Configuration Manual

MSc Research Project
MSc In Cybersecurity
Practicum

Burhanuddin Shabbar
Student ID: 23142502

School of Computing
National College of Ireland

Supervisor: Khadija Hafeez

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Burhanuddin Shabbar…………………………………………………………………………
Student ID:	23142502………………………………………………………………………………….……
Programme:	MSc In Cybersecurity……………………………… **Year:** 2024……………….….
Module:	Practicum.…………………………………………………………………………….………
Lecturer:	Khadija Hafeez………………………………………………………………………………
Submission Due Date:	12/08/2024……………………………………………………………………………..………
Project Title:	Detection of blackhole and DDoS attack in 5g VANET using multiple machine learning algorithms.
Word Count:	812 words ………………… **Page Count:** 15 pages ………………..……..………

Signature: Burhanuddin Shabbar…………………….………………………………………………

Date: 12/08/2024………………………………………………………………………………………

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
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.	□

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

Configuration Manual

Burhanuddin Shabbar
23142502

1 INTRODUCTION:

The configuration manual explains the requirement to create the environment, implementation steps, necessary hardware, software and code snippets used for completion of research work. The main purpose of this manual is to demonstrate step by step coding procedure taken for this project. It will help to replicate and verify the results in future.

2 SOFTWARE AND HARDWARE REQUIREMENTS

2.1 Hardware requirements:

Following are the hardware requirements.

- CPU: Intel core i5 or i7 processor.
- 16 GB RAM
- Storage: 512 SSD

2.2 Software requirements:

Following are the software requirements.

- Windows 11
- Ubuntu 20.04
- OMNET++ version 5.6.2
- Venis version 5.0
- SUMO version 1.0.1
- Python 3.10

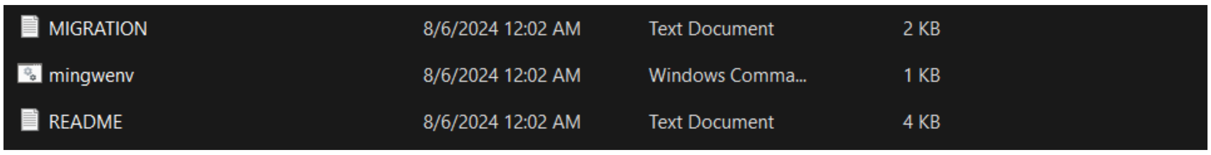
2.3 Python Libraries:

Following are the python libraries that are required to run the AI part of the project.

- Numpy
- Pandas
- Scikit-learn
- Matplotlib
- TensorFlow
- XGBOOST
- Datetime

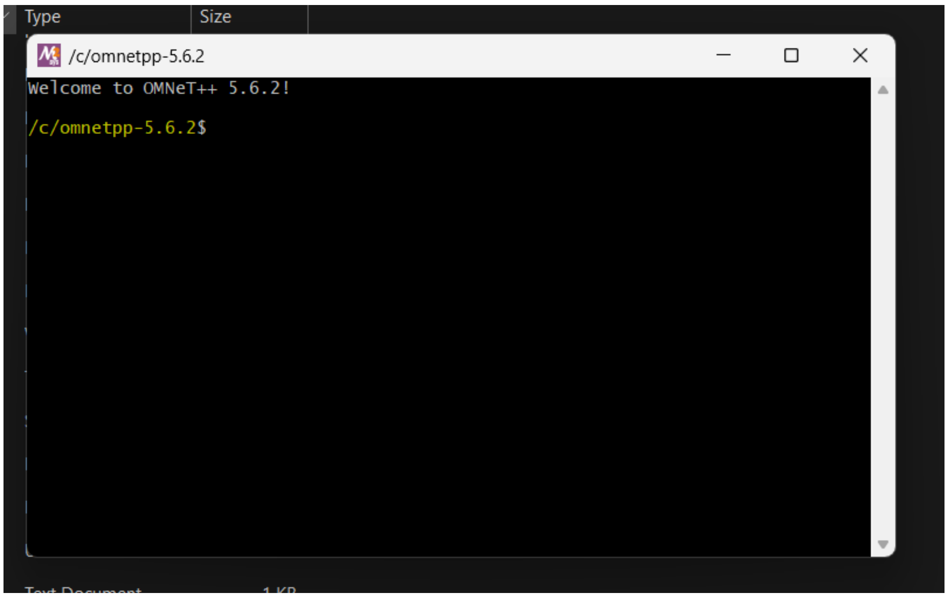
3 SETTING UP THE SIMULATION

- First go to the folder where you have installed the simulator. Then find mingwenv file and run as administrator.



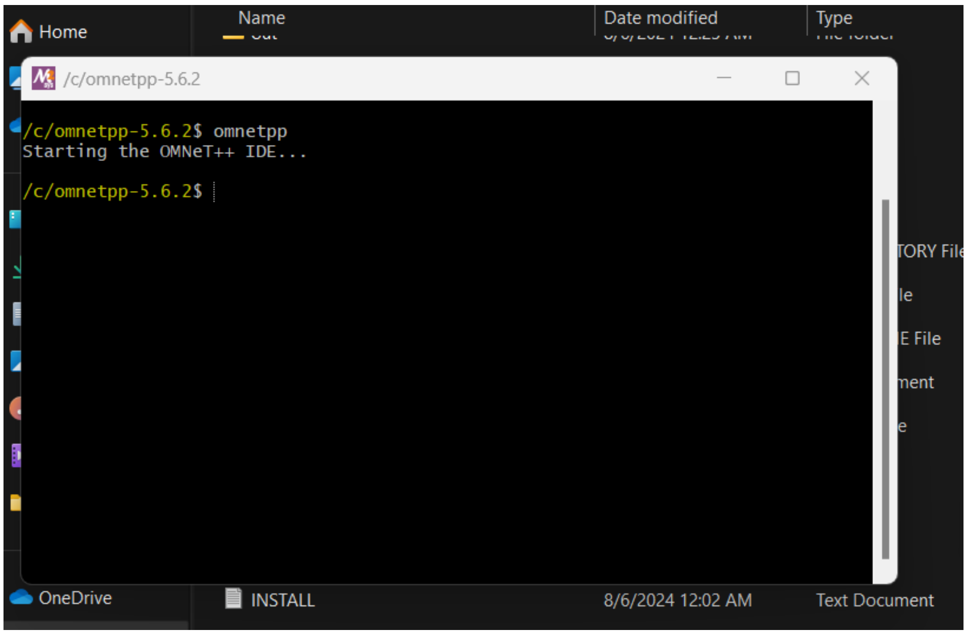
MIGRATION	8/6/2024 12:02 AM	Text Document	2 KB
mingwenv	8/6/2024 12:02 AM	Windows Comma...	1 KB
README	8/6/2024 12:02 AM	Text Document	4 KB

- A popup will open like this



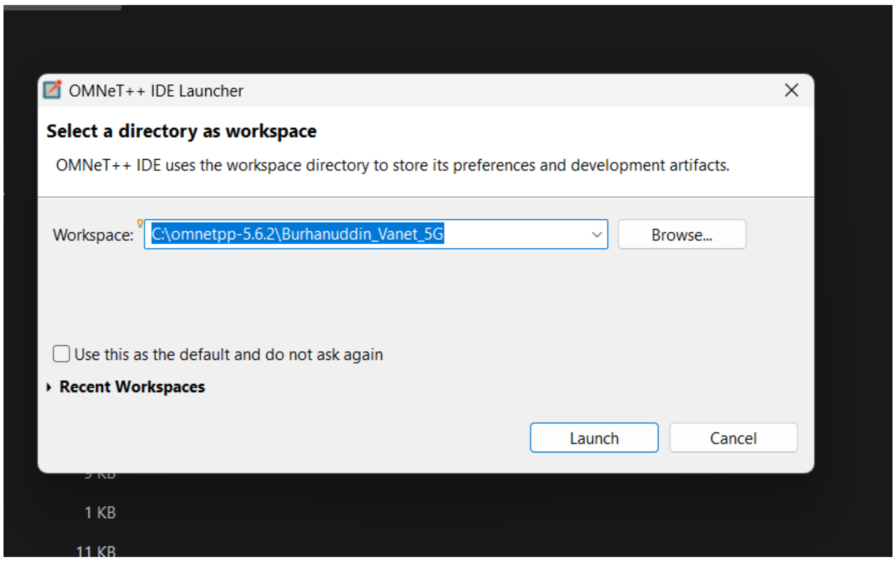
```
Welcome to OMNeT++ 5.6.2!

/c/omnetpp-5.6.2$
```

- Here type omnetpp and hit enter



```
/c/omnetpp-5.6.2$ omnetpp
Starting the OMNeT++ IDE...

/c/omnetpp-5.6.2$
```

- This popup will show up and now select the workspace you want to open. In our case it is the Burhanuddin_Vanet_5G and then hit launch



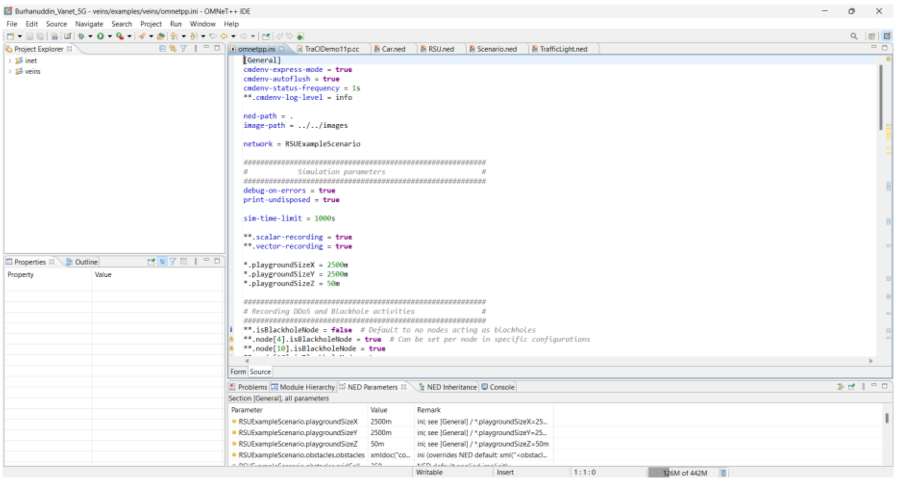
OMNeT++ IDE Launcher

Select a directory as workspace

OMNeT++ IDE uses the workspace directory to store its preferences and development artifacts.

Workspace: C:\omnetpp-5.6.2\Burhanuddin_Vanet_5G

- The simulator looks like this



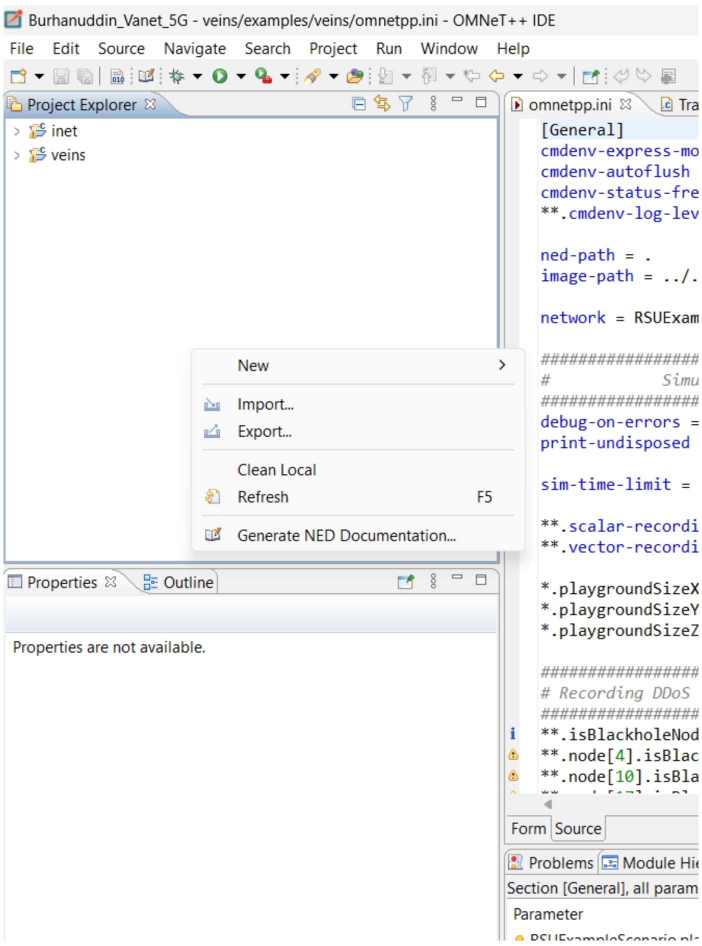
```
[General]
cmdenv-express-mode = true
cmdenv-autoflush = true
cmdenv-status-frequency = 1s
**.cmdenv-log-level = info

ned-path = .
image-path = ../../images

network = RSUExampleScenario

##########################################################
#            Simulation parameters                       #
##########################################################
debug-on-errors = true
print-undisposed = true

sim-time-limit = 1000s

**.scalar-recording = true
**.vector-recording = true

*.playgroundSizeX = 2500m
*.playgroundSizeY = 2500m
*.playgroundSizeZ = 50m

##########################################################
# Recording DDoS and Blackhole activities                #
##########################################################
**.isBlackholeNode = false  # Default to no nodes acting as blackholes
**.node[4].isBlackholeNode = true  # Can be set per node in specific configurations
**.node[10].isBlackholeNode = true
```

Section [General], all parameters

Parameter	Value	Remark
RSUExampleScenario.playgroundSizeX	2500m	ini; see [General] / *.playgroundSizeX=25...
RSUExampleScenario.playgroundSizeY	2500m	ini; see [General] / *.playgroundSizeY=25...
RSUExampleScenario.playgroundSizeZ	50m	ini; see [General] / *.playgroundSizeZ=50m
RSUExampleScenario.obstacles.obstacles	xmldoc("co...	ini (overrides NED default: xml("<obstacl...

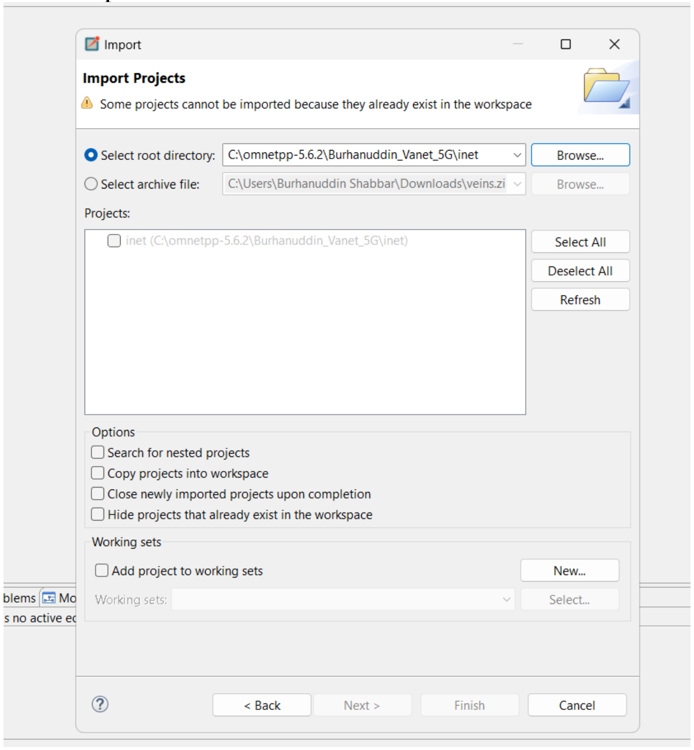
- First step is to install inet and viens frameworks. This is done by downloading both the frameworks from their official site and import it into the OMNET++ simulator.

- Right click the taskbar to the left it will give you the option to import


Burhanuddin_Vanet_5G - veins/examples/veins/omnetpp.ini - OMNeT++ IDE
File Edit Source Navigate Search Project Run Window Help
Project Explorer
inet
veins
New
Import...
Export...
Clean Local
Refresh F5
Generate NED Documentation...
Properties
Outline
Properties are not available.
omnetpp.ini
Form
Source
Problems
Module Hie
Section [General], all param
Parameter

```
[General]
cmdenv-express-mo
cmdenv-autoflush
cmdenv-status-fre
**.cmdenv-log-lev

ned-path = .
image-path = ../.

network = RSUExam

#################
#            Simu
#################
debug-on-errors =
print-undisposed

sim-time-limit =

**.scalar-recordi
**.vector-recordi

*.playgroundSizeX
*.playgroundSizeY
*.playgroundSizeZ

#################
# Recording DDoS
#################
**.isBlackholeNod
**.node[4].isBlac
**.node[10].isBla
```

- First import inet



Import Projects

Some projects cannot be imported because they already exist in the workspace

[✓ Select root directory:] C:\omnetpp-5.6.2\Burhanuddin_Vanet_5G\inet

Select archive file: C:\Users\Burhanuddin Shabbar\Downloads\veins.zi

Projects:

inet (C:\omnetpp-5.6.2\Burhanuddin_Vanet_5G\inet)

Options

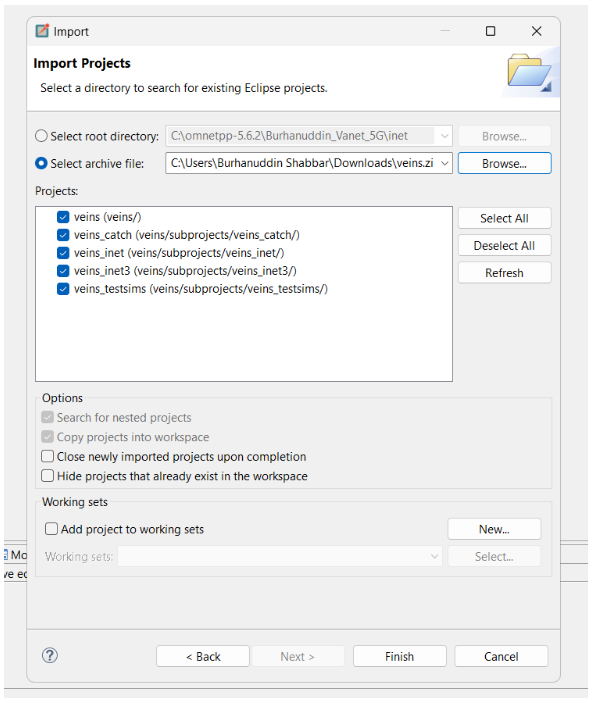
- Search for nested projects
- Copy projects into workspace
- Close newly imported projects upon completion
- Hide projects that already exist in the workspace

Working sets

- Add project to working sets

Working sets:

- After importing inet built the project
- After the built is completed. Now import viens into the simulator.



Import

Import Projects

Select a directory to search for existing Eclipse projects.

Select root directory: C:\omnetpp-5.6.2\Burhanuddin_Vanet_5G\inet

[✓ Select archive file:] C:\Users\Burhanuddin Shabbar\Downloads\veins.zi

Projects:

- [✓] veins (veins/)
- [✓] veins_catch (veins/subprojects/veins_catch/)
- [✓] veins_inet (veins/subprojects/veins_inet/)
- [✓] veins_inet3 (veins/subprojects/veins_inet3/)
- [✓] veins_testsims (veins/subprojects/veins_testsims/)

Options

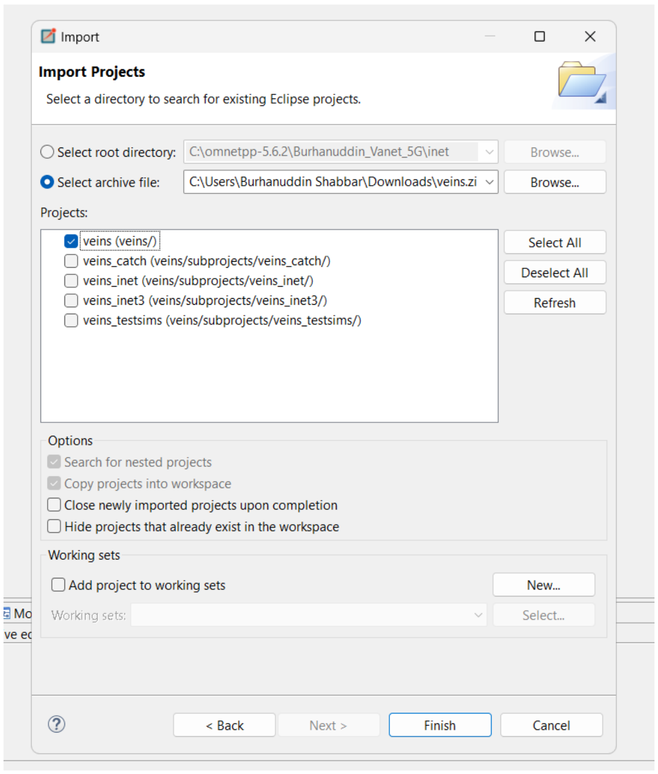
- [✓] Search for nested projects
- [✓] Copy projects into workspace
- [ ] Close newly imported projects upon completion
- [ ] Hide projects that already exist in the workspace

Working sets

- [ ] Add project to working sets

Working sets:

- Now unselect the other projects that is associated with veins. After doing that hit finish.



Import

Import Projects

Select a directory to search for existing Eclipse projects.

Select root directory: C:\omnetpp-5.6.2\Burhanuddin_Vanet_5G\inet

[✓ Select archive file:] C:\Users\Burhanuddin Shabbar\Downloads\veins.zi

Projects:

- [✓] veins (veins/)
- [ ] veins_catch (veins/subprojects/veins_catch/)
- [ ] veins_inet (veins/subprojects/veins_inet/)
- [ ] veins_inet3 (veins/subprojects/veins_inet3/)
- [ ] veins_testsims (veins/subprojects/veins_testsims/)

Options

- [✓] Search for nested projects
- [✓] Copy projects into workspace
- [ ] Close newly imported projects upon completion
- [ ] Hide projects that already exist in the workspace

Working sets

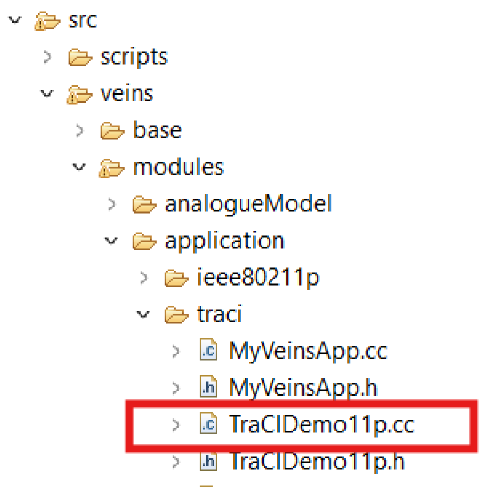
- [ ] Add project to working sets

Working sets:

- Now select veins and built the viens framework.
- Setting up the simulation is completed.

4 IMPLEMENTATION OF THE ATTACK:

The following file contains the attack code:



```
src
  scripts
  veins
    base
    modules
      analogueModel
      application
        ieee80211p
        traci
          MyVeinsApp.cc
          MyVeinsApp.h
          TraCIDemo11p.cc
          TraCIDemo11p.h
```

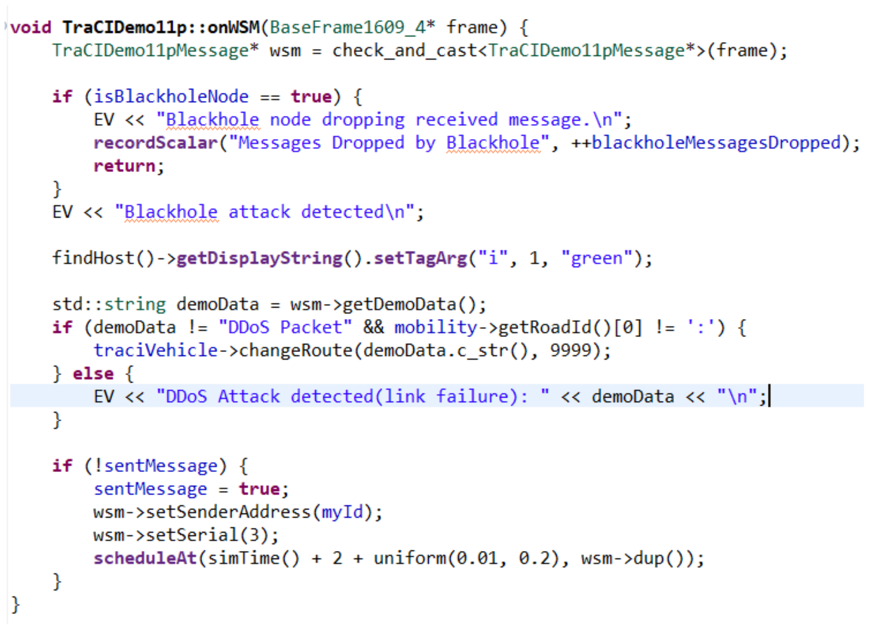
4.1 DDoS Attack:

The following is the code for DDoS attack.

```
void TraCIDemo11p::handleSelfMsg(cMessage* msg) {
    if (msg->isName("DDoSAttack")) {
        int attackMessages = uniform(0, 4);
        double fakePacketSize = uniform(512, 2048);
        double fakeTransmissionTime = uniform(0.005, 0.02);

        for (int i = 0; i < attackMessages; ++i) {
            TraCIDemo11pMessage* attackMsg = new TraCIDemo11pMessage();
            populateWSM(attackMsg);
            attackMsg->setDemoData("DDoS Packet");
            sendDown(attackMsg->dup());

            // Record additional attack parameters
            recordScalar("Packet Size (bytes)", fakePacketSize);
            recordScalar("Transmission Time (s)", fakeTransmissionTime);

            // Increment attack statistics
            attackMessagesSent++;
        }

        EV << "DDoS attack executed, sending " << attackMessages << " messages.\n";
        recordScalar("DDoS Attack Messages Sent", attackMessagesSent);

        // Schedule the next attack
        scheduleAt(simTime() + uniform(1, 5), msg);
    } else if (auto* wsm = dynamic_cast<TraCIDemo11pMessage*>(msg)) {
        sendDown(wsm->dup());
        wsm->setSerial(wsm->getSerial() + 1);
        if (wsm->getSerial() >= 3) {
            stopService();
            delete wsm;
        } else {
            scheduleAt(simTime() + 1, wsm);
        }
    } else {
        DemoBaseApplLayer::handleSelfMsg(msg);
    }
}
```

4.2 Blackhole Attack:

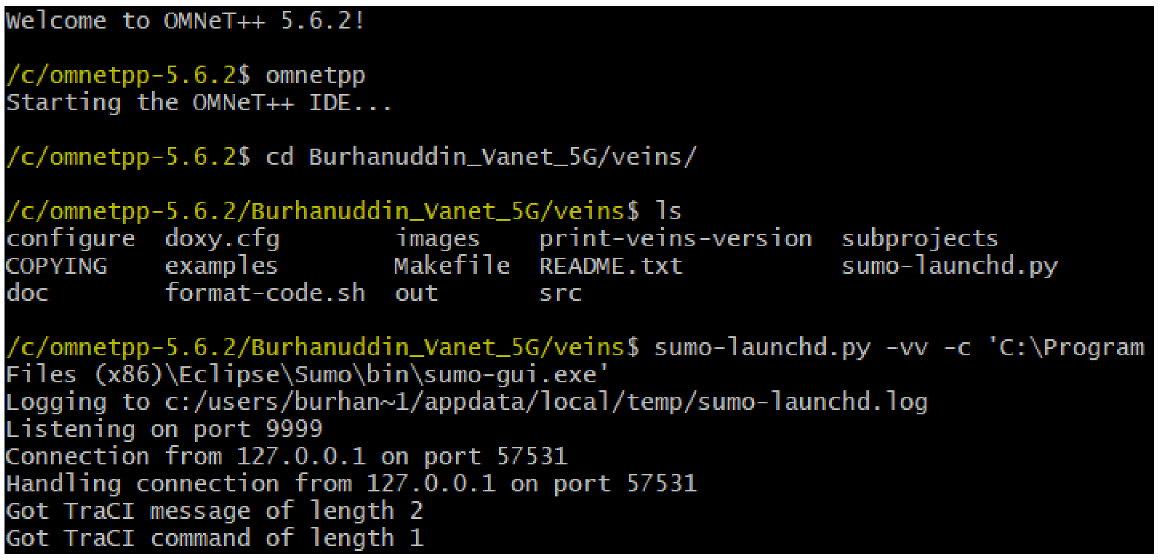
The Following is the code for blackhole attack.



```
void TraCIDemo11p::onWSM(BaseFrame1609_4* frame) {
    TraCIDemo11pMessage* wsm = check_and_cast<TraCIDemo11pMessage*>(frame);

    if (isBlackholeNode == true) {
        EV << "Blackhole node dropping received message.\n";
        recordScalar("Messages Dropped by Blackhole", ++blackholeMessagesDropped);
        return;
    }
    EV << "Blackhole attack detected\n";

    findHost()->getDisplayString().setTagArg("i", 1, "green");

    std::string demoData = wsm->getDemoData();
    if (demoData != "DDoS Packet" && mobility->getRoadId()[0] != ':') {
        traciVehicle->changeRoute(demoData.c_str(), 9999);
    } else {
        EV << "DDoS Attack detected(link failure): " << demoData << "\n";
    }

    if (!sentMessage) {
        sentMessage = true;
        wsm->setSenderAddress(myId);
        wsm->setSerial(3);
        scheduleAt(simTime() + 2 + uniform(0.01, 0.2), wsm->dup());
    }
}
```

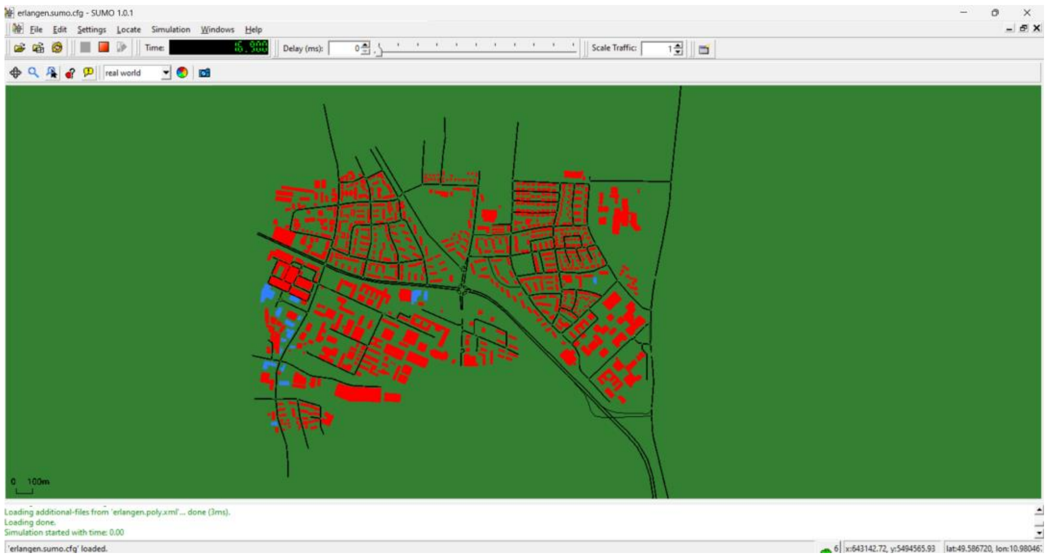
5 Run the simulation:

- To run the simulation first we have to launch sumo-gui.



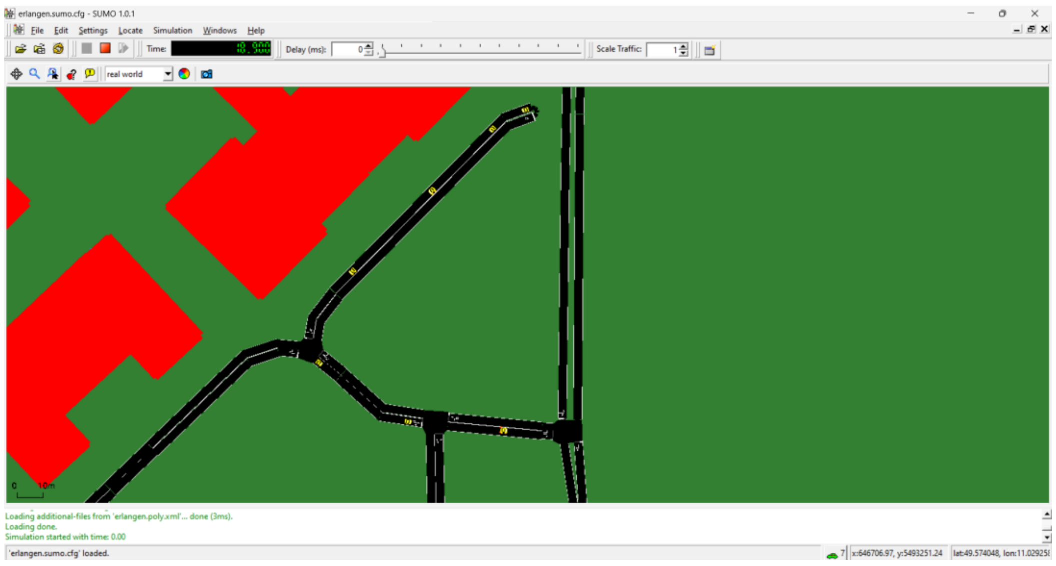
```
Welcome to OMNeT++ 5.6.2!

/c/omnetpp-5.6.2$ omnetpp
Starting the OMNeT++ IDE...

/c/omnetpp-5.6.2$ cd Burhanuddin_Vanet_5G/veins/

/c/omnetpp-5.6.2/Burhanuddin_Vanet_5G/veins$ ls
configure  doxy.cfg        images    print-veins-version  subprojects
COPYING    examples        Makefile  README.txt           sumo-launchd.py
doc        format-code.sh  out       src

/c/omnetpp-5.6.2/Burhanuddin_Vanet_5G/veins$ sumo-launchd.py -vv -c 'C:\Program
Files (x86)\Eclipse\Sumo\bin\sumo-gui.exe'
Logging to c:/users/burhan~1/appdata/local/temp/sumo-launchd.log
Listening on port 9999
Connection from 127.0.0.1 on port 57531
Handling connection from 127.0.0.1 on port 57531
Got TraCI message of length 2
Got TraCI command of length 1
```

- After the sumo is launch. Hit the play button on the OMNET++
- Sumo Gui will be launched.



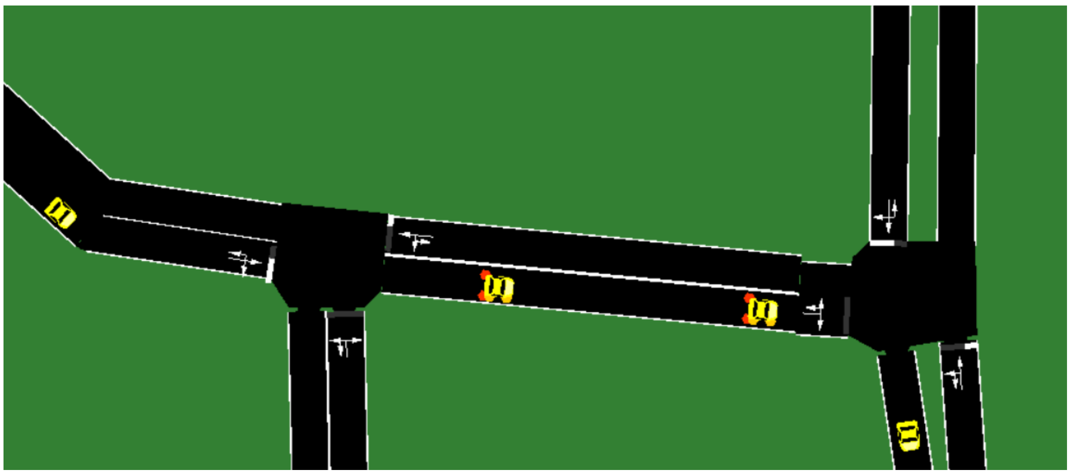
```
Loading additional-files from 'erlangen.poly.xml'... done (3ms).
Loading done.
Simulation started with time: 0.00
'erlangen.sumo.cfg' loaded.
6 | x:643142.72, y:5494565.93 | lat:49.586720, lon:10.98046
```

- The vehicles are automatically moving within the map. Some of these vehicles are malicious.

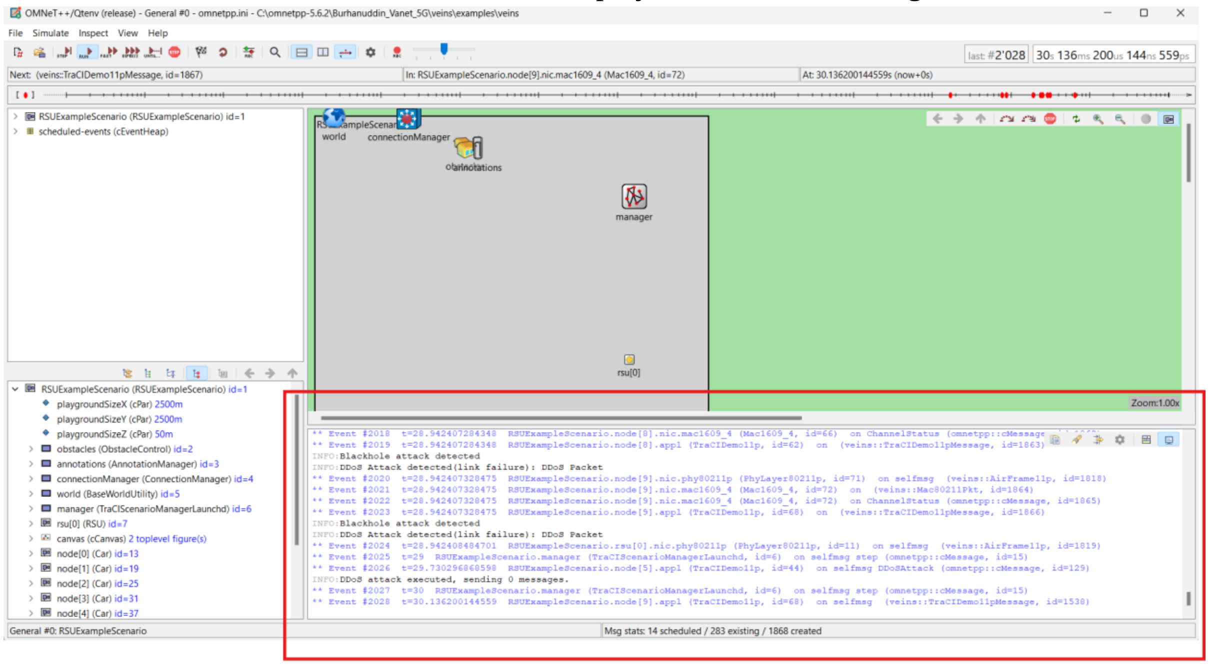


```
Loading additional-files from 'erlangen.poly.xml'... done (3ms).
Loading done.
Simulation started with time: 0.00
'erlangen.sumo.cfg' loaded.
7 | x:646706.97, y:5493251.24 | lat:49.574048, lon:11.02925
```

- The vehicles will red nodes are malicious ones.



- When ever the attack is launched it is been displayed on the console.log



OMNeT++/Qtenv (release) - General #0 - omnetpp.ini - C:\omnetpp-5.6.2\Burhanuddin_Vanet_5G\veins\examples\veins

Next: (veins::TraCIDemo11pMessage, id=1867) | In: RSUExampleScenario.node[9].nic.mac1609_4 (Mac1609_4, id=72) | At: 30.136200144559s (now+0s)

last: #2'028 | 30s 136ms 200us 144ns 559ps

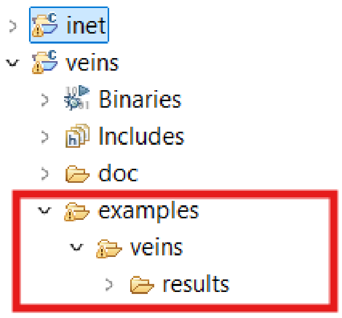
```
RSUExampleScenario (RSUExampleScenario) id=1
    playgroundSizeX (cPar) 2500m
    playgroundSizeY (cPar) 2500m
    playgroundSizeZ (cPar) 50m
  > obstacles (ObstacleControl) id=2
  > annotations (AnnotationManager) id=3
  > connectionManager (ConnectionManager) id=4
  > world (BaseWorldUtility) id=5
  > manager (TraCIScenarioManagerLaunchd) id=6
  > rsu[0] (RSU) id=7
  > canvas (cCanvas) 2 toplevel figure(s)
  > node[0] (Car) id=13
  > node[1] (Car) id=19
  > node[2] (Car) id=25
  > node[3] (Car) id=31
  > node[4] (Car) id=37
```

```
** Event #2018  t=28.942407284348  RSUExampleScenario.node[8].nic.mac1609_4 (Mac1609_4, id=66)  on ChannelStatus (omnetpp::cMessage
** Event #2019  t=28.942407284348  RSUExampleScenario.node[8].appl (TraCIDemo11p, id=62)  on  (veins::TraCIDemo11pMessage, id=1863)
INFO:Blackhole attack detected
INFO:DDoS Attack detected(link failure): DDoS Packet
** Event #2020  t=28.942407328475  RSUExampleScenario.node[9].nic.phy80211p (PhyLayer80211p, id=71)  on selfmsg  (veins::AirFrame11p, id=1818)
** Event #2021  t=28.942407328475  RSUExampleScenario.node[9].nic.mac1609_4 (Mac1609_4, id=72)  on  (veins::Mac80211Pkt, id=1864)
** Event #2022  t=28.942407328475  RSUExampleScenario.node[9].nic.mac1609_4 (Mac1609_4, id=72)  on ChannelStatus (omnetpp::cMessage, id=1865)
** Event #2023  t=28.942407328475  RSUExampleScenario.node[9].appl (TraCIDemo11p, id=68)  on  (veins::TraCIDemo11pMessage, id=1866)
INFO:Blackhole attack detected
INFO:DDoS Attack detected(link failure): DDoS Packet
** Event #2024  t=28.942408484701  RSUExampleScenario.rsu[0].nic.phy80211p (PhyLayer80211p, id=11)  on selfmsg  (veins::AirFrame11p, id=1819)
** Event #2025  t=29  RSUExampleScenario.manager (TraCIScenarioManagerLaunchd, id=6)  on selfmsg step (omnetpp::cMessage, id=15)
** Event #2026  t=29.730296868598  RSUExampleScenario.node[5].appl (TraCIDemo11p, id=44)  on selfmsg DDoSAttack (omnetpp::cMessage, id=129)
INFO:DDoS attack executed, sending 0 messages.
** Event #2027  t=30  RSUExampleScenario.manager (TraCIScenarioManagerLaunchd, id=6)  on selfmsg step (omnetpp::cMessage, id=15)
** Event #2028  t=30.136200144559  RSUExampleScenario.node[9].appl (TraCIDemo11p, id=68)  on selfmsg  (veins::TraCIDemo11pMessage, id=1538)
```

General #0: RSUExampleScenario | Msg stats: 14 scheduled / 283 existing / 1868 created

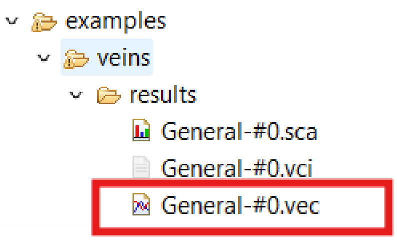
6 RESULTS OF ATTACK SIMULATION:

- The results of the attacks are being stored in the results folder.



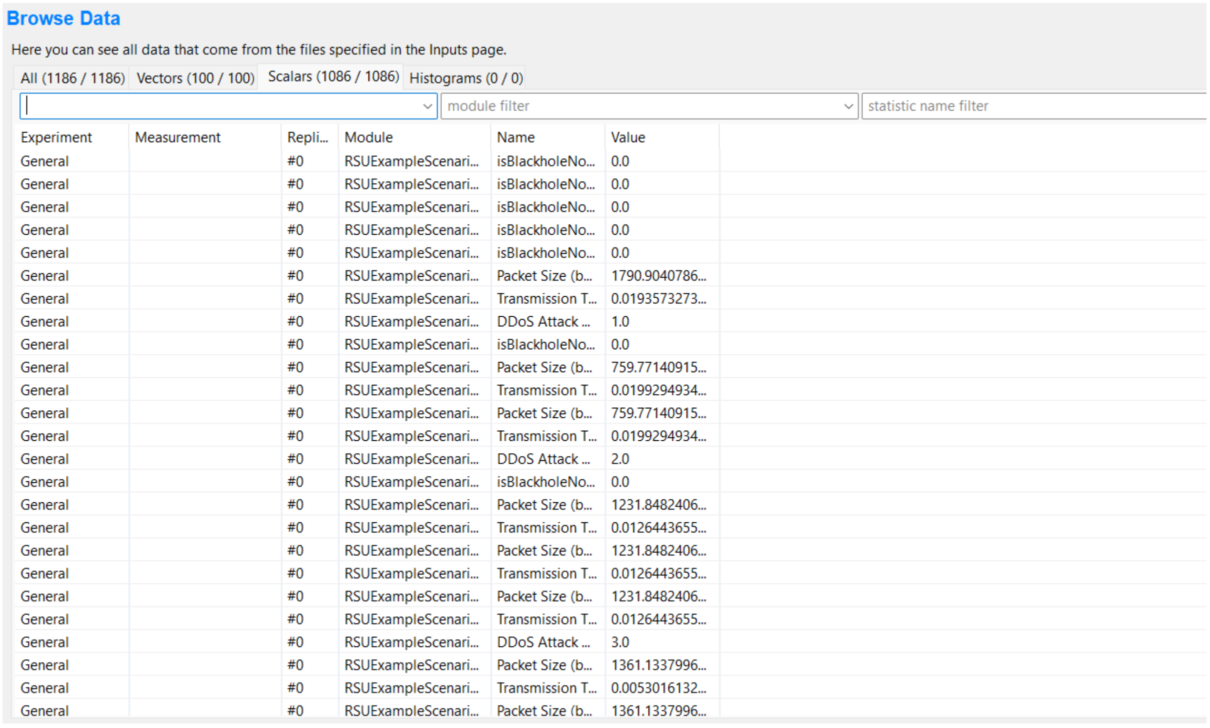
```
> inet
v veins
    > Binaries
    > Includes
    > doc
    v examples
        v veins
            > results
```

- The simulation generates a new dataset each time it runs, replacing the previous set of findings.
- This is the file where the data is generated



```
v examples
    v veins
        v results
            General-#0.sca
            General-#0.vci
            General-#0.vec
```

- So, this is what data set looks like



Browse Data

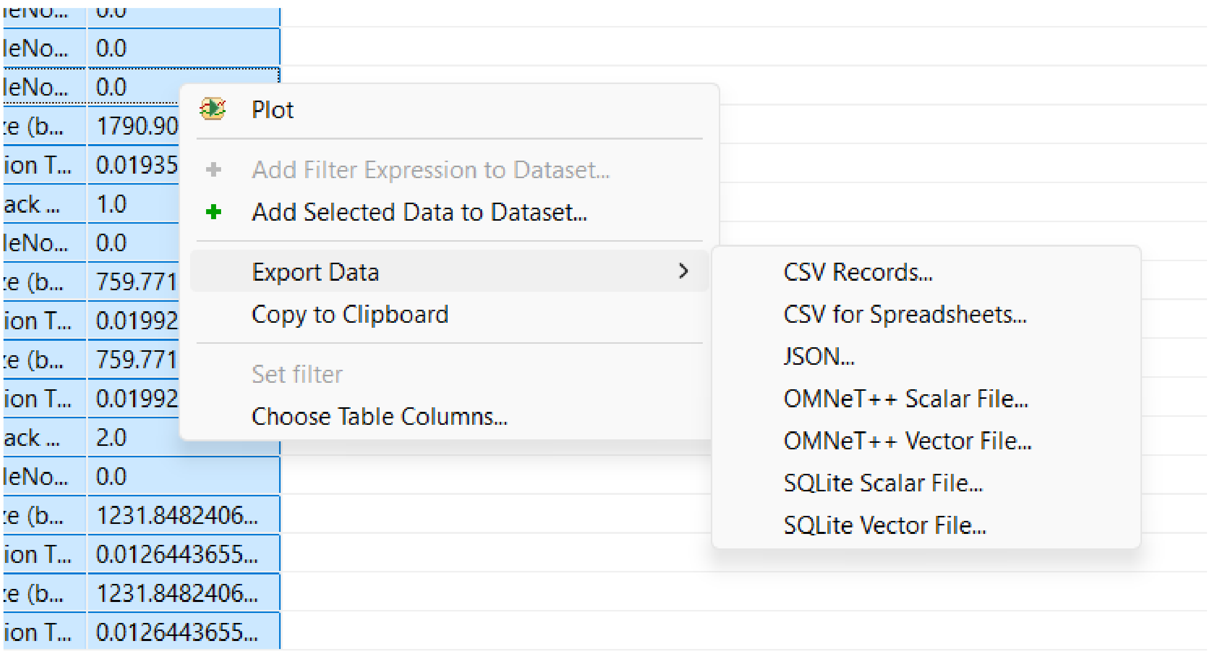
Here you can see all data that come from the files specified in the Inputs page.

All (1186 / 1186) | Vectors (100 / 100) | Scalars (1086 / 1086) | Histograms (0 / 0)

module filter | statistic name filter

Experiment	Measurement	Repli...	Module	Name	Value
General		#0	RSUExampleScenari...	isBlackholeNo...	0.0
General		#0	RSUExampleScenari...	isBlackholeNo...	0.0
General		#0	RSUExampleScenari...	isBlackholeNo...	0.0
General		#0	RSUExampleScenari...	isBlackholeNo...	0.0
General		#0	RSUExampleScenari...	isBlackholeNo...	0.0
General		#0	RSUExampleScenari...	Packet Size (b...	1790.9040786...
General		#0	RSUExampleScenari...	Transmission T...	0.0193573273...
General		#0	RSUExampleScenari...	DDoS Attack ...	1.0
General		#0	RSUExampleScenari...	isBlackholeNo...	0.0
General		#0	RSUExampleScenari...	Packet Size (b...	759.77140915...
General		#0	RSUExampleScenari...	Transmission T...	0.0199294934...
General		#0	RSUExampleScenari...	Packet Size (b...	759.77140915...
General		#0	RSUExampleScenari...	Transmission T...	0.0199294934...
General		#0	RSUExampleScenari...	DDoS Attack ...	2.0
General		#0	RSUExampleScenari...	isBlackholeNo...	0.0
General		#0	RSUExampleScenari...	Packet Size (b...	1231.8482406...
General		#0	RSUExampleScenari...	Transmission T...	0.0126443655...
General		#0	RSUExampleScenari...	Packet Size (b...	1231.8482406...
General		#0	RSUExampleScenari...	Transmission T...	0.0126443655...
General		#0	RSUExampleScenari...	Packet Size (b...	1231.8482406...
General		#0	RSUExampleScenari...	Transmission T...	0.0126443655...
General		#0	RSUExampleScenari...	DDoS Attack ...	3.0
General		#0	RSUExampleScenari...	Packet Size (b...	1361.1337996...
General		#0	RSUExampleScenari...	Transmission T...	0.0053016132...
General		#0	RSUExampleScenari...	Packet Size (b...	1361.1337996...

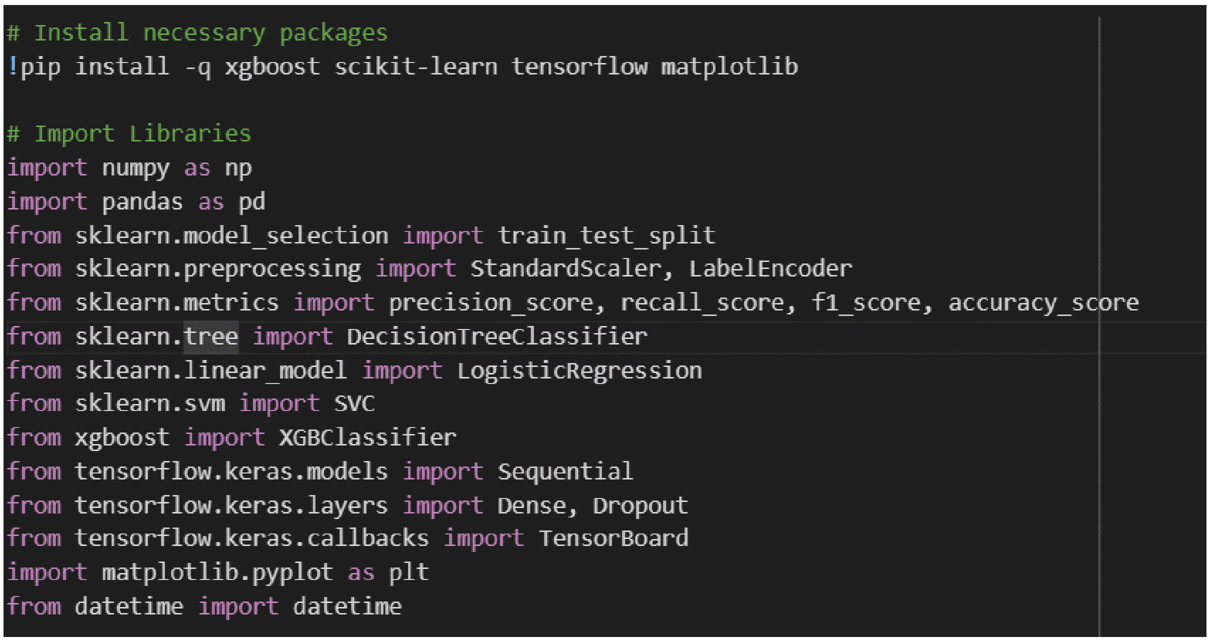
- To export the data, select the data and click right and select the option for export data and click on csv records. The is now save on the system in csv format.



7 DATA PRE-PROCESSING:

7.1 Packages:

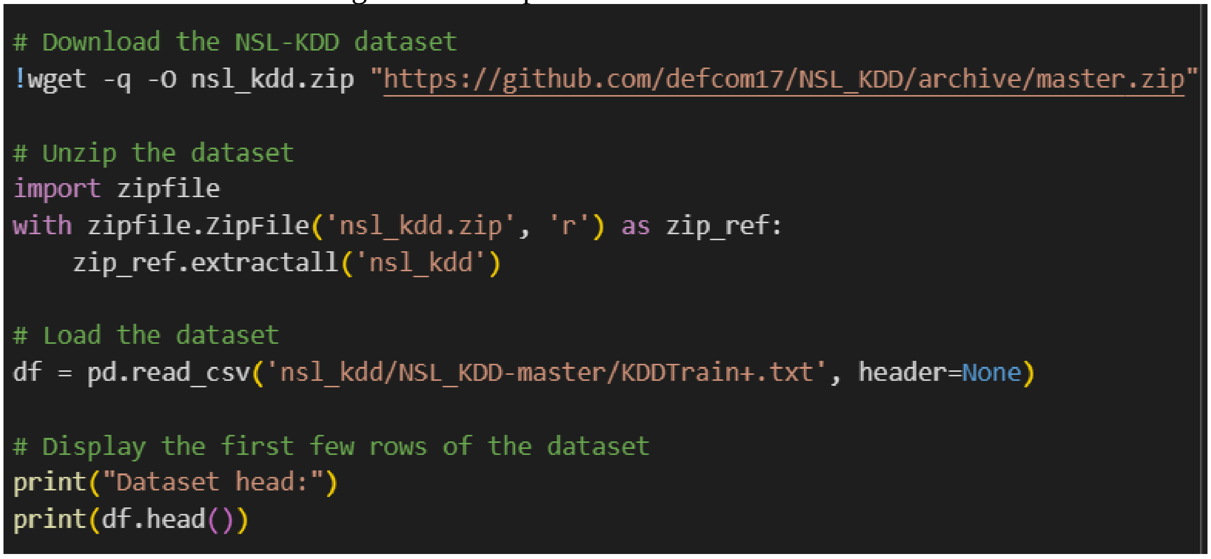
First install the necessary packages



```python
# Install necessary packages
!pip install -q xgboost scikit-learn tensorflow matplotlib

# Import Libraries
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler, LabelEncoder
from sklearn.metrics import precision_score, recall_score, f1_score, accuracy_score
from sklearn.tree import DecisionTreeClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.svm import SVC
from xgboost import XGBClassifier
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Dropout
from tensorflow.keras.callbacks import TensorBoard
import matplotlib.pyplot as plt
from datetime import datetime
```

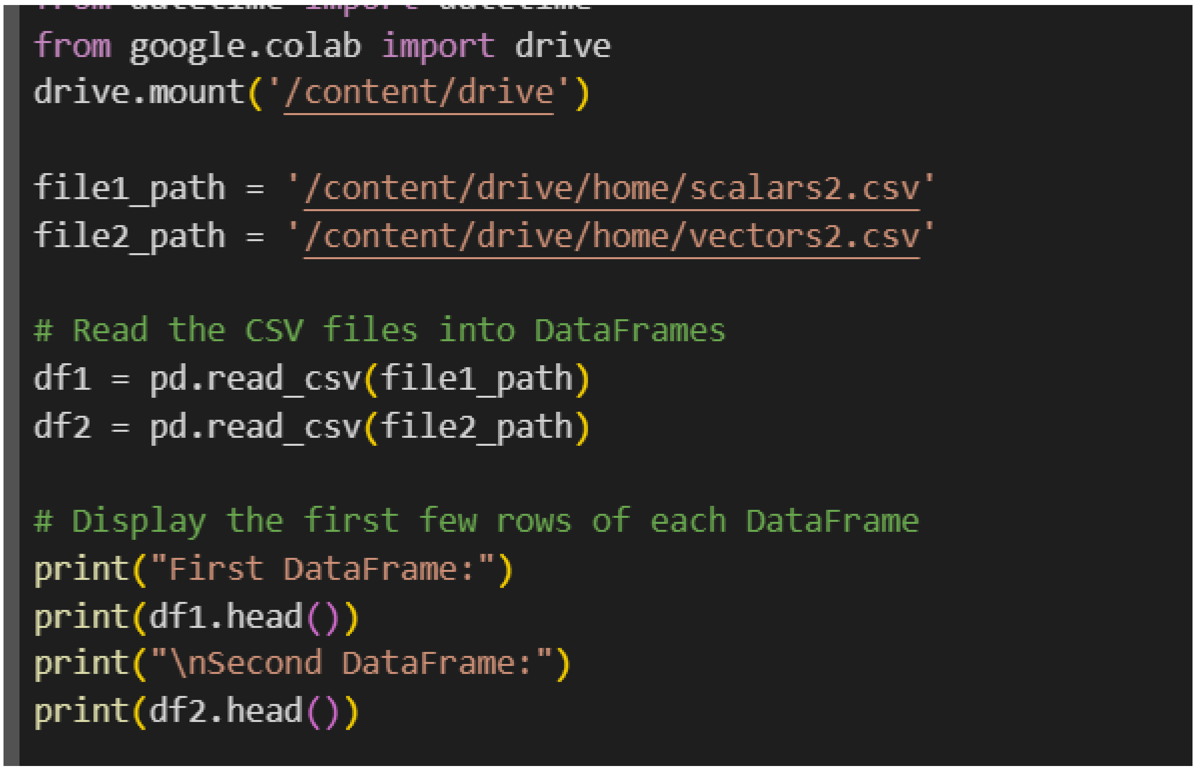
7.2 Dataset:

the NSL-KDD dataset being loaded and printed.



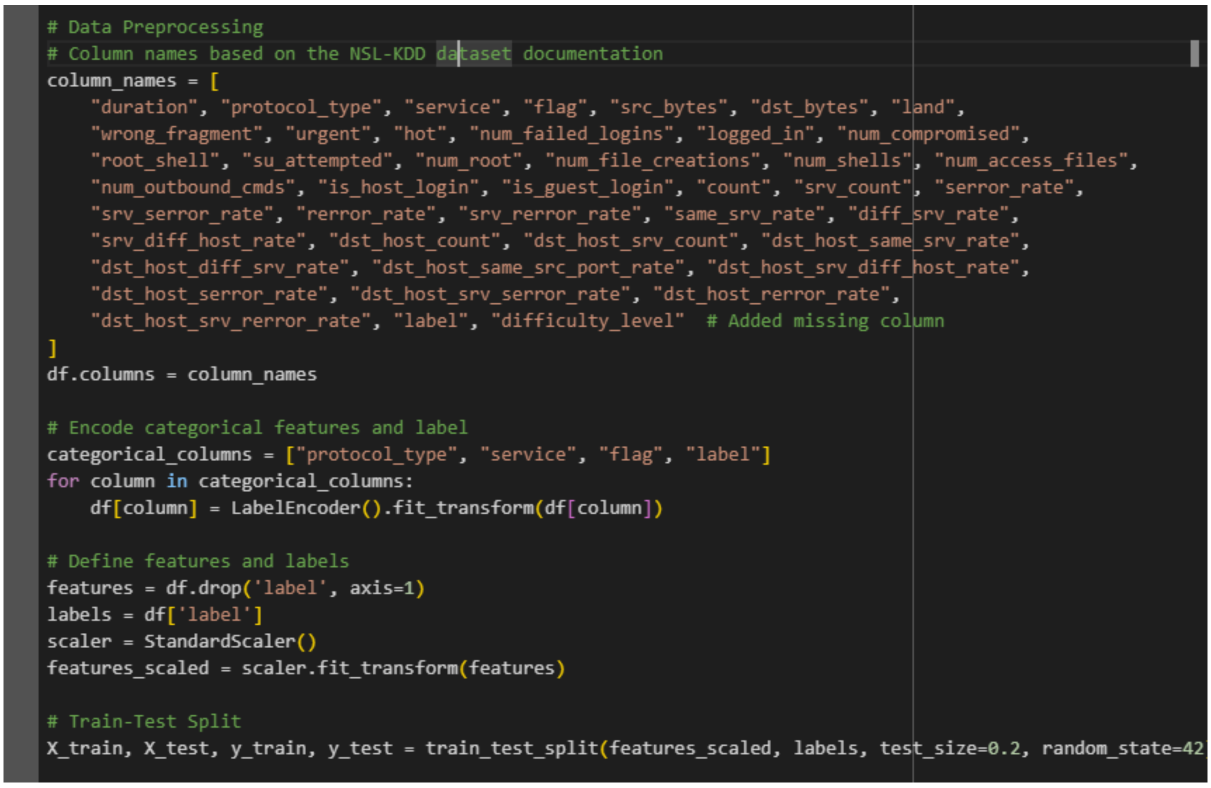
```python
# Download the NSL-KDD dataset
!wget -q -O nsl_kdd.zip "https://github.com/defcom17/NSL_KDD/archive/master.zip"

# Unzip the dataset
import zipfile
with zipfile.ZipFile('nsl_kdd.zip', 'r') as zip_ref:
    zip_ref.extractall('nsl_kdd')

# Load the dataset
df = pd.read_csv('nsl_kdd/NSL_KDD-master/KDDTrain+.txt', header=None)

# Display the first few rows of the dataset
print("Dataset head:")
print(df.head())
```

Import the simulation dataset from the google drive



```
from google.colab import drive
drive.mount('/content/drive')

file1_path = '/content/drive/home/scalars2.csv'
file2_path = '/content/drive/home/vectors2.csv'

# Read the CSV files into DataFrames
df1 = pd.read_csv(file1_path)
df2 = pd.read_csv(file2_path)

# Display the first few rows of each DataFrame
print("First DataFrame:")
print(df1.head())
print("\nSecond DataFrame:")
print(df2.head())
```

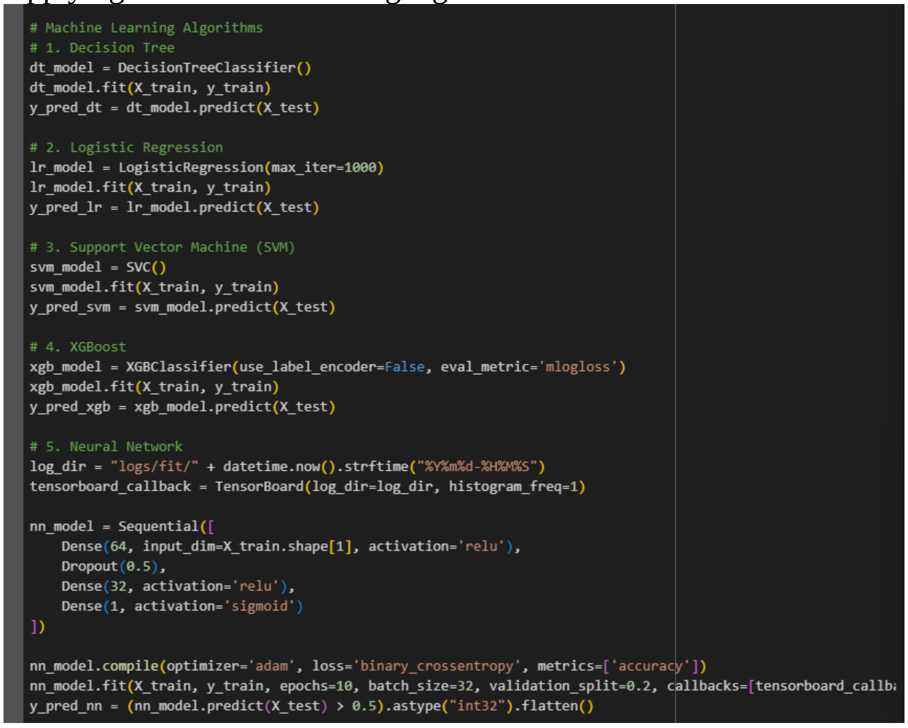
7.3 Data pre-Processing:



```
# Data Preprocessing
# Column names based on the NSL-KDD dataset documentation
column_names = [
    "duration", "protocol_type", "service", "flag", "src_bytes", "dst_bytes", "land",
    "wrong_fragment", "urgent", "hot", "num_failed_logins", "logged_in", "num_compromised",
    "root_shell", "su_attempted", "num_root", "num_file_creations", "num_shells", "num_access_files",
    "num_outbound_cmds", "is_host_login", "is_guest_login", "count", "srv_count", "serror_rate",
    "srv_serror_rate", "rerror_rate", "srv_rerror_rate", "same_srv_rate", "diff_srv_rate",
    "srv_diff_host_rate", "dst_host_count", "dst_host_srv_count", "dst_host_same_srv_rate",
    "dst_host_diff_srv_rate", "dst_host_same_src_port_rate", "dst_host_srv_diff_host_rate",
    "dst_host_serror_rate", "dst_host_srv_serror_rate", "dst_host_rerror_rate",
    "dst_host_srv_rerror_rate", "label", "difficulty_level"  # Added missing column
]
df.columns = column_names

# Encode categorical features and label
categorical_columns = ["protocol_type", "service", "flag", "label"]
for column in categorical_columns:
    df[column] = LabelEncoder().fit_transform(df[column])

# Define features and labels
features = df.drop('label', axis=1)
labels = df['label']
scaler = StandardScaler()
features_scaled = scaler.fit_transform(features)

# Train-Test Split
X_train, X_test, y_train, y_test = train_test_split(features_scaled, labels, test_size=0.2, random_state=42
```

7.4 Algorithms:

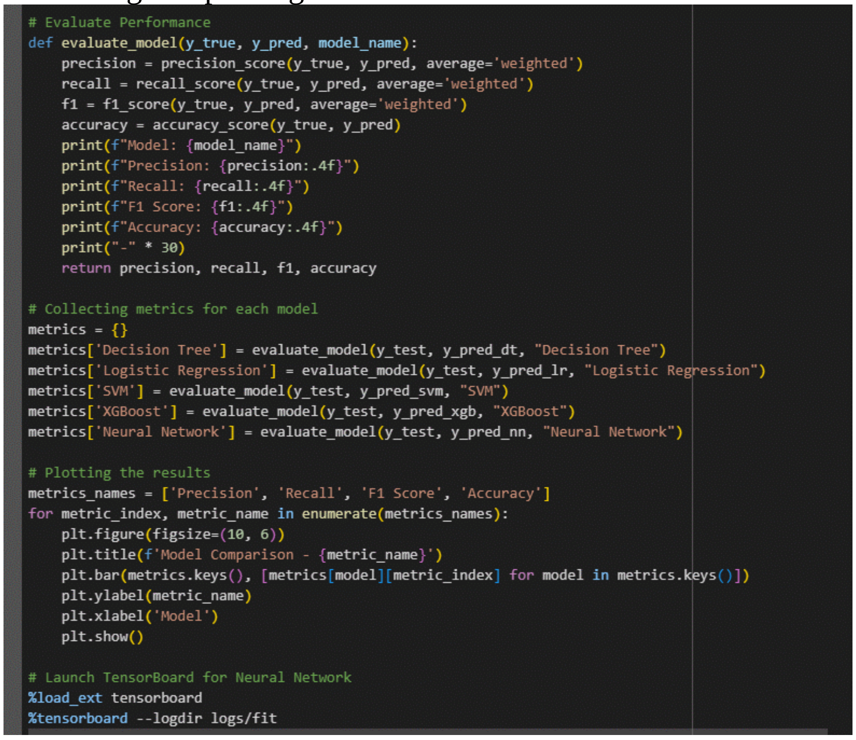
Applying the machine learning algorithms.



```
# Machine Learning Algorithms
# 1. Decision Tree
dt_model = DecisionTreeClassifier()
dt_model.fit(X_train, y_train)
y_pred_dt = dt_model.predict(X_test)

# 2. Logistic Regression
lr_model = LogisticRegression(max_iter=1000)
lr_model.fit(X_train, y_train)
y_pred_lr = lr_model.predict(X_test)

# 3. Support Vector Machine (SVM)
svm_model = SVC()
svm_model.fit(X_train, y_train)
y_pred_svm = svm_model.predict(X_test)

# 4. XGBoost
xgb_model = XGBClassifier(use_label_encoder=False, eval_metric='mlogloss')
xgb_model.fit(X_train, y_train)
y_pred_xgb = xgb_model.predict(X_test)

# 5. Neural Network
log_dir = "logs/fit/" + datetime.now().strftime("%Y%m%d-%H%M%S")
tensorboard_callback = TensorBoard(log_dir=log_dir, histogram_freq=1)

nn_model = Sequential([
    Dense(64, input_dim=X_train.shape[1], activation='relu'),
    Dropout(0.5),
    Dense(32, activation='relu'),
    Dense(1, activation='sigmoid')
])

nn_model.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])
nn_model.fit(X_train, y_train, epochs=10, batch_size=32, validation_split=0.2, callbacks=[tensorboard_callba
y_pred_nn = (nn_model.predict(X_test) > 0.5).astype("int32").flatten()
```

7.5 Generating results:

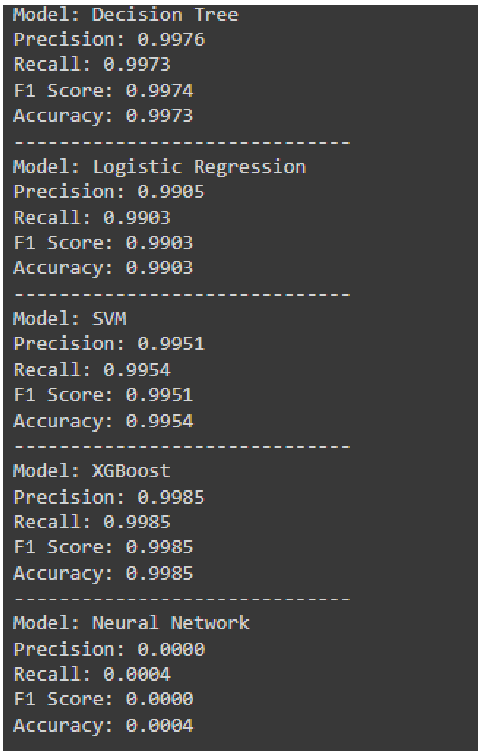
Evaluating and plotting the results.



```
# Evaluate Performance
def evaluate_model(y_true, y_pred, model_name):
    precision = precision_score(y_true, y_pred, average='weighted')
    recall = recall_score(y_true, y_pred, average='weighted')
    f1 = f1_score(y_true, y_pred, average='weighted')
    accuracy = accuracy_score(y_true, y_pred)
    print(f"Model: {model_name}")
    print(f"Precision: {precision:.4f}")
    print(f"Recall: {recall:.4f}")
    print(f"F1 Score: {f1:.4f}")
    print(f"Accuracy: {accuracy:.4f}")
    print("-" * 30)
    return precision, recall, f1, accuracy

# Collecting metrics for each model
metrics = {}
metrics['Decision Tree'] = evaluate_model(y_test, y_pred_dt, "Decision Tree")
metrics['Logistic Regression'] = evaluate_model(y_test, y_pred_lr, "Logistic Regression")
metrics['SVM'] = evaluate_model(y_test, y_pred_svm, "SVM")
metrics['XGBoost'] = evaluate_model(y_test, y_pred_xgb, "XGBoost")
metrics['Neural Network'] = evaluate_model(y_test, y_pred_nn, "Neural Network")

# Plotting the results
metrics_names = ['Precision', 'Recall', 'F1 Score', 'Accuracy']
for metric_index, metric_name in enumerate(metrics_names):
    plt.figure(figsize=(10, 6))
    plt.title(f'Model Comparison - {metric_name}')
    plt.bar(metrics.keys(), [metrics[model][metric_index] for model in metrics.keys()])
    plt.ylabel(metric_name)
    plt.xlabel('Model')
    plt.show()

# Launch TensorBoard for Neural Network
%load_ext tensorboard
%tensorboard --logdir logs/fit
```

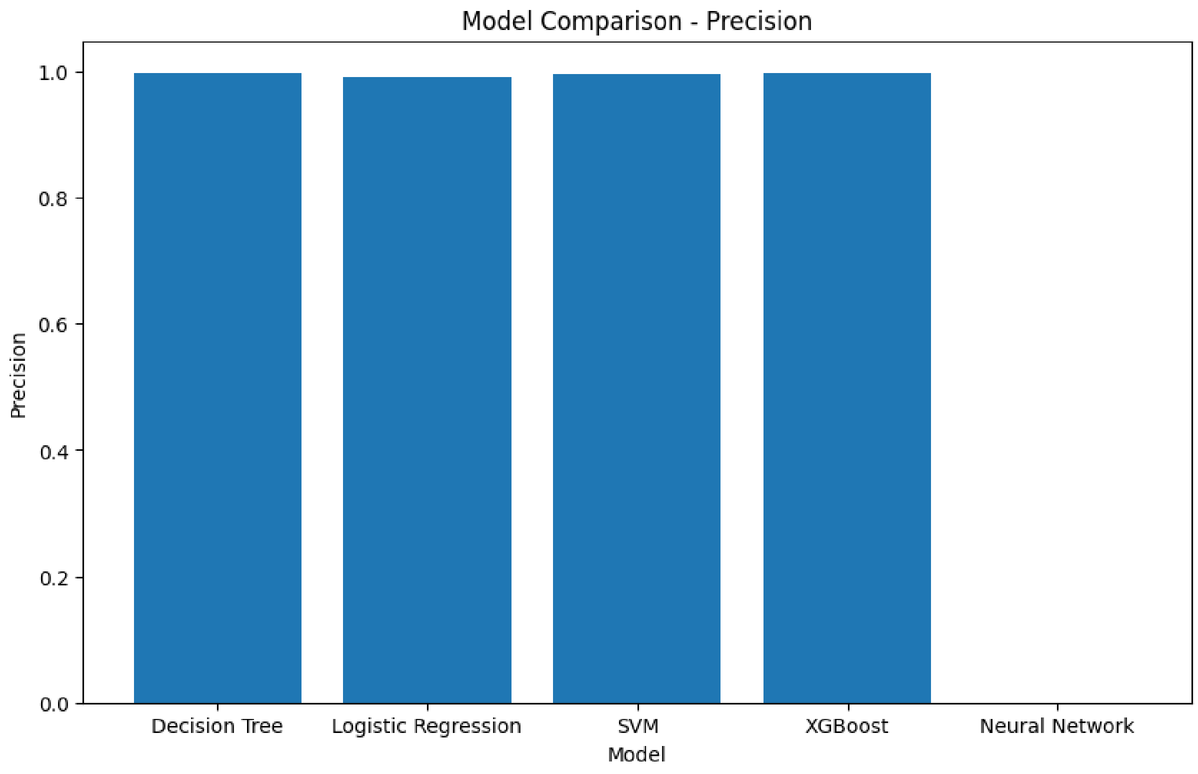
7.6 Results:

For are the results we got from this research

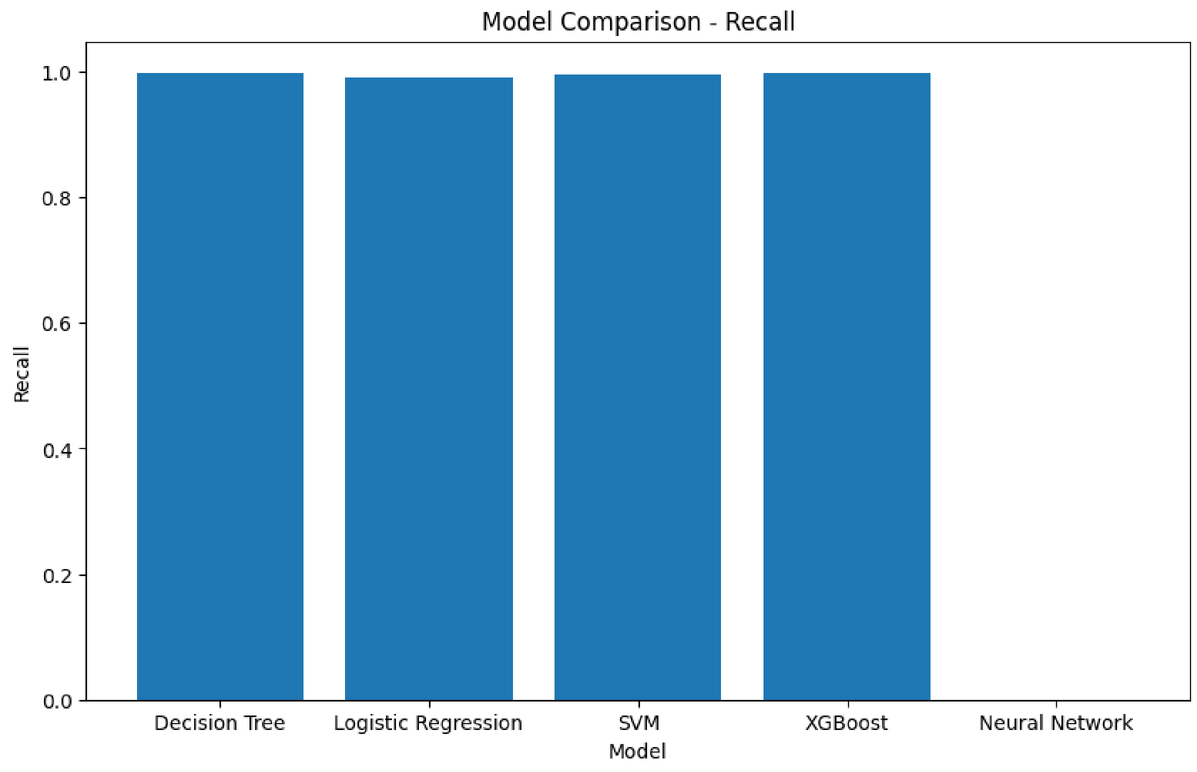


```
Model: Decision Tree
Precision: 0.9976
Recall: 0.9973
F1 Score: 0.9974
Accuracy: 0.9973
------------------------------
Model: Logistic Regression
Precision: 0.9905
Recall: 0.9903
F1 Score: 0.9903
Accuracy: 0.9903
------------------------------
Model: SVM
Precision: 0.9951
Recall: 0.9954
F1 Score: 0.9951
Accuracy: 0.9954
------------------------------
Model: XGBoost
Precision: 0.9985
Recall: 0.9985
F1 Score: 0.9985
Accuracy: 0.9985
------------------------------
Model: Neural Network
Precision: 0.0000
Recall: 0.0004
F1 Score: 0.0000
Accuracy: 0.0004
```

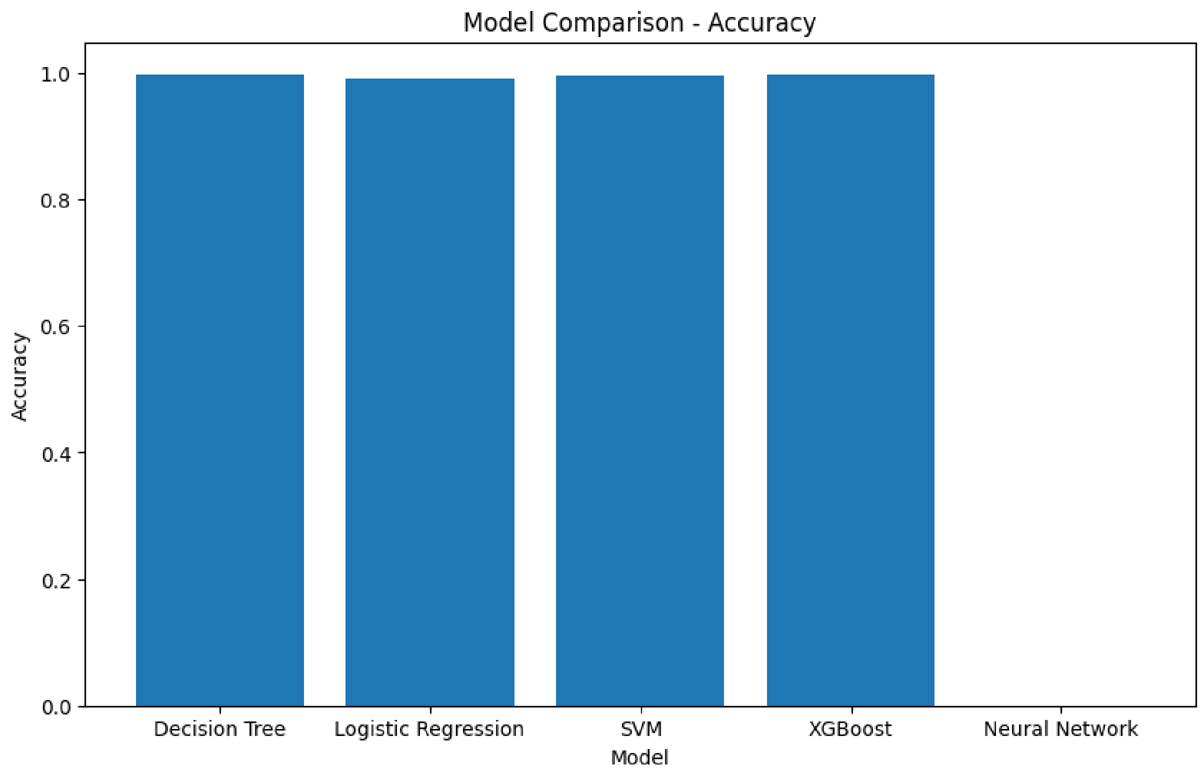
For precision:



For Recall:



For accuacy:


Model Comparison - Accuracy
Accuracy
Decision Tree
Logistic Regression
SVM
XGBoost
Neural Network
Model