

Configuration Manual

MSc Internship
Cyber Security

Ritesh Naresh Gohil
Student ID: X18205836

School of Computing
National College of Ireland

Supervisor: Mr. Vikas Sahni

National College of Ireland
MSc Project Submission Sheet



School of Computing

Student Name:	Ritesh Naresh Gohil		
Student ID:	X18205836		
Programme:	Cyber Security	Year:	2020
Module:	MSc Internship		
Lecturer:	Mr. Vikas Sahni		
Submission Due Date:	17/08/2020		
Project Title:	Fast and accurate classification of threats in IDS using Distributed Machine Learning techniques.		
Word Count:	1038	Page Count:	15

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

I agree to an electronic copy of my thesis being made publicly available on NORMA the National College of Ireland's Institutional Repository for consultation.

Signature: Ritesh Naresh Gohil
.....

Date: 17/08/2020
.....

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	<input type="checkbox"/>
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	<input type="checkbox"/>
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	<input type="checkbox"/>

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:
Date:
Penalty Applied (if applicable):

Configuration Manual

Ritesh Naresh Gohil
Student ID: X18205836

1. Introduction

This configuration manual will provide in depth information about configuration and implementation of these Research Project which includes system specifications and software requirements. Python is used for Traditional Machine Learning whereas Hadoop Spark framework and Pyspark language is used for classification of threats in IDS using Distributed Machine Learning Techniques.

2. Project Setup

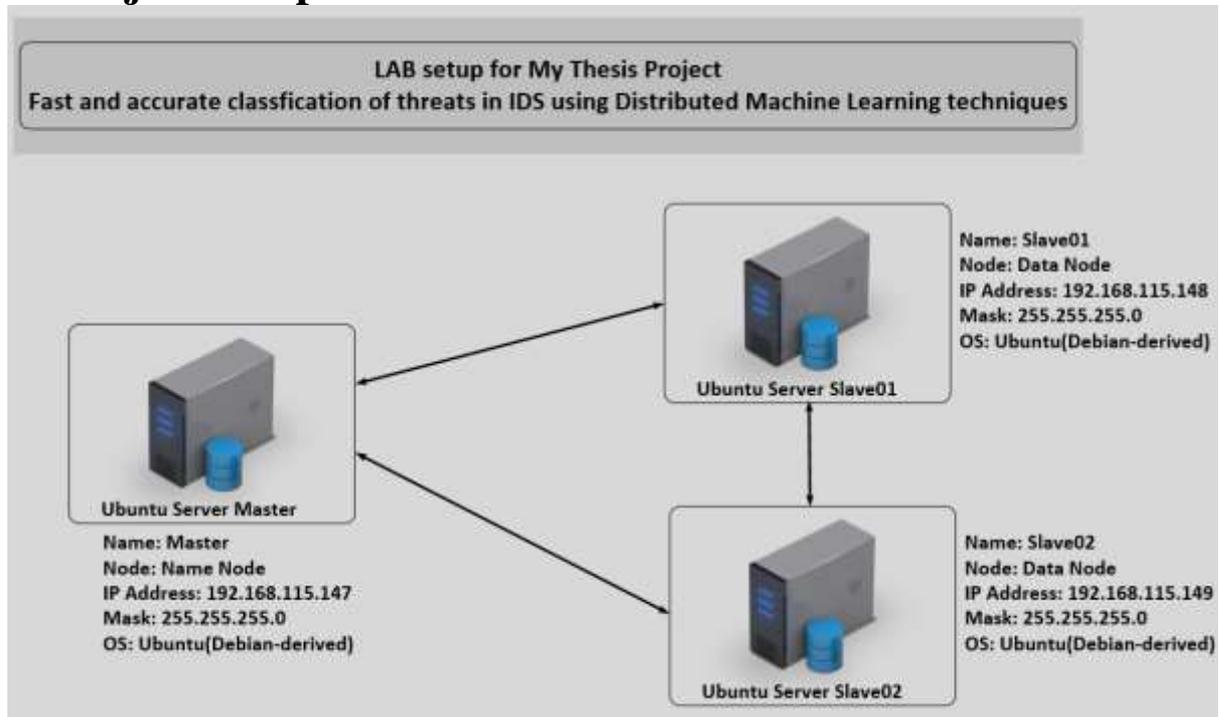


Figure 1: Project Setup

2.1 System Requirements

Ubuntu Server (Master)	Ubuntu Server (Slave01)	Ubuntu Server (Slave02)
4 GB RAM	4 GB RAM	4 GB RAM
40GB HDD	40GB HDD	40GB HDD
Number of Processor 2	Number of Processor 2	Number of Processor 2
Number of Cores per Processor 2	Number of Cores per Processor 2	Number of Cores per Processor 2
Network Adapter: NAT	Network Adapter: NAT	Network Adapter: NAT

Table 1: System Requirements

3. Software and Tools Used

- VMWare Workstation 15.0 Pro
- Anaconda 2020
- Python 3.8
- Jupyter Notebook 6.0
- Pyspark

4. Configuration Steps

Below are the steps followed for Hadoop Distributed File System (HDFS) installation and configuration:-

Step 1: Installation of Java JDK version 8.

Command: `sudo apt-get install openjdk-8-jdk`

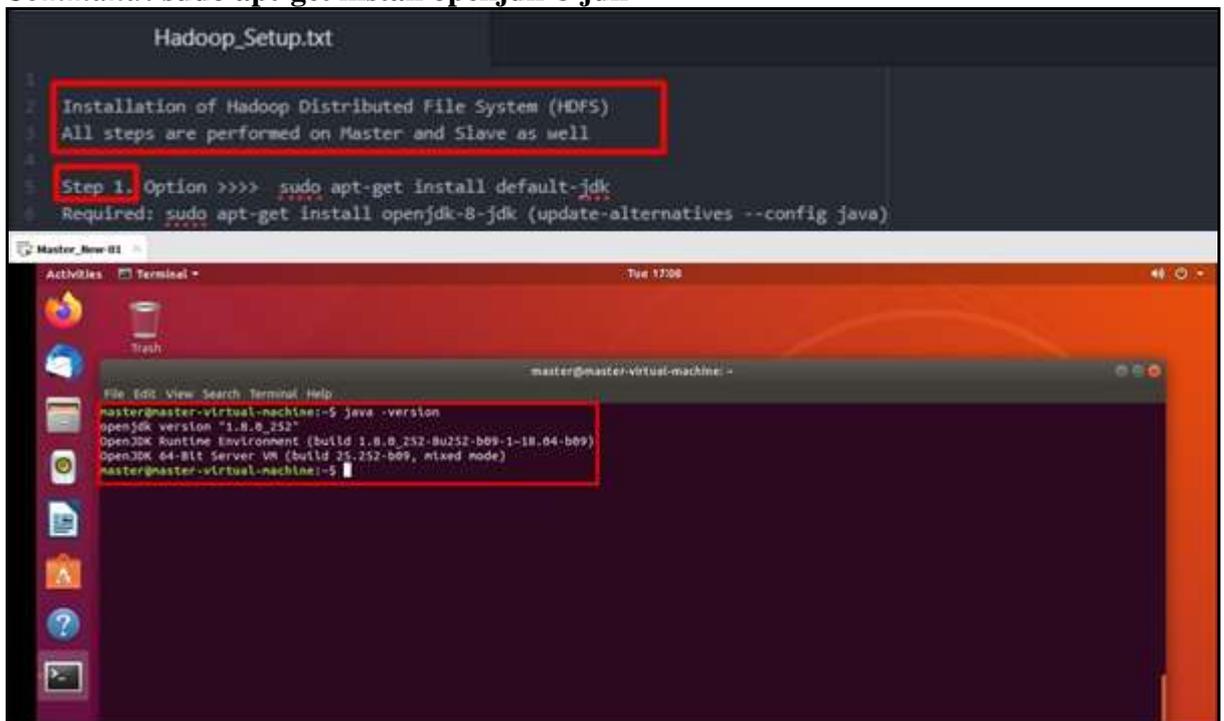


Figure 2: Validate Java Version

Step 2: Creating a user for Hadoop group whose name is hduser.



Figure 3: Created hduser

Step 3: Add all hostname and IP addresses in /etc/hosts file

Command: vim /etc/hosts

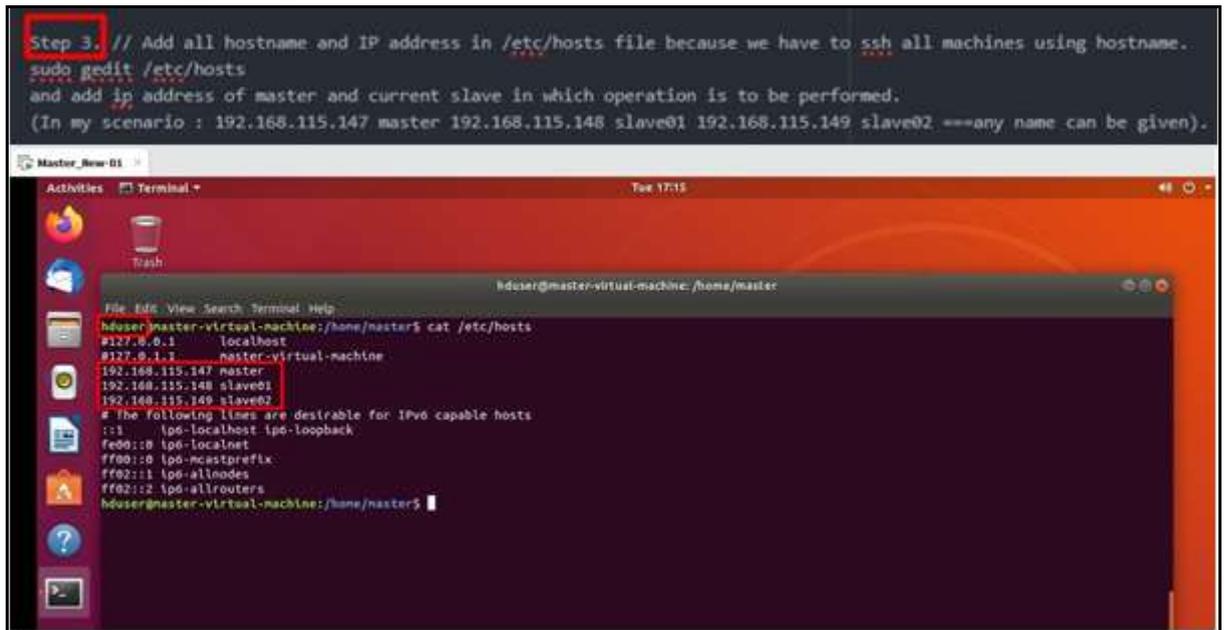


Figure 4: Entry of hosts in /etc/hosts file

Step 4: Install OpenSSH and Share SSH public key to other machines

Command: sudo apt-get install openssh-server

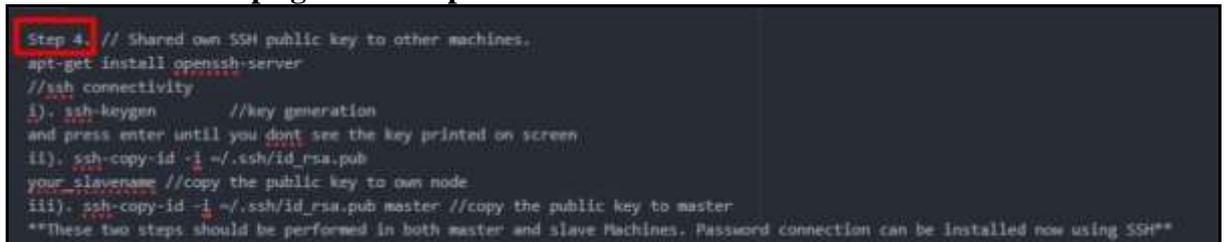


Figure 5: Generate and share SSH public key

Step 5: Global variable JAVA_HOME, HADOOP_HOME, SPARK_HOME in .bashrc file.

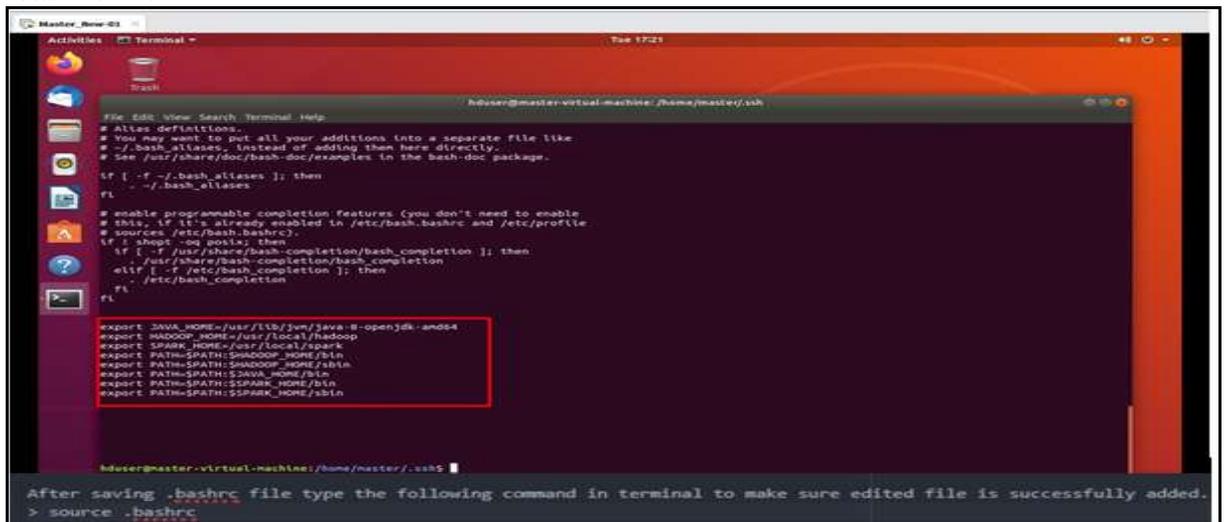


Figure 6: .bashrc file

Step 6: Install Hadoop and verify \$HADOOP_HOME.

Command: `sudo wget https://archive.apache.org/dist/hadoop/common/hadoop-3.2.0/`

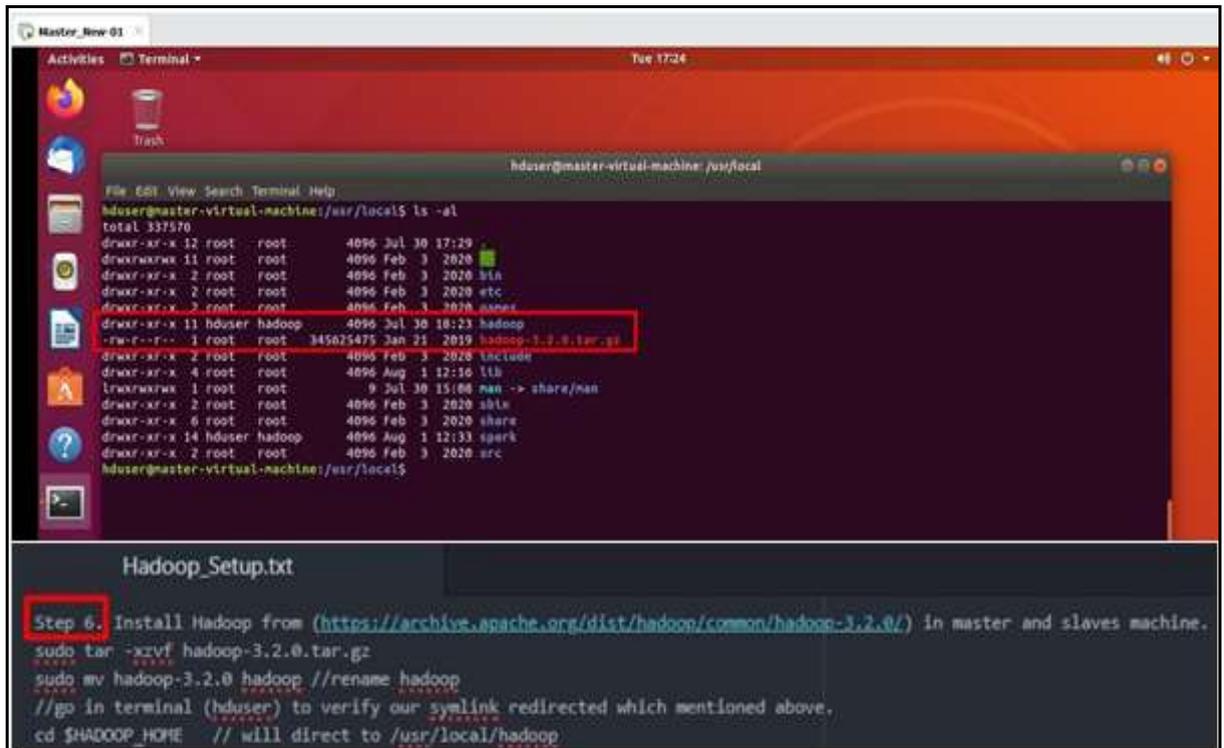


Figure 7: Install Hadoop

Step 7: edit hadoop-env.sh file which is located at /hadoop/etc/hadoop/.

Command: `vim /hadoop.hadoop-env.sh`

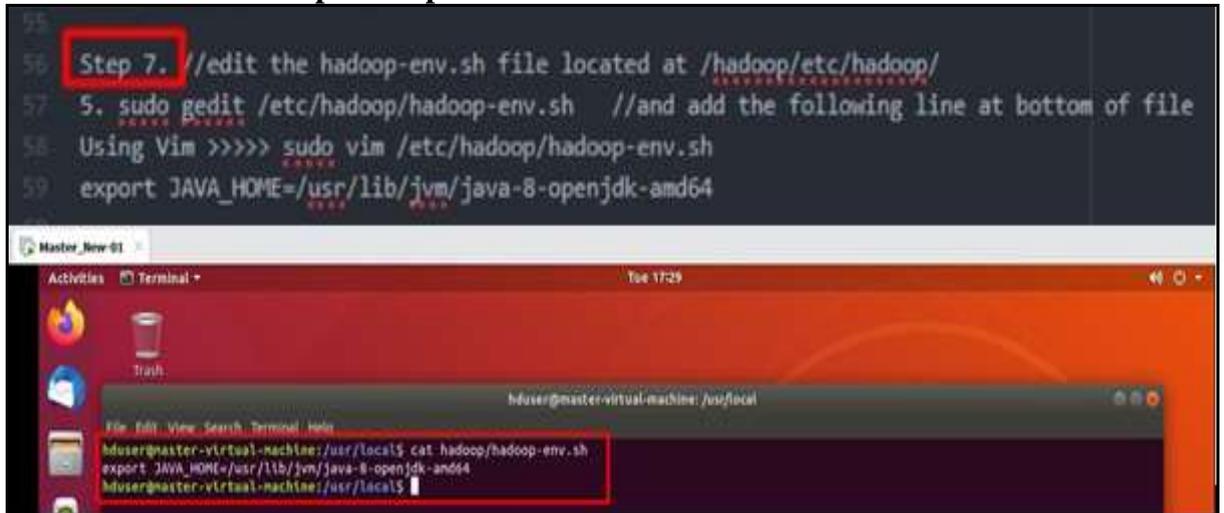
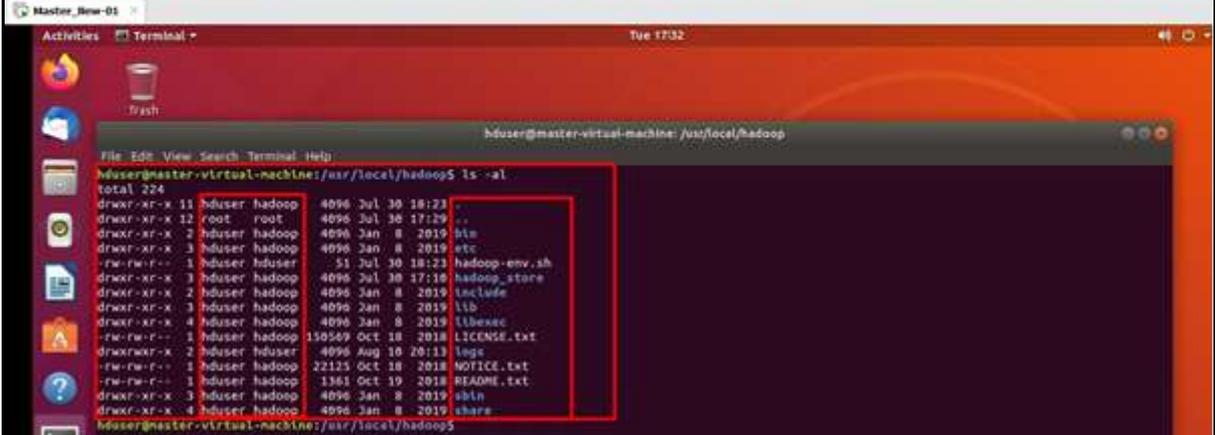


Figure 8: hadoop-env.sh

Step 8: Make directories and provide permissions for Hadoop hduser.

```
61 Step 8. //make some directories and give them permission
62 sudo mkdir -p /app/hadoop/tmp
63 sudo chown hduser:hadoop /app/hadoop/tmp
64 sudo mkdir -p /usr/local/hadoop/hadoop_store/hdfs/namenode
65 sudo mkdir -p /usr/local/hadoop/hadoop_store/hdfs/datanode
66 sudo chown -R hduser:hadoop /usr/local/hadoop/hadoop_store
```



The screenshot shows a terminal window with the following output:

```
hduser@master-virtual-machine:/usr/local/hadoop$ ls -al
total 224
drwxr-xr-x 11 hduser  hadoop   4096 Jul 30 18:23 .
drwxr-xr-x 12 root    root    4096 Jul 30 17:29 ..
drwxr-xr-x  2 hduser  hadoop   4096 Jan  8 2019 bin
drwxr-xr-x  3 hduser  hadoop   4096 Jan  8 2019 etc
-rw-rw-r--  1 hduser  hduser   51 Jul 30 18:23 hadoop-env.sh
drwxr-xr-x  3 hduser  hadoop   4096 Jul 30 17:18 hadoop_store
drwxr-xr-x  2 hduser  hadoop   4096 Jan  8 2019 include
drwxr-xr-x  3 hduser  hadoop   4096 Jan  8 2019 lib
drwxr-xr-x  4 hduser  hadoop   4096 Jan  8 2019 libexec
-rw-rw-r--  1 hduser  hadoop 150569 Oct 18 2018 LICENSE.txt
-rw-rw-r--  2 hduser  hduser   4096 Aug 10 2013 logs
-rw-rw-r--  1 hduser  hadoop 22125 Oct 18 2018 NOTICE.txt
-rw-rw-r--  1 hduser  hadoop 1361 Oct 19 2018 README.txt
drwxr-xr-x  3 hduser  hadoop   4096 Jan  8 2019 sbin
drwxr-xr-x  4 hduser  hadoop   4096 Jan  8 2019 share
```

Figure 9: Permission for Hadoop hduser

Step 9: Installation of Spark (Master and Slaves).

Command:

```
sudo wget https://mirrors.estointernet.in/apache/spark/spark-3.0.0/spark-3.0.0-bin-hadoop2.7.tgz
```

```
sudo tar -xzf spark-3.0.0-bin-hadoop2.7.tgz
```

```
Step 9. // Installation of Spark using below command. Perform this task in both master and slaves.
sudo wget https://mirrors.estointernet.in/apache/spark/spark-3.0.0/spark-3.0.0-bin-hadoop2.7.tgz
sudo tar -xzf spark-3.0.0-bin-hadoop2.7.tgz
Rename: sudo mv spark-3.0.0-bin-hadoop2.7.tgz spark
sudo chown -R hduser:hadoop /usr/local/spark
mv hadoop /usr/local/
//go in terminal (hduser) to verify our symlink redirected which mentioned above.
cd $SPARK_HOME // will direct to /usr/local/Spark
```

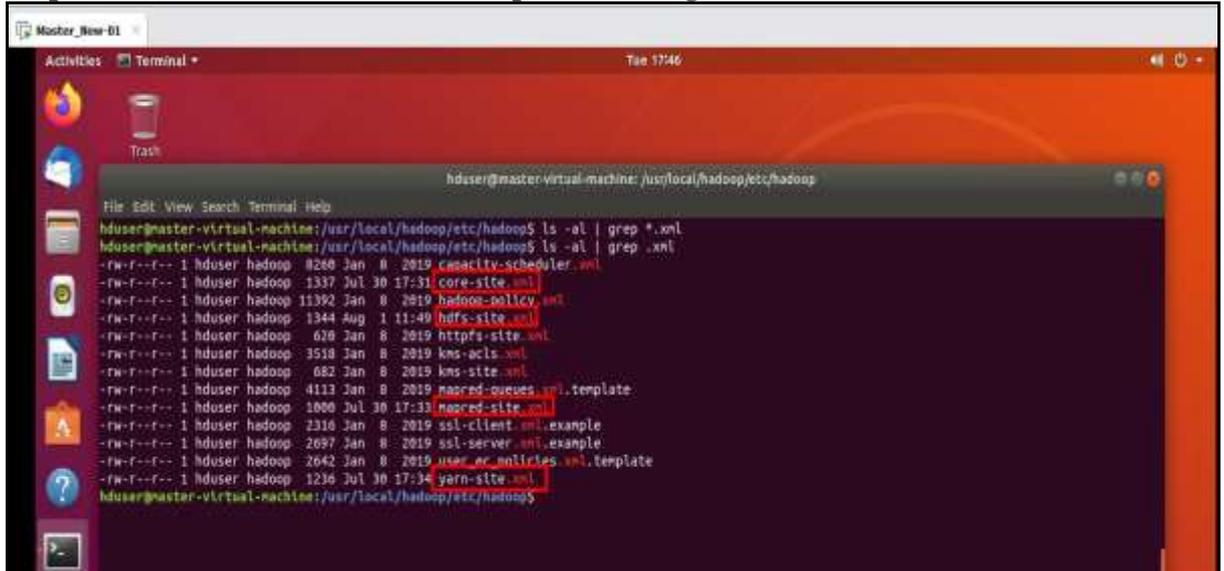


The screenshot shows a terminal window with the following output:

```
hduser@master-virtual-machine:/usr/local$ ls -al
total 337576
drwxr-xr-x 12 root    root    4096 Jul 30 17:29 .
drwxr-xr-x 11 root    root    4096 Feb  3 2020 ..
drwxr-xr-x  2 root    root    4096 Feb  3 2020 bin
drwxr-xr-x  2 root    root    4096 Feb  3 2020 etc
drwxr-xr-x  2 root    root    4096 Feb  3 2020 games
drwxr-xr-x 11 hduser  hadoop   4096 Jul 30 18:23 hadoop
-rw-rw-r--  1 root    root   34565475 Jan 21 2019 hadoop-2.7.0.tar.gz
drwxr-xr-x  4 root    root    4096 Aug  1 12:16 lib
lrwxrwxrwx  1 root    root      9 Jul 30 15:08 man -> share/man
drwxr-xr-x  2 root    root    4096 Feb  3 2020 sbin
drwxr-xr-x  6 root    root    4096 Feb  3 2020 share
drwxr-xr-x 14 hduser  hadoop   4096 Aug  1 12:33 spark
drwxr-xr-x  2 root    root    4096 Feb  3 2020 src
hduser@master-virtual-machine:/usr/local$
```

Figure 10: Spark Install

Step 10 to 14: Modification in Hadoop main config (xml) file.

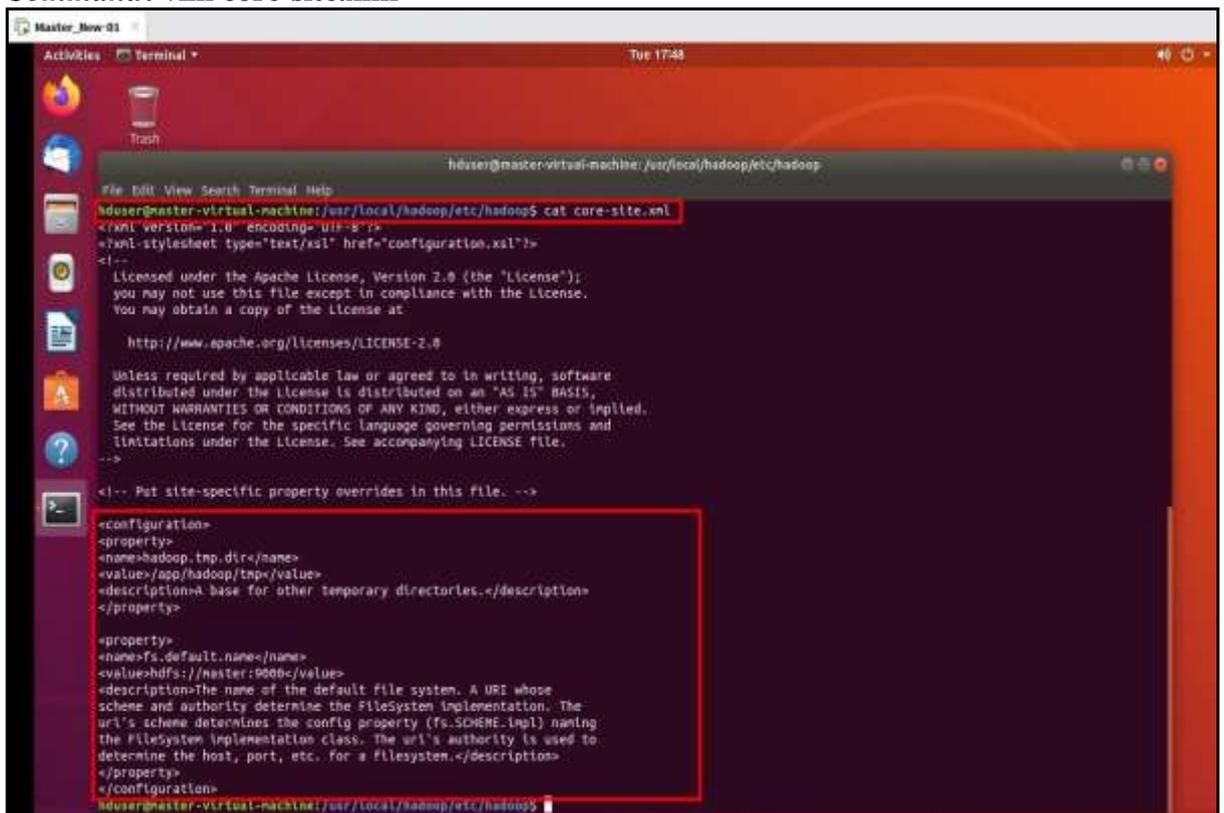


```
hduser@master-virtual-machine: /usr/local/hadoop/etc/hadoop
hduser@master-virtual-machine: /usr/local/hadoop/etc/hadoop$ ls -al | grep *.xml
-rw-r--r-- 1 hduser hadoop 8268 Jan 8 2019 capacity-scheduler.xml
-rw-r--r-- 1 hduser hadoop 1337 Jul 30 17:31 core-site.xml
-rw-r--r-- 1 hduser hadoop 11392 Jan 8 2019 hadoop-policy.xml
-rw-r--r-- 1 hduser hadoop 1344 Aug 1 11:49 hdfs-site.xml
-rw-r--r-- 1 hduser hadoop 678 Jan 8 2019 https-site.xml
-rw-r--r-- 1 hduser hadoop 3518 Jan 8 2019 kms-acls.xml
-rw-r--r-- 1 hduser hadoop 682 Jan 8 2019 kms-site.xml
-rw-r--r-- 1 hduser hadoop 4113 Jan 8 2019 mapred-queues.xml.template
-rw-r--r-- 1 hduser hadoop 1000 Jul 30 17:33 mapred-site.xml
-rw-r--r-- 1 hduser hadoop 2316 Jan 8 2019 ssl-client.xml.example
-rw-r--r-- 1 hduser hadoop 2697 Jan 8 2019 ssl-server.xml.example
-rw-r--r-- 1 hduser hadoop 2642 Jan 8 2019 user-er-policies.xml.template
-rw-r--r-- 1 hduser hadoop 1236 Jul 30 17:34 yarn-site.xml
hduser@master-virtual-machine: /usr/local/hadoop/etc/hadoop$
```

Figure 11: List of Hadoop Configuration File

Make changes in **core-site.xml** file under **/usr/local/Hadoop/etc/Hadoop** directory.

Command: `vim core-site.xml`



```
hduser@master-virtual-machine: /usr/local/hadoop/etc/hadoop$ cat core-site.xml
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the license at

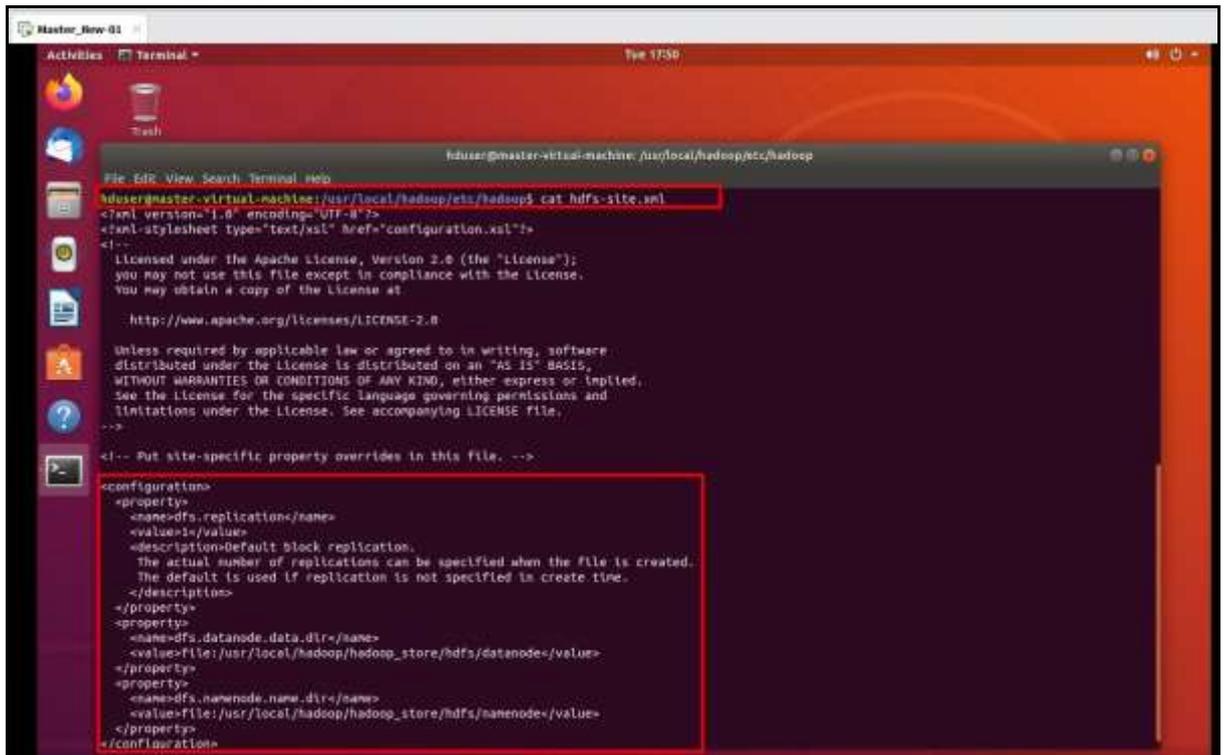
http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the license is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the license for the specific language governing permissions and
limitations under the license. See accompanying LICENSE file.
-->
<!-- Put site-specific property overrides in this file. -->
<configuration>
<property>
<name>hadoop.tmp.dir</name>
<value>app/hadoop/tmp</value>
<description>A base for other temporary directories.</description>
</property>
<property>
<name>fs.default.name</name>
<value>hdfs://master:9000</value>
<description>The name of the default file system. A URI whose
scheme and authority determine the FileSystem implementation. The
url's scheme determines the config property (fs.SCHEME.impl) naming
the FileSystem implementation class. The url's authority is used to
determine the host, port, etc. for a filesystem.</description>
</property>
</configuration>
hduser@master-virtual-machine: /usr/local/hadoop/etc/hadoop$
```

Figure 12: core-site.xml

Make changes in **hdfs-site.xml** file under /usr/local/Hadoop/etc/Hadoop directory.

Command: vim hdfs-site.xml



```
hdfsuser@master-virtual-machine: /usr/local/hadoop/etc/hadoop$ cat hdfs-site.xml
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

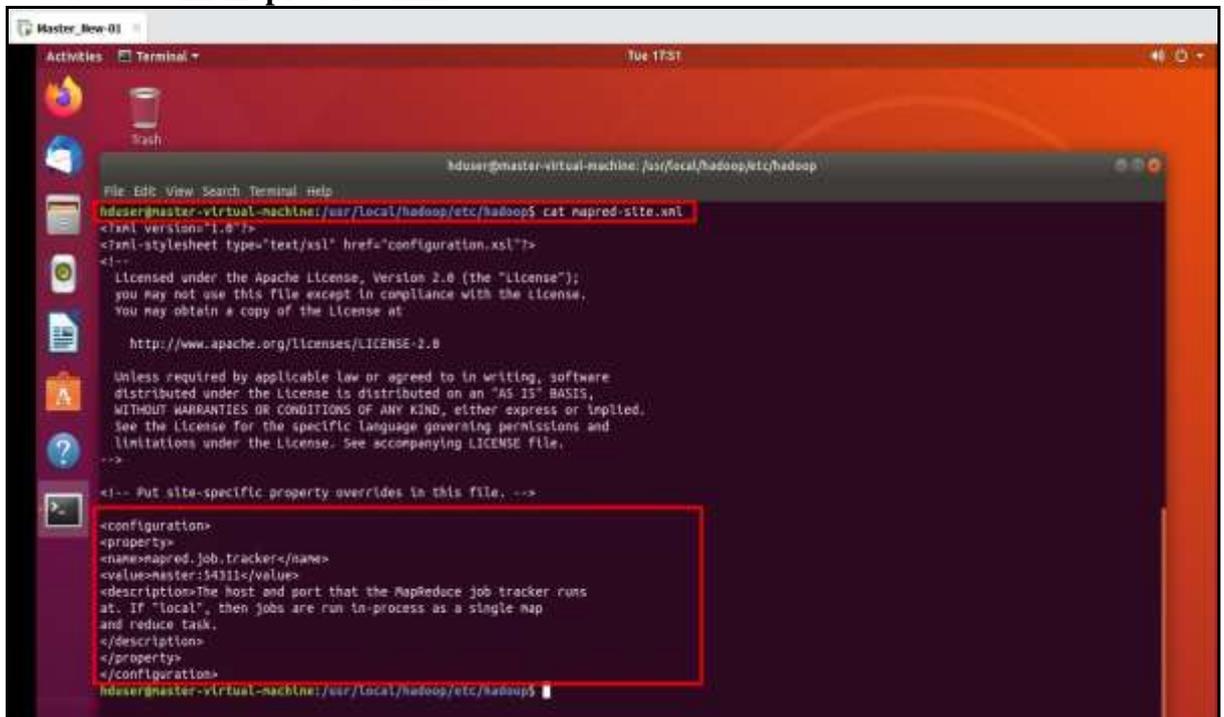
    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the license is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License. See accompanying LICENSE file.
-->
<!-- Put site-specific property overrides in this file. -->
<configuration>
  <property>
    <name>dfs.replication</name>
    <value>1</value>
    <description>Default block replication.
    The actual number of replications can be specified when the file is created.
    The default is used if replication is not specified in create time.
    </description>
  </property>
  <property>
    <name>dfs.datanode.data.dir</name>
    <value>file:/usr/local/hadoop/hadoop_store/hdfs/datanode</value>
  </property>
  <property>
    <name>dfs.namenode.name.dir</name>
    <value>file:/usr/local/hadoop/hadoop_store/hdfs/namenode</value>
  </property>
</configuration>
```

Figure 13: hdfs-site.xml

Make changes in **mapred-site.xml** file under /usr/local/Hadoop/etc/Hadoop directory.

Command: vim mapred-site.xml



```
hdfsuser@master-virtual-machine: /usr/local/hadoop/etc/hadoop$ cat mapred-site.xml
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the license.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License. See accompanying LICENSE file.
-->
<!-- Put site-specific property overrides in this file. -->
<configuration>
  <property>
    <name>mapred.job.tracker</name>
    <value>master:54311</value>
    <description>The host and port that the MapReduce job tracker runs
    at. If "local", then jobs are run in-process as a single map
    and reduce task.
    </description>
  </property>
</configuration>
hdfsuser@master-virtual-machine: /usr/local/hadoop/etc/hadoop$
```

Figure 14: mapred-site.xml

Make changes in **yarn-site.xml** file under /usr/local/Hadoop/etc/Hadoop directory.

Command: vim yarn-site.xml

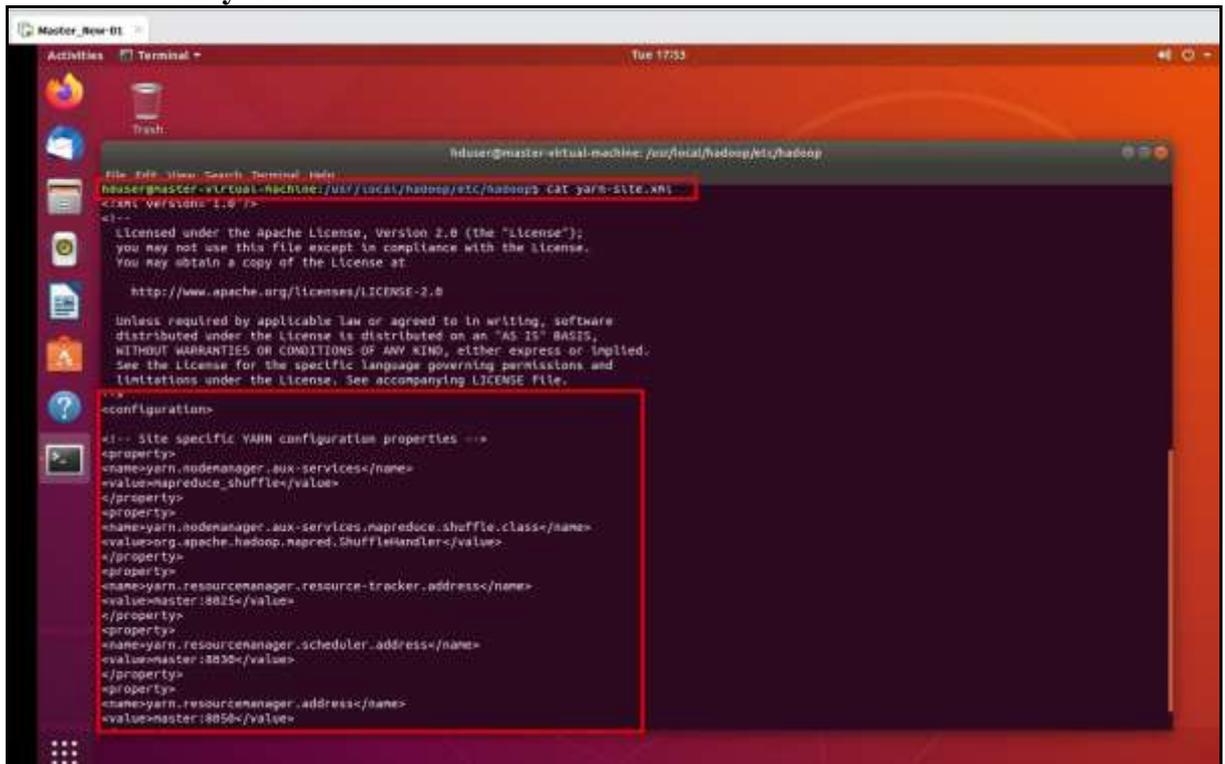


Figure 15: yarn-site.xml

5. Data Processing and Visualisation of Dataset

Data set contains 23 different types of attack which can be mapped in 4 various categories i.e Dos, User to root attack (U2R), Remote to local attack (R2L) and probing attack. Also, prepared column data which include protocol types and flag.



Figure 16: Data Processing

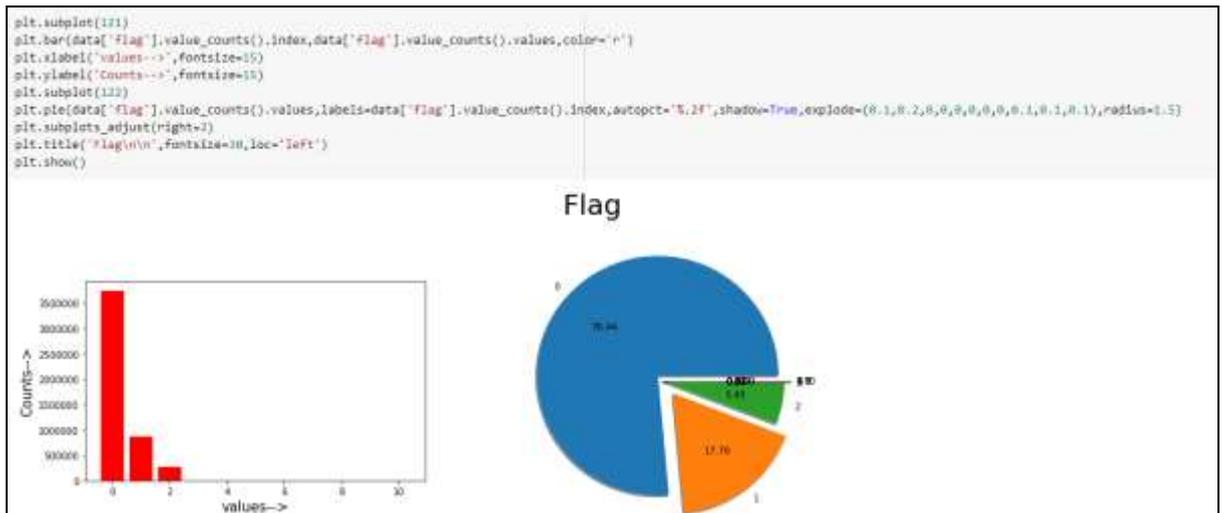


Figure 17: Data Visualisation

6. Implementation and Validation by Hadoop GUI portal

- Command for loading data in Hadoop

Command: `hadoop fs -put *.csv /IDS_data`

Figure 18: Command for loading data in Hadoop

- Starting Hadoop (HDFS)

Command: `start-dfs.sh`

Figure 19: Command for starting Hadoop

- Starting Spark for Master and Slaves (User: hduser)

Command: `start-master.sh`

Figure 20: Command for starting Spark

- **Hadoop GUI portal (http://ip_address:9870)**

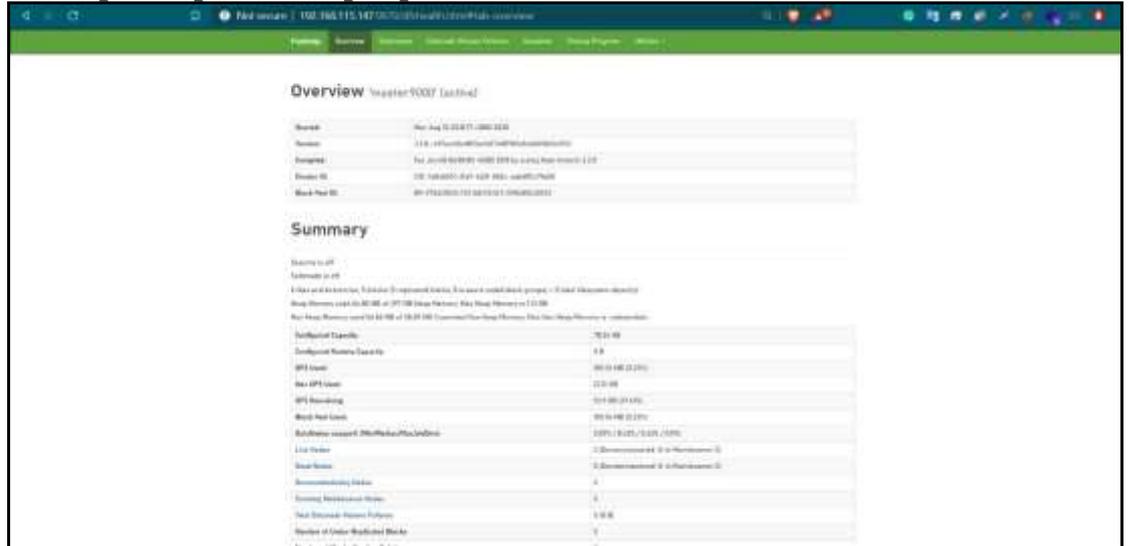


Figure 21: Hadoop GUI Portal

- **Data Node information of Slave01 and Slave02 (http://ip_address:9870)**

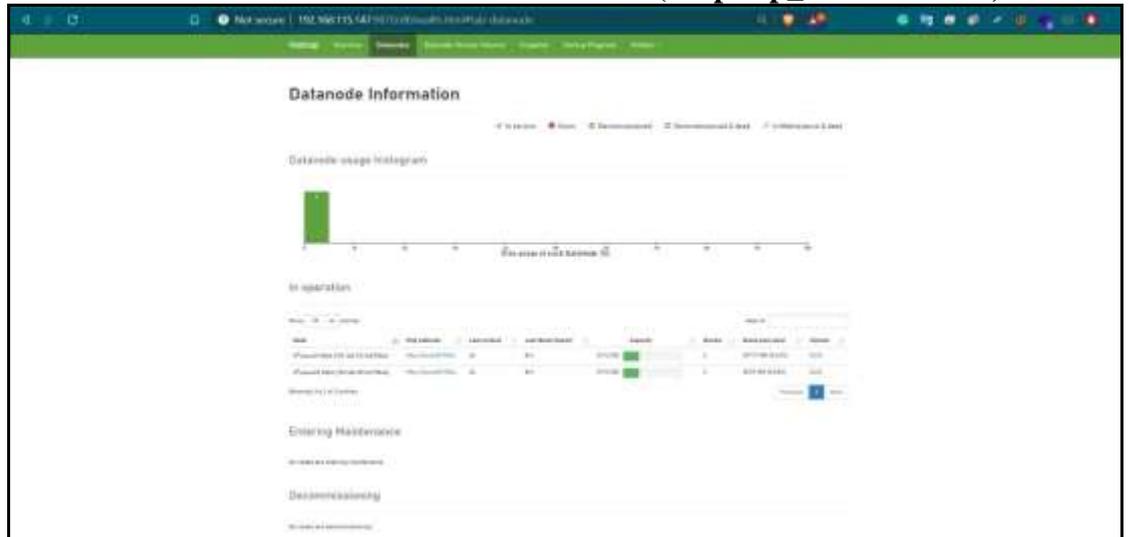


Figure 22: Data Node Information

- **Validating Spark GUI portal and Justification of Slave - Masters connection (http://ip_address:8080)**



Figure 23: Spark GUI Portal

- **Job completed by logistic regression algorithm. (http://ip_address:8080)**

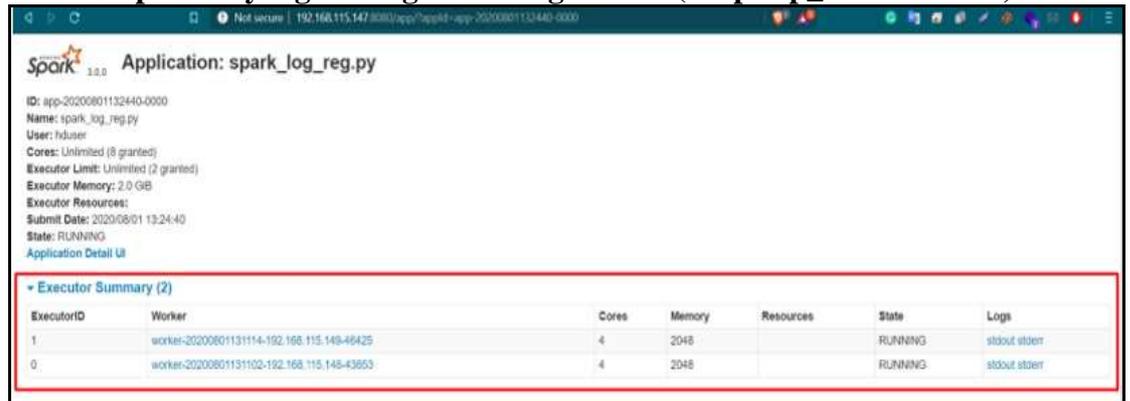


Figure 24: Spark Logistic Regression

7. Steps for executing command to analyse output.

- **Execute python file data_processing.py.**
python3 data_processing.py

After executing **data_processing.py**. It will create 1k, 10k, 1L and 10L csv. These files used in to analyse Logistic Regression, Naïve Bayes and Random Forest results.

List of Algorithm used:

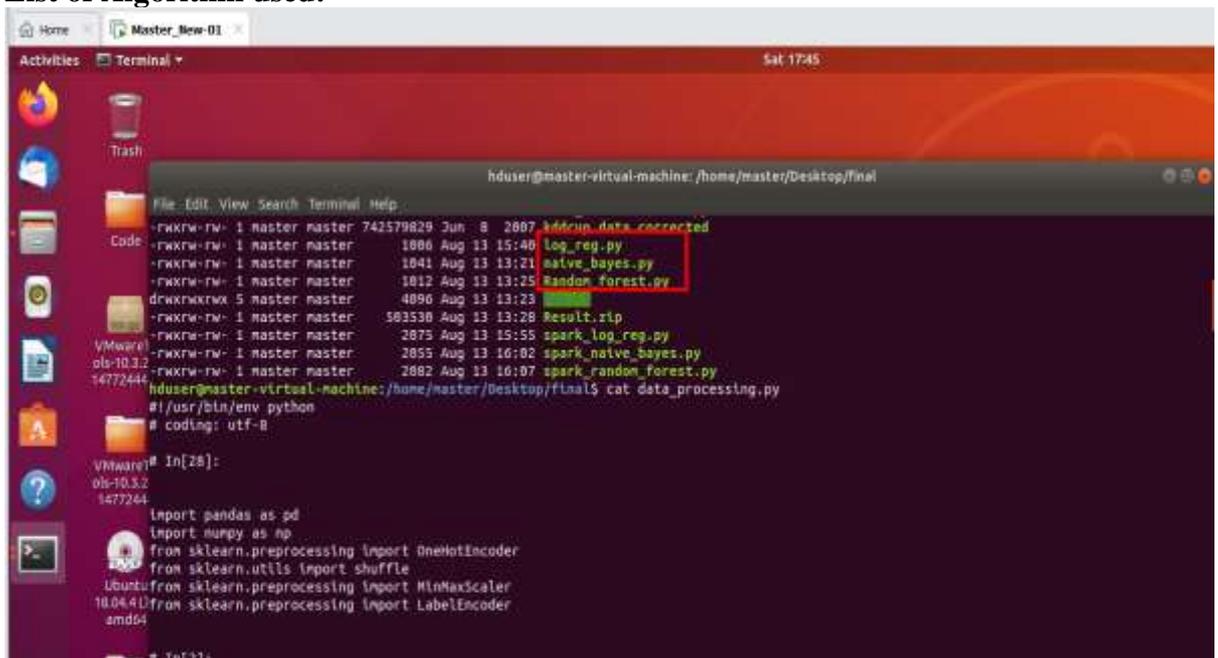


Figure 25: Algorithm Scripts

To perform every algorithm output, I must define 1k, 10k, 1L and 10L csv file to see result individually for both machine learning. As per below snapshot:

```

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, f1_score, precision_score, recall_score

data1 = pd.read_csv('1k_data.csv')
labels1 = data1['labels']
data1 = data1.drop('labels',axis=1)

X_train,X_test,Y_train,Y_test = train_test_split(data1,labels1,test_size = 30)

log_clf = LogisticRegression()
s_time = time.time()
log_clf.fit(X_train,Y_train)
times1 = time.time()-s_time

Y_pred = log_clf.predict(X_test)
accuracy1 = accuracy_score(Y_test,Y_pred)
precision1 = precision_score(Y_test,Y_pred,average='weighted')
recall1 = recall_score(Y_test,Y_pred,average='weighted')
fscore1 = f1_score(Y_test,Y_pred,average='weighted')

print('accuracy\n',accuracy1)
print('precision\n',precision1)
print('recall\n',recall1)
print('fscore\n',fscore1)
print('times\n',times1)

```

Figure 26: Define sample data file in Algorithm

```

hduser@master-virtual-machine:/home/master/Desktop/final$ start dfs.sh
Starting namenodes on [master]
Starting datanodes
Starting secondary namenodes [master-virtual-machine]
hduser@master-virtual-machine:/home/master/Desktop/final$ hadoop fs -put *.csv /I
OS_data/
put: '/OS_data/': No such file or directory
hduser@master-virtual-machine:/home/master/Desktop/final$ hadoop fs -put *.J

hduser@master-virtual-machine:/home/master/Desktop/final$ start-master.sh
starting org.apache.spark.deploy.master.Master, logging to /usr/local/spark/logs/
spark-hduser-org.apache.spark.deploy.master.Master-1-master-virtual-machine.out
hduser@master-virtual-machine:/home/master/Desktop/final$ start-slaves.sh
192.168.115.148: starting org.apache.spark.deploy.worker.Worker, logging to /usr/
local/spark/logs/spark-hduser-org.apache.spark.deploy.worker.Worker-1-slave01-vir
tual-machine.out
192.168.115.149: starting org.apache.spark.deploy.worker.Worker, logging to /usr/
local/spark/logs/spark-hduser-org.apache.spark.deploy.worker.Worker-1-slave02-vir
tual-machine.out
hduser@master-virtual-machine:/home/master/Desktop/final$

hduser@master-virtual-machine:/home/master/Desktop/final$ Hadoop fs -put *.csv /I
OS_data/
Hadoop: command not found or invalid
hduser@master-virtual-machine:/home/master/Desktop/final$ hadoop fs -put *.csv /I
OS_data/
put: '/OS_data/1k_data.csv': file exists
put: '/OS_data/1L_data.csv': file exists
hduser@master-virtual-machine:/home/master/Desktop/final$

```

Figure 27: All in one snapshot to start Hadoop and spark

- Execute below command to run algorithm in **tradition machine learning**.
python3 spark_log_reg.py
- Execute below command to run algorithm in **distributed machine learning**
spark-submit --master spark://192.168.115.147:7077 --executor-memory 2G spark_log_reg.py
- After that, analyse and compare the result for all algorithm i.e Logistic Regression, Naïve Bayes and Random Forest for 1k,10k,1L and 10L sample data.

References:

- [1] “Installing Python 3 on Linux — The Hitchhiker’s Guide to Python,” *docs.python-guide.org*. Available: docs.python-guide.org/starting/install3/linux/.
- [2] “Anaconda,” *Anaconda*, 2018. anaconda.com/
- [3] “VMware Maintenance,” *maintenance.vmware.com*. my.vmware.com/en/web/vmware/downloads/info/slug/desktop_end_user_computing/vmware_workstation_pro/15_0
- [4] “KDD Cup 1999 Data,” *kdd.ics.uci.edu*. kdd.ics.uci.edu/databases/kddcup99/kddcup99.html.