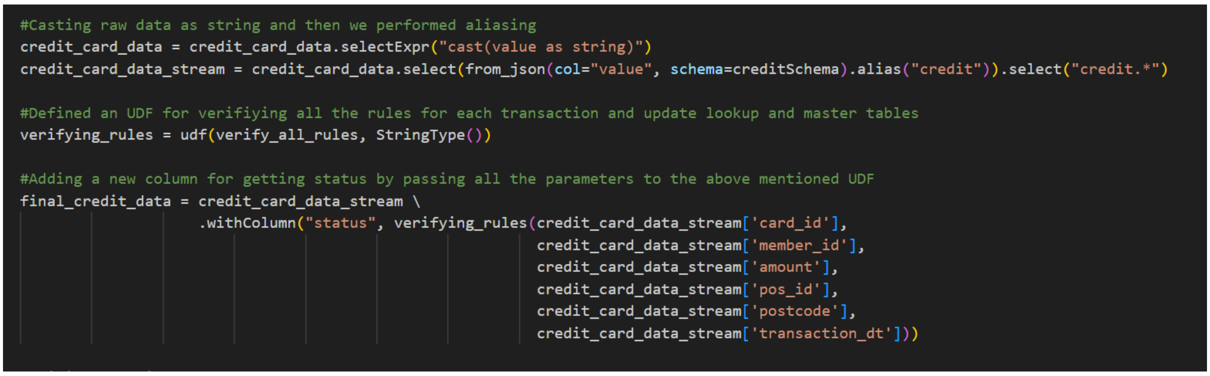
```
#Lets initialize the spark session
spark = SparkSession \
        .builder \
        .appName("CreditCardFraud") \
        .getOrCreate()
spark.sparkContext.setLogLevel('ERROR')

#Reading input data from Kafka
credit_card_data = spark \
                   .readStream \
                   .format("kafka") \
                   .option("kafka.bootstrap.servers", "18.211.252.152:9092") \
                   .option("startingOffsets", "earliest") \
                   .option("failOnDataLoss", "false") \
                   .option("subscribe", "transactions-topic-verified") \
                   .load()
```

Step 23: Next we defined the JSON schema for the transactions that will come from Kafka.



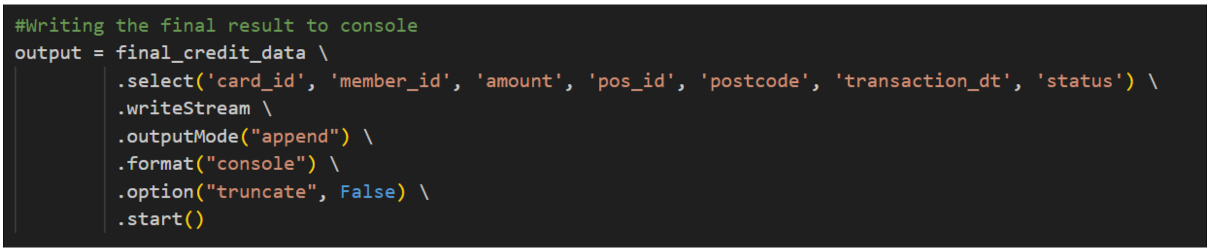
```
#Defining schema for the transactional payload data
creditSchema = StructType() \
               .add("card_id", LongType()) \
               .add("member_id", LongType()) \
               .add("amount", DoubleType()) \
               .add("pos_id", LongType()) \
               .add("postcode", IntegerType()) \
               .add("transaction_dt", StringType())
```

Step 24: Next we read the raw JSON data from Kafka as 'credit_card_data_stream' and defined a UDF to verify rules and also update the lookup table and master table accordingly as coded in verify_all_rules.



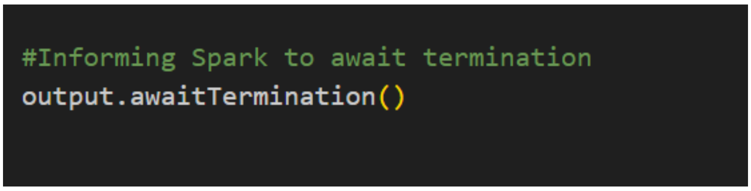
```
#Casting raw data as string and then we performed aliasing
credit_card_data = credit_card_data.selectExpr("cast(value as string)")
credit_card_data_stream = credit_card_data.select(from_json(col="value", schema=creditSchema).alias("credit")).select("credit.*")

#Defined an UDF for verifiying all the rules for each transaction and update lookup and master tables
verifying_rules = udf(verify_all_rules, StringType())

#Adding a new column for getting status by passing all the parameters to the above mentioned UDF
final_credit_data = credit_card_data_stream \
                    .withColumn("status", verifying_rules(credit_card_data_stream['card_id'],
                                                          credit_card_data_stream['member_id'],
                                                          credit_card_data_stream['amount'],
                                                          credit_card_data_stream['pos_id'],
                                                          credit_card_data_stream['postcode'],
                                                          credit_card_data_stream['transaction_dt']))
```

Step 25: Next we wrote the below code snippet to write the output to console.



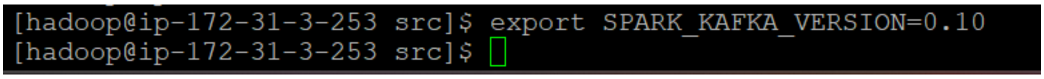
```
#Writing the final result to console
output = final_credit_data \
          .select('card_id', 'member_id', 'amount', 'pos_id', 'postcode', 'transaction_dt', 'status') \
          .writeStream \
          .outputMode("append") \
          .format("console") \
          .option("truncate", False) \
          .start()
```

Step 26: Next we wrote below line for awaiting termination.



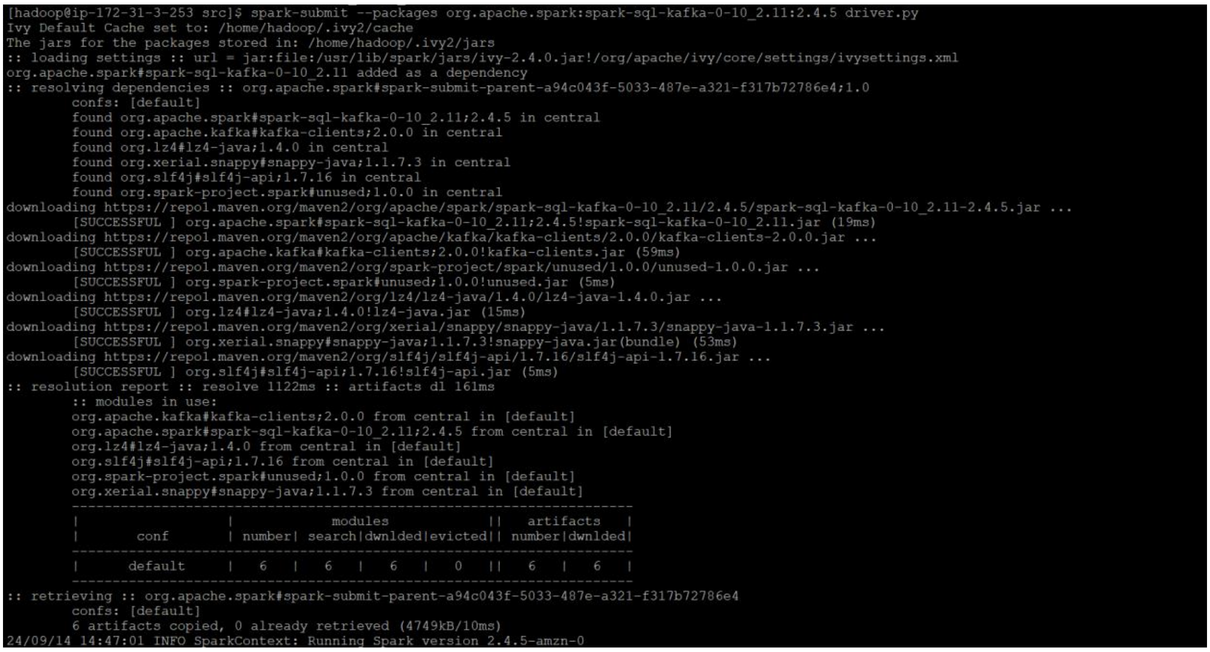
```
#Informing Spark to await termination
output.awaitTermination()
```

Step 27: Next we need to setup spark-kafka for which we ran the command **export SPARK_KAFKA_VERSION=0.10**



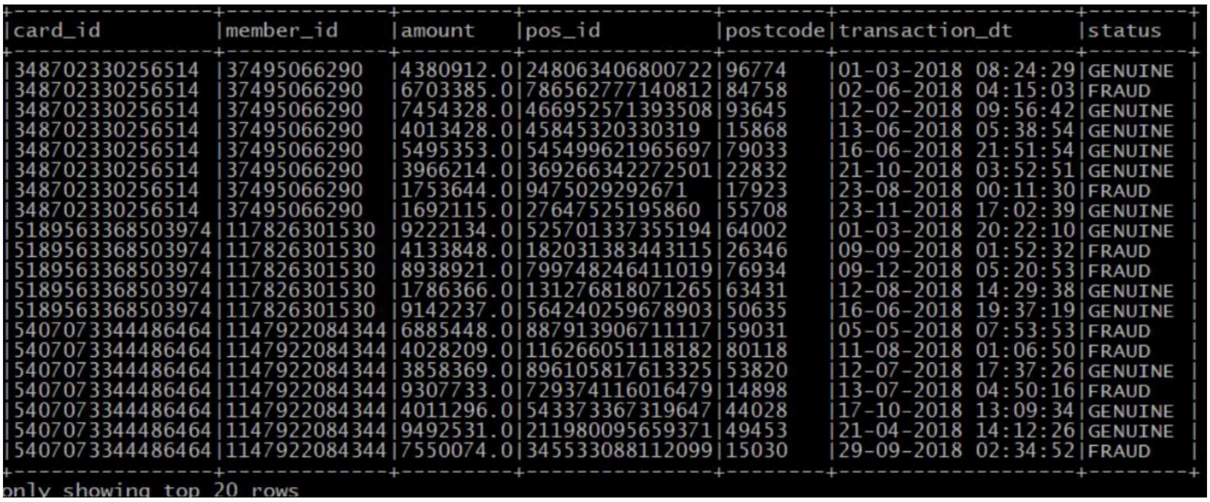
```
[hadoop@ip-172-31-3-253 src]$ export SPARK_KAFKA_VERSION=0.10
[hadoop@ip-172-31-3-253 src]$ 
```

Step 28: Next we ran the spark-submit command in which we specified the Spark-SQL-Kafka package and python driver file **spark-submit --packages org.apache.spark:spark-sql-kafka-0-10_2.11:2.4.5 driver.py**



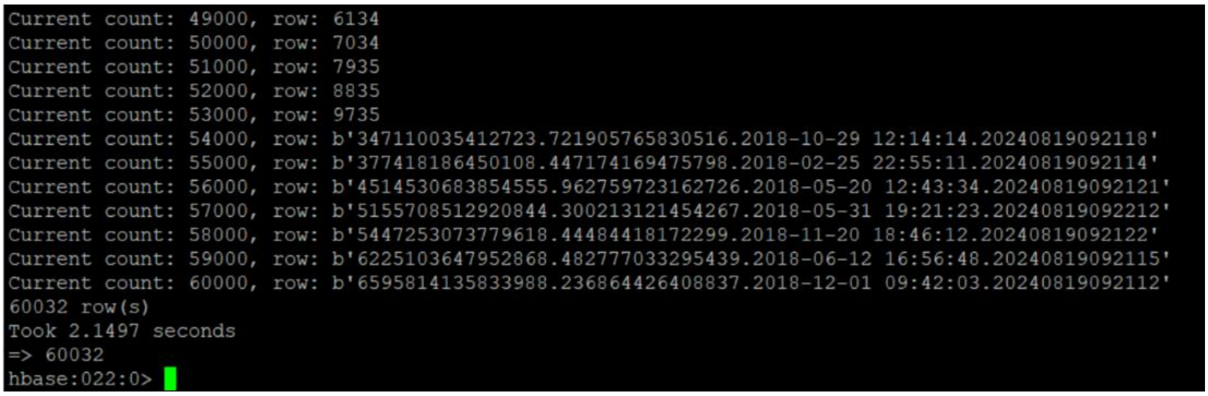
```
[hadoop@ip-172-31-3-253 src]$ spark-submit --packages org.apache.spark:spark-sql-kafka-0-10_2.11:2.4.5 driver.py
Ivy Default Cache set to: /home/hadoop/.ivy2/cache
The jars for the packages stored in: /home/hadoop/.ivy2/jars
:: loading settings :: url = jar:file:/usr/lib/spark/jars/ivy-2.4.0.jar!/org/apache/ivy/core/settings/ivysettings.xml
org.apache.spark#spark-sql-kafka-0-10_2.11 added as a dependency
:: resolving dependencies :: org.apache.spark#spark-submit-parent-a94c043f-5033-487e-a321-f317b72786e4;1.0
        confs: [default]
        found org.apache.spark#spark-sql-kafka-0-10_2.11;2.4.5 in central
        found org.apache.kafka#kafka-clients;2.0.0 in central
        found org.lz4#lz4-java;1.4.0 in central
        found org.xerial.snappy#snappy-java;1.1.7.3 in central
        found org.slf4j#slf4j-api;1.7.16 in central
        found org.spark-project.spark#unused;1.0.0 in central
downloading https://repo1.maven.org/maven2/org/apache/spark/spark-sql-kafka-0-10_2.11/2.4.5/spark-sql-kafka-0-10_2.11-2.4.5.jar ...
        [SUCCESSFUL ] org.apache.spark#spark-sql-kafka-0-10_2.11;2.4.5!spark-sql-kafka-0-10_2.11.jar (19ms)
downloading https://repo1.maven.org/maven2/org/apache/kafka/kafka-clients/2.0.0/kafka-clients-2.0.0.jar ...
        [SUCCESSFUL ] org.apache.kafka#kafka-clients;2.0.0!kafka-clients.jar (59ms)
downloading https://repo1.maven.org/maven2/org/spark-project/spark/unused/1.0.0/unused-1.0.0.jar ...
        [SUCCESSFUL ] org.spark-project.spark#unused;1.0.0!unused.jar (5ms)
downloading https://repo1.maven.org/maven2/org/lz4/lz4-java/1.4.0/lz4-java-1.4.0.jar ...
        [SUCCESSFUL ] org.lz4#lz4-java;1.4.0!lz4-java.jar (15ms)
downloading https://repo1.maven.org/maven2/org/xerial/snappy/snappy-java/1.1.7.3/snappy-java-1.1.7.3.jar ...
        [SUCCESSFUL ] org.xerial.snappy#snappy-java;1.1.7.3!snappy-java.jar(bundle) (53ms)
downloading https://repo1.maven.org/maven2/org/slf4j/slf4j-api/1.7.16/slf4j-api-1.7.16.jar ...
        [SUCCESSFUL ] org.slf4j#slf4j-api;1.7.16!slf4j-api.jar (5ms)
:: resolution report :: resolve 1122ms :: artifacts dl 161ms
        :: modules in use:
        org.apache.kafka#kafka-clients;2.0.0 from central in [default]
        org.apache.spark#spark-sql-kafka-0-10_2.11;2.4.5 from central in [default]
        org.lz4#lz4-java;1.4.0 from central in [default]
        org.slf4j#slf4j-api;1.7.16 from central in [default]
        org.spark-project.spark#unused;1.0.0 from central in [default]
        org.xerial.snappy#snappy-java;1.1.7.3 from central in [default]
        ---------------------------------------------------------------------
        |                  |            modules            ||   artifacts   |
        |       conf       | number| search|dwnlded|evicted|| number|dwnlded|
        ---------------------------------------------------------------------
        |      default     |   6   |   6   |   6   |   0   ||   6   |   6   |
        ---------------------------------------------------------------------
:: retrieving :: org.apache.spark#spark-submit-parent-a94c043f-5033-487e-a321-f317b72786e4
        confs: [default]
        6 artifacts copied, 0 already retrieved (4749kB/10ms)
24/09/14 14:47:01 INFO SparkContext: Running Spark version 2.4.5-amzn-0
```

Step 29: Below is how the output will come on console.



```
+----------------+-------------+---------+---------------+--------+-------------------+-------+
|card_id         |member_id    |amount   |pos_id         |postcode|transaction_dt     |status |
+----------------+-------------+---------+---------------+--------+-------------------+-------+
|348702330256514 |37495066290  |4380912.0|248063406800722|96774   |01-03-2018 08:24:29|GENUINE|
|348702330256514 |37495066290  |6703385.0|786562777140812|84758   |02-06-2018 04:15:03|FRAUD  |
|348702330256514 |37495066290  |7454328.0|466952571393508|93645   |12-02-2018 09:56:42|GENUINE|
|348702330256514 |37495066290  |4013428.0|45845320330319 |15868   |13-06-2018 05:38:54|GENUINE|
|348702330256514 |37495066290  |5495353.0|545499621965697|79033   |16-06-2018 21:51:54|GENUINE|
|348702330256514 |37495066290  |3966214.0|369266342272501|22832   |21-10-2018 03:52:51|GENUINE|
|348702330256514 |37495066290  |1753644.0|9475029292671  |17923   |23-08-2018 00:11:30|FRAUD  |
|348702330256514 |37495066290  |1692115.0|27647525195860 |55708   |23-11-2018 17:02:39|GENUINE|
|5189563368503974|117826301530 |9222134.0|525701337355194|64002   |01-03-2018 20:22:10|GENUINE|
|5189563368503974|117826301530 |4133848.0|182031383443115|26346   |09-09-2018 01:52:32|FRAUD  |
|5189563368503974|117826301530 |8938921.0|799748246411019|76934   |09-12-2018 05:20:53|FRAUD  |
|5189563368503974|117826301530 |1786366.0|131276818071265|63431   |12-08-2018 14:29:38|GENUINE|
|5189563368503974|117826301530 |9142237.0|564240259678903|50635   |16-06-2018 19:37:19|GENUINE|
|5407073344486464|1147922084344|6885448.0|887913906711117|59031   |05-05-2018 07:53:53|FRAUD  |
|5407073344486464|1147922084344|4028209.0|116266051118182|80118   |11-08-2018 01:06:50|FRAUD  |
|5407073344486464|1147922084344|3858369.0|896105817613325|53820   |12-07-2018 17:37:26|GENUINE|
|5407073344486464|1147922084344|9307733.0|729374116016479|14898   |13-07-2018 04:50:16|FRAUD  |
|5407073344486464|1147922084344|4011296.0|543373367319647|44028   |17-10-2018 13:09:34|GENUINE|
|5407073344486464|1147922084344|9492531.0|211980095659371|49453   |21-04-2018 14:12:26|GENUINE|
|5407073344486464|1147922084344|7550074.0|345533088112099|15030   |29-09-2018 02:34:52|FRAUD  |
+----------------+-------------+---------+---------------+--------+-------------------+-------+
only showing top 20 rows
```

Step 30: Next we ran the command **count ‘card_transactions_hbase1’** for which we got 60032 which is more than 53292 (validation given - when we classify all the incoming transactions as fraud or genuine and then update this in the card_transactions table, the final count of that table should be more than 53292).



```
Current count: 49000, row: 6134
Current count: 50000, row: 7034
Current count: 51000, row: 7935
Current count: 52000, row: 8835
Current count: 53000, row: 9735
Current count: 54000, row: b'347110035412723.721905765830516.2018-10-29 12:14:14.20240819092118'
Current count: 55000, row: b'377418186450108.447174169475798.2018-02-25 22:55:11.20240819092114'
Current count: 56000, row: b'4514530683854555.962759723162726.2018-05-20 12:43:34.20240819092121'
Current count: 57000, row: b'5155708512920844.300213121454267.2018-05-31 19:21:23.20240819092212'
Current count: 58000, row: b'5447253073779618.44484418172299.2018-11-20 18:46:12.20240819092122'
Current count: 59000, row: b'6225103647952868.482777033295439.2018-06-12 16:56:48.20240819092115'
Current count: 60000, row: b'6595814135833988.236864426408837.2018-12-01 09:42:03.20240819092112'
60032 row(s)
Took 2.1497 seconds
=> 60032
hbase:022:0>
```