Network Attack Detection Using Machine Learning

Pranjal Uday Parkar, Perina Sudarshan Ranpise, Prof Sulakshana Mane

Pranjal Uday Parkar, Dept. of Computer Engineering, Bharati Vidyapeeth college of Engineering, Navi Mumbai ,Maharashtra, India
Perina Sudarshan Ranpise, Dept. of Computer Engineering, Bharati Vidyapeeth college of Engineering, Navi Mumbai ,Maharashtra, India
Prof. Sulakshana Mane, Dept. of Computer Engineering, Bharati Vidyapeeth college of Engineering, Navi Mumbai, Maharashtra, India

---***---

Abstract - *In this paper, we have a tendency to illustrate the event of Network Attack Detection not to mention cubic centimetre technology. thanks to the advancement of net over years, the amount of attacks over net has additionally exaggerated. a robust intrusion detection system (IDS) is needed to confirm the safety and safety of the network. a completely unique supervised machine learning system is developed to classify network traffic whether or not it causes hurt or benign. to seek out the most effective model considering detection success rate, combination of supervised learning algorithmic program and have choice technique are utilized in this technique. it's found that artificial neural network (ANN) supported machine learning with wrapper feature choice exceed support vector machine (SVM) technique whereas classifying network traffic. to extend the performance, NSL-KDD dataset is employed to classify network traffic victimization SVM and ANN supervised machine learning techniques. Comparative study shows that the projected model is economical than the other existing models with relevancy intrusion detection success rate.*

Key Words: **Network attack detection, Machine Learning, Artificial Intelligence, Artificial Neural Network, SVM**

1. INTRODUCTION

With the fast development of network technology, network users square measure stern higher and better speed and quality of network services. Therefore, it's become one in all the challenges within the field of network operation and maintenance management to manage and management numerous network business traffic through effective technical means that, notice intrusion in network, distinguish completely different services, offer completely different quality assurance, and meet users' business desires. Currently With the massive quantity of information and knowledge, web has variety of challenges to form it stable and secure system. the' security may be ensured through change of firewall and software system, dynamic mechanisms may also be exploited. Intrusion detection system is one in all dynamic mechanisms beside net-work analysers et al. Intrusion detection determines specific goal of police investigation attacks. Intrusion monitors processes prevailing during a ADPS or network and analyses them to detect any deviation or any quite abnormalities, that square measure violations of pc security policies. With the wide spreading usages of web and will increase in access to on-line contents, crime is additionally happening at associate increasing rate. Intrusion detection is that the commencement to stop security attack. therefore, the protection solutions like Firewall, Intrusion Detection System (IDS), Unified Threat Modelling (UTM) and Intrusion hindrance System (IPS) have gotten abundant attention in studies. IDS detects attacks from a spread of systems and network sources by assembling info then analyses the data for potential security breaches. The network primarily based IDS analyses the info packets that travel over a network and this analysis square measure administrated in 2 ways in which. until these days anomaly { primarily based} detection is much behind than the detection that works supported signature and therefore anomaly based detection still remains a significant space for analysis. The challenges with anomaly primarily based intrusion detection square measure that it has to handle novel attack that there's no previous information to spot the anomaly. therefore, the system somehow has to have the intelligence to segregate that traffic is harmless and that one is malicious or abnormal and for that machine learning techniques square measure being explored by the researchers over the previous couple of years. IDS but isn't a solution to any or all security connected issues. for instance, IDS cannot compensate weak identification and authentication mechanisms or if there's a weakness within the network protocols. Studying the sphere of intrusion detection initial started in 1980 and also the initial such model was revealed in 1987. For the previous couple of decades, the' large business investments and substantial analysis were done, intrusion detection technology remains immature and therefore not effective. whereas network IDS that works supported signature have seen business success and widespread adoption by the technology primarily based organization throughout the world, anomaly primarily based network IDS haven't gained success within the same scale. because of that reason within the field of IDS, presently anomaly primarily based detection could be a major focus space of analysis and development. And before planning to any wide scale readying of anomaly primarily based intrusion detection system, key problems stay to be resolved. however, the literature these days is restricted once it involves compare on however intrusion detection performs once victimization supervised machine learning techniques. to guard target systems and networks against malicious activities anomaly-based network IDS could be a valuable technology. Despite the range of anomaly-based network intrusion detection techniques delineated within the literature in recent years, anomaly detection functionalities enabled security tools square measure simply commencing to seem, and a few vital issues stay to be resolved. many anomalies primarily based techniques are projected together with regression toward the mean, Support Vector Machines (SVM), Genetic rule, Gaussian mixture model, k-nearest neighbors rule, Naive Thomas Bayes classifier, call Tree. Among them the foremost wide used learning rule is SVM because it has already established itself on differing types of downside. One major issue on anomaly primarily based notion is the' of these projected techniques will detect novel attacks however all of them suffer a high warning rate generally. The cause behind is that the complexness of generating profiles of sensible traditional behaviour by learning from the coaching knowledge sets. these days Artificial Neural Network (ANN) square measure typically trained by the rear propagation rule, that had been around since 1970 because the reverse mode of automatic differentiation. The major challenges in evaluating performance of network IDS is that the inaccessibility of a comprehensive network primarily based knowledge set. Most of the projected anomaly primarily based techniques found within the literature were evaluated victimization KDD CUP ninety-nine dataset. during this project we have a tendency to used SVM and ANN –two machine learning techniques, on NSLKDD that could be a well-liked benchmark dataset for network intrusion. The promise and also the contribution machine learning did until these days square measure fascinating. There square measure several world applications we have a tendency to square measure victimization these days offered by machine learning. It appears that machine learning can rule the globe in returning days. therefore, we have a tendency to came out into a hypothesis that the challenge of distinguishing new attacks or zero day attacks facing by the technology enabled organizations these days may be overcome victimization machine learning techniques. Here we have a tendency to developed a supervised machine learning model which will classify unseen network traffic supported what's learnt from the seen traffic. we have a tendency to used each SVM and ANN learning rule to seek out the most effective classifier with higher accuracy and success rate.

1.1 Aim of the Project

The aim of IDS is to monitor the processes prevailing in a network and to analyse them for signs of any possible deviations. Some studies have been done in this field but a deep and exhaustive work has still not been done. In this project, we are using machine learning technique, Artificial Neural Network (ANN) is used for feature selection and SVM is used for classifying the network traffic.

2. EXISTING SYSTEM

Intrusion detection systems have become a standard component of network security infrastructure now-a-days. So far, several IDS's have been proposed. And they all have their own limitations. But there has been not enough significant works to put them all together. In existing system no any machine learning algorithm are used for intrusion detection.

Disadvantages of existing system

1. Lack of classification it not works properly.
2. No machine learning methods are used.
3. Less accuracy of existing system.

Irjet Template sample paragraph .Define abbreviations and acronyms the first time they are used in the text, even after they have been defined in the abstract. Abbreviations such as IEEE, SI, MKS, CGS, sc, dc, and rms do not have to be defined. Do not use abbreviations in the title or heads unless they are unavoidable.

3. PROPOSED SYSTEM

The system proposed is composed of feature selection and learning algorithm. Feature selection component are responsible to extract most relevant features or attributes to identify the instance to a particular group or class.

The learning algorithm component builds the necessary intelligence or knowledge using the result found from the feature selection component.

Using the training dataset, the model gets trained and builds its intelligence. Then the learned intelligences are applied to the testing dataset to measure the accuracy of home much the model correctly classified on unseen data.

4. METHEDOLOGY

There are various methods for handling the implementation and the consequent conversion from the old system to the new computerized system.

The most secure method for conversion from the old system to the new system is to run both the systems in parallel. In this new method, a person may operate in the manual older processing system as well as start operating the new computerized system. This method offers highsecurity and protection because even if there is a flaw in the computerized system, we can depend upon the manual system. However, the cost for maintaining these two systems in parallel is very high. This outweighs its benefits.

Another common method is a direct cut over from the existing manual system to the computerized system. The change may be within a week or within a day. There are no parallel activities. However, there is no remedy in case of a problem. This strategy requires careful planning and implementation.

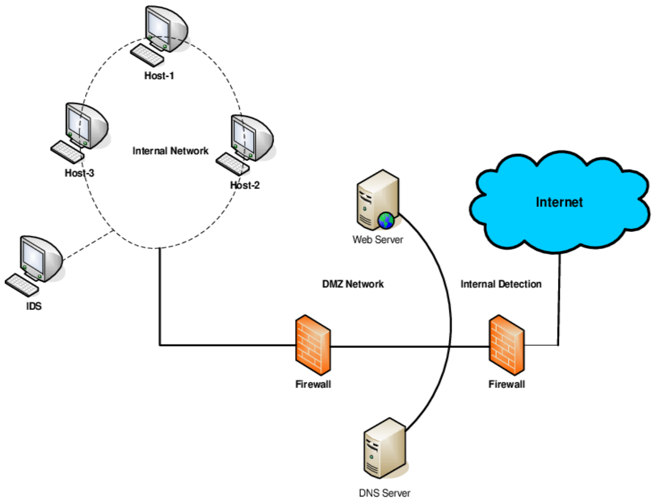


fig -1: Network Connection

1. Denial-of-Service(DoS)

A Denial-of-Service (DoS) attack is associate attack meant to wash up a machine or network by making it inaccessible to its users. Denial-of-service attacks accomplish this by flooding the target with traffic, or exploit it information that triggers. In every the cases, the Denial-of-service attack deprives legitimate users of the service or resource they expected. This attack is often targeted on high-profile organizations like banking, commerce, government and trade organizations. though this attack do not usually cause loss of any information but can price a victim a wonderful deal of it slow and money to handle. There ar two general ways that of DoS attack : Flooding services or blinking services. Flooding happens once system receives AN excessive quantity of traffic for the server to buffer, inflicting them to forestall then stop. variety of the popular flood attacks ar Buffer overflow attacks, ICMP flood, SYN flood. DoS attack simply exploit vulnerabilities to crash.

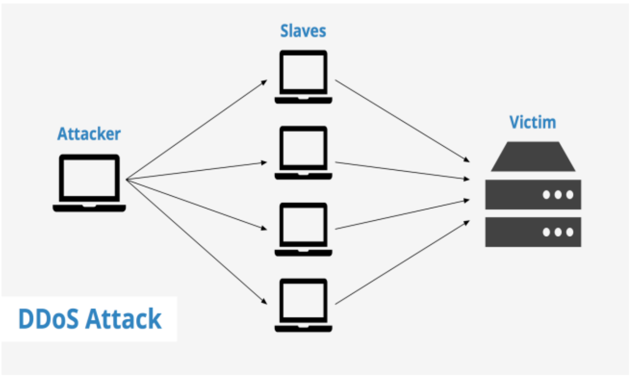


fig -1: DOS Attack

2. Man-in-the-Middle (MITM)

Man-in-the-middle attacks are a common type of cybersecurity attack that allows attackers to secretly listen to the communication between two targets. The attack takes place between two communicating hosts, allowing the attackers to "listen" to a conversation they should not be able to listen, hence the name "man-in-the-middle"

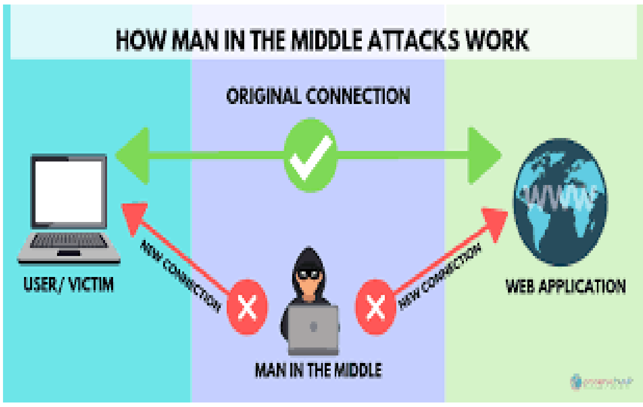


fig - 2: MIM Attack

3. Buffer overflow (BO)

Buffers are memory storage regions that briefly hold knowledge whereas it's being transferred from one location to a different. A buffer overflow (or buffer overrun) happens once the degree of information exceeds the storage capability of the memory buffer. As a result, the program trying to put in writing the information to the buffer overwrites adjacent memory locations. For example, a buffer for log-in credentials is also designed to expect username and password inputs of eight bytes, thus if a dealing involves associate input of ten bytes (that is, two bytes over expected), the program could write the surplus knowledge past the buffer boundary. Buffer overflows will have an effect on all kinds of computer code. They usually result from misshapen inputs or failure to allot enough area for the buffer. If the dealing overwrites viable code, it will cause the program to behave erratically and generate incorrect results, operation errors, or crashes.

A. Python

Python is an object-oriented, high-level programing language with dynamic semantics. Its high-level inbuilt data structures, combined with dynamic typing and dynamic binding, make it very alluring for Rapid Application Development, also as to be used as a scripting or glue language to connect existing components together. It is simple, easy to find out syntax emphasizes readability and it reduces the value of program maintenance. Python supports modules and packages, which inspires program standard and code reuse. This language interpreter and thus the extensive standard library are available in source or binary form for free of charge of charge for all major platforms, and should be freely distributed.

B. Anaconda

It is a free and open-source distribution of the Python and R programming languages for data science and machine learning related applications (large-scale data processing, predictive analytics, scientific computing), that aims to simplify package management and deployment. Package versions are managed by the package management system anaconda. The Anaconda distribution is used by over 6 million users, and it includes more than 250 popular data science packages suitable for Windows, Linux, and MacOS.

C. Artificial Neural Network (ANN)

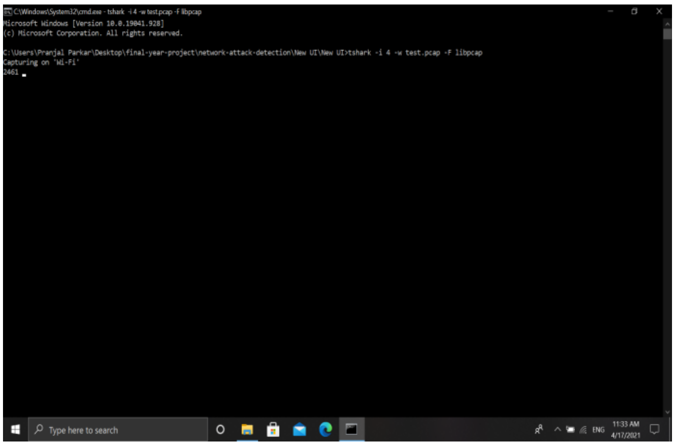
An artificial somatic cell network (ANN) might even be a process model supported the structure and functions of biological neural networks. data that flows through the network affects the structure of the ANN as a result of a neural network changes - or learns, throughout the simplest way - supported that input and output. ANNs square measure thought-about as nonlinear applied math information modeling tools wherever the complicated relationships between inputs and outputs square measure shapely or patterns square measure found. ANN is to boot brought up as a neural network.

An ANN has many blessings however one amongst the foremost recognized of these is that the undeniable fact that it will really learn from perceptive information sets. throughout this manner, ANN is employed as a random operate approximation tool. These types of tools facilitate estimate the foremost efficient and ideal ways for inbound at solutions whereas shaping computing functions or distributions. ANN takes information samples rather than entire information sets to achieve solutions, that saves each time and cash. ANNs square measure thought-about fairly easy mathematical models to strengthen existing information analysis technologies. ANNs have 3 layers that square measure interconnected. the first layer consists of input neurons. Those neurons send information on to the second layer, which in turn sends the output neurons to the third layer.

5. RESULT

In this section, the simulation is performed to validate the performance of the proposed algorithm. In the first phase, the experiment is conducted and analyzed that whether the proposed machine learning algorithm is able to differentiate between normal and anomaly behavior or not. The percentage of detection accuracy, false positive, false negative and time have been evaluated for the proposed method.

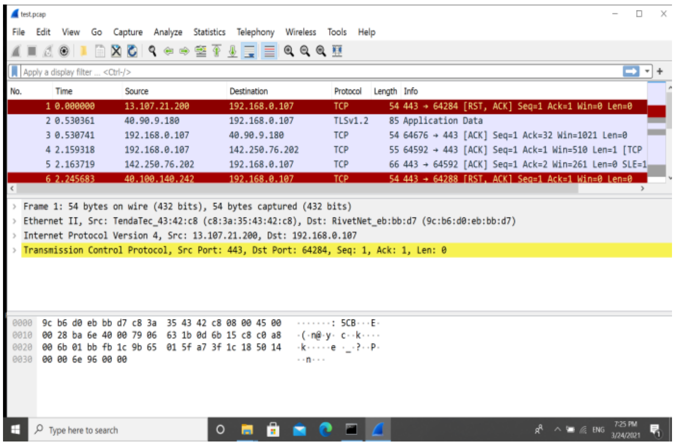
From the dataset, 10,000 samples are selected randomly which contains some normal and anomaly samples. 80% of the number of samples are used for training and rest of them are used for testing the algorithm. Then if any abnormal activity happens it detect the activity and also detect the types of the attack using ANN and SVM algorithm with good accuracy.



```
C:\Windows\System32\cmd.exe - tshark -i 4 -w test.pcap -F libpcap
Microsoft Windows [Version 10.0.19041.928]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Pranjal Parkar\Desktop\final-year-project\network-attack-detection\New UI\New UI>tshark -i 4 -w test.pcap -F libpcap
Capturing on 'Wi-Fi'
2461
```

fig -1: Packet Capturing



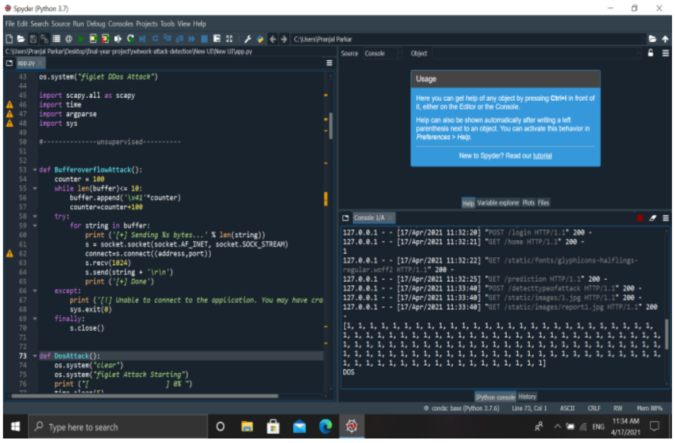
test.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>

```
No.   Time        Source          Destination     Protocol Length Info
    1 0.000000    13.107.21.200   192.168.0.107   TCP      54 443 → 64284 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
    2 0.530361    40.90.9.180     192.168.0.107   TLSv1.2  85 Application Data
    3 0.530741    192.168.0.107   40.90.9.180     TCP      54 64676 → 443 [ACK] Seq=1 Ack=32 Win=1021 Len=0
    4 2.159318    192.168.0.107   142.250.76.202  TCP      55 64592 → 443 [ACK] Seq=1 Ack=1 Win=510 Len=1 [TCP
    5 2.163719    142.250.76.202  192.168.0.107   TCP      66 443 → 64592 [ACK] Seq=1 Ack=2 Win=261 Len=0 SLE=1
    6 2.245683    40.100.140.242  192.168.0.107   TCP      54 443 → 64288 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0

> Frame 1: 54 bytes on wire (432 bits), 54 bytes captured (432 bits)
> Ethernet II, Src: TendaTec_43:42:c8 (c8:3a:35:43:42:c8), Dst: RivetNet_eb:bb:d7 (9c:b6:d0:eb:bb:d7)
> Internet Protocol Version 4, Src: 13.107.21.200, Dst: 192.168.0.107
> Transmission Control Protocol, Src Port: 443, Dst Port: 64284, Seq: 1, Ack: 1, Len: 0

0000  9c b6 d0 eb bb d7 c8 3a  35 43 42 c8 08 00 45 00   ·······: 5CB···E·
0010  00 28 ba 6e 40 00 79 06  63 1b 0d 6b 15 c8 c0 a8   ·(·n@·y· c··k····
0020  00 6b 01 bb fb 1c 9b 65  01 5f a7 3f 1c 18 50 14   ·k·····e ·_·?··P·
0030  00 00 6e 96 00 00                                  ··n···
```

fig -1: Captured packets



Spyder (Python 3.7)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\Pranjal Parkar
C:\Users\Pranjal Parkar\Desktop\final-year-project\network-attack-detection\New UI\New UI\app.py
app.py
Source Console Object

```python
os.system("figlet DDos Attack")

import scapy.all as scapy
import time
import argparse
import sys

#-------------unsupervised---------


def BufferoverflowAttack():
    counter = 100
    while len(buffer)<= 10:
        buffer.append('\x41'*counter)
        counter=counter+100
    try:
        for string in buffer:
            print ('[+] Sending %s bytes...' % len(string))
            s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
            connect=s.connect((address,port))
            s.recv(1024)
            s.send(string + '\r\n')
            print ('[+] Done')
    except:
        print ('[!] Unable to connect to the application. You may have cra
        sys.exit(0)
    finally:
        s.close()


def DosAttack():
    os.system("clear")
    os.system("figlet Attack Starting")
    print ("[                    ] 0% ")
```

Usage

Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

New to Spyder? Read our tutorial

```
127.0.0.1 - - [17/Apr/2021 11:32:20] "POST /login HTTP/1.1" 200 -
127.0.0.1 - - [17/Apr/2021 11:32:21] "GET /home HTTP/1.1" 200 -
1
127.0.0.1 - - [17/Apr/2021 11:32:22] "GET /static/fonts/glyphicons-halflings-
regular.woff2 HTTP/1.1" 200 -
127.0.0.1 - - [17/Apr/2021 11:32:25] "GET /prediction HTTP/1.1" 200 -
127.0.0.1 - - [17/Apr/2021 11:33:40] "POST /detecttypeofattack HTTP/1.1" 200 -
127.0.0.1 - - [17/Apr/2021 11:33:40] "GET /static/images/1.jpg HTTP/1.1" 200 -
127.0.0.1 - - [17/Apr/2021 11:33:40] "GET /static/images/report1.jpg HTTP/1.1" 200
-
[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]
DOS
```

IPython console History
conda: base (Python 3.7.6) Line 73, Col 1 ASCII CRLF RW Mem 88%
Type here to search
ENG
11:34 AM
4/17/2021

fig -1: Source code

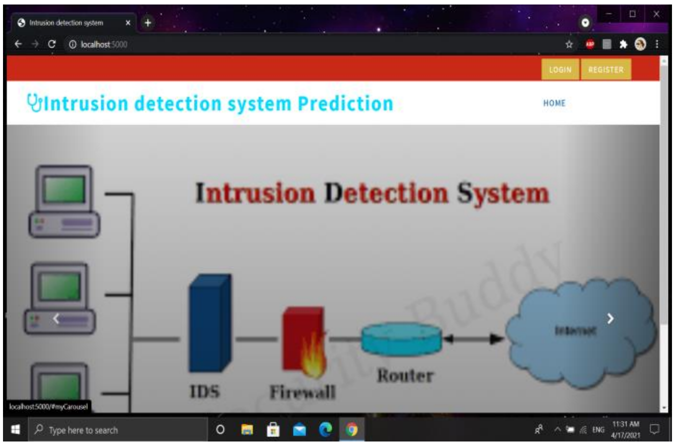
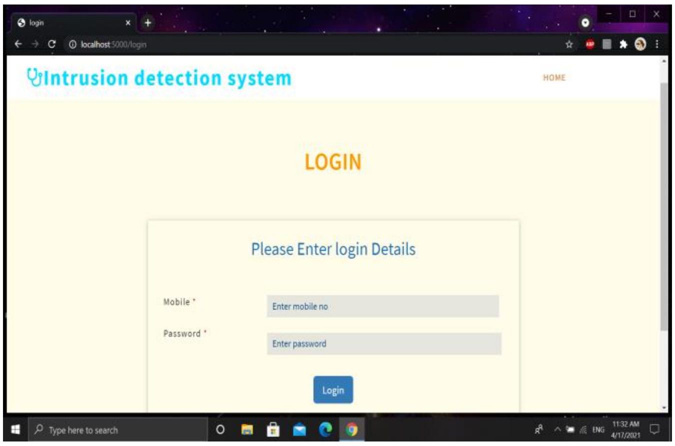

Intrusion detection system
localhost:5000
LOGIN
REGISTER
Intrusion detection system Prediction
HOME
Intrusion Detection System
IDS
Firewall
Router
Internet
localhost:5000/#myCarousel
Type here to search
11:31 AM
4/17/2021

fig -1: Application page



LOGIN

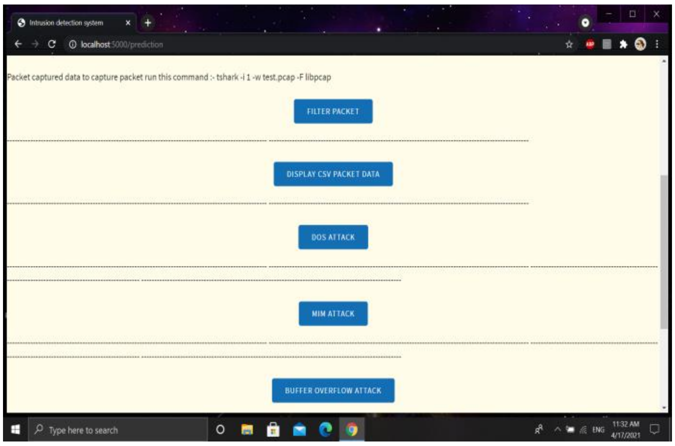
Please Enter login Details

Mobile * Enter mobile no

Password * Enter password

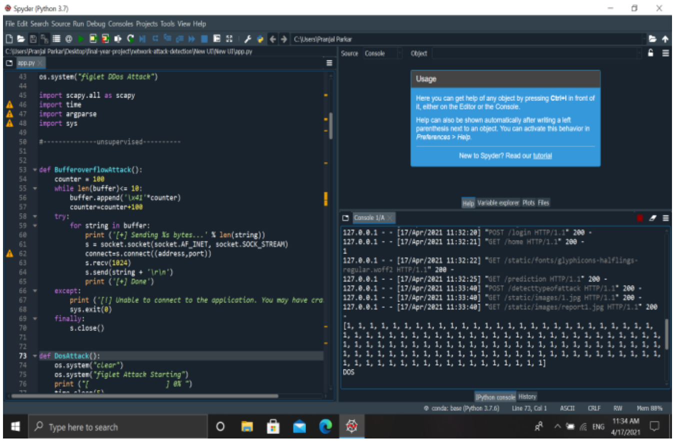
Login

fig -1: Login Page



Packet captured data to capture packet run this command :- tshark -i 1 -w test.pcap -F libpcap

fig -1: Prediction Page


Spyder (Python 3.7)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\Pranjal Parkar
C:\Users\Pranjal Parkar\Desktop\final-year-project\network-attack-detection\New UI\New UI\app.py
app.py
Source Console Object
Help Variable explorer Plots Files
Console 1/A
IPython console History
conda: base (Python 3.7.6) Line 73, Col 1 ASCII CRLF RW Mem 88%
Type here to search
11:34 AM
4/17/2021

```
os.system("figlet DDos Attack")

import scapy.all as scapy
import time
import argparse
import sys

#-------------unsupervised---------


def BufferoverflowAttack():
    counter = 100
    while len(buffer)<= 10:
        buffer.append('\x41'*counter)
        counter=counter+100
    try:
        for string in buffer:
            print ('[+] Sending %s bytes...' % len(string))
            s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
            connect=s.connect((address,port))
            s.recv(1024)
            s.send(string + '\r\n')
            print ('[+] Done')
    except:
        print ('[!] Unable to connect to the application. You may have cra
        sys.exit(0)
    finally:
        s.close()


def DosAttack():
    os.system("clear")
    os.system("figlet Attack Starting")
    print ("[                    ] 0% ")
```

Usage

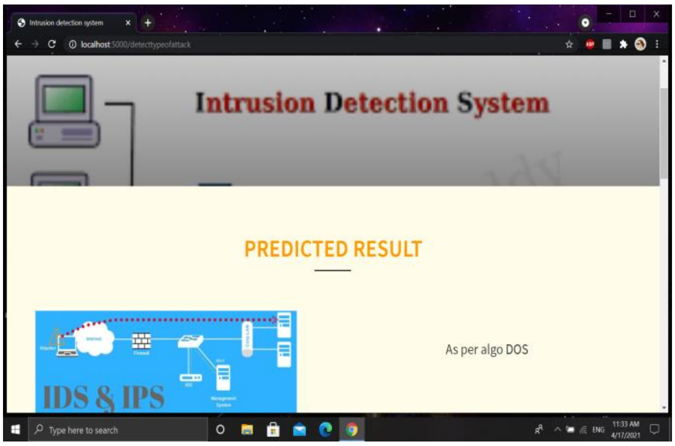
Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

New to Spyder? Read our tutorial

```
127.0.0.1 - - [17/Apr/2021 11:32:20] "POST /login HTTP/1.1" 200 -
127.0.0.1 - - [17/Apr/2021 11:32:21] "GET /home HTTP/1.1" 200 -
1
127.0.0.1 - - [17/Apr/2021 11:32:22] "GET /static/fonts/glyphicons-halflings-
regular.woff2 HTTP/1.1" 200 -
127.0.0.1 - - [17/Apr/2021 11:32:25] "GET /prediction HTTP/1.1" 200 -
127.0.0.1 - - [17/Apr/2021 11:33:40] "POST /detecttypeofattack HTTP/1.1" 200 -
127.0.0.1 - - [17/Apr/2021 11:33:40] "GET /static/images/1.jpg HTTP/1.1" 200 -
127.0.0.1 - - [17/Apr/2021 11:33:40] "GET /static/images/report1.jpg HTTP/1.1" 200
-
[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]
DOS
```

fig -1: Output after attack performed


Intrusion detection system
localhost:5000/detecttypeofattack
Intrusion Detection System
IDS & IPS
Type here to search
11:33 AM
4/17/2021

PREDICTED RESULT

As per algo DOS

fig -1: Result displayed

6. CONCLUSIONS

In this project, we have presented different machine learning models using different machine learning algorithms and different feature selection methods to find a best model. The analysis of the result shows that the model built using ANN and wrapper feature selection outperformed and all other models in classifying network traffic correctly with detection rate of 94.02%. We believe that these findings will contribute to research further in the domain of building a detection system that can detect known attacks as well as novel attacks. This attack detection system exist today can only detect known attacks, detecting new attacks or zero day attack still remains a research topic due to the high false positive rate of the existing systems.

7. REFERENCES

[1] Tchakoucht TA, Ezziyyani M. Building a fast intrusion detection system for high-speed-networks: probe and DoS attacks detection. Procedia Comput Sci. 2018;127:521–30.

[2] Zuech R, Khoshgoftaar TM, Wald R. Intrusion detection and big heterogeneous data: a survey. J Big Data. 2015;2:3.

[3] Sahasrabuddhe A, et al. Survey on intrusion detection system using data mining techniques. Int Res J Eng Technol. 2017;4(5):1780–4.

[4] Dali L, et al. A survey of intrusion detection system. In: 2nd world symposium on web applications and networking (WSWAN). Piscataway: IEEE; 2015. p. 1–6.

[5] Scarfone K, Mell P. Guide to intrusion detection and prevention systems (idps). NIST Spec Publ. 2007;2007(800):94.

[6] Debar H. An introduction to intrusion-detection systems. In: Proceedings of Connect, 2000. 2000.

[7] Ferhat K, Sevcan A. Big Data: controlling fraud by using machine learning libraries on Spark. Int J Appl Math Electron Comput. 2018;6(1):1–5.

[8] Peng K, Leung VC, Huang Q. Clustering approach based on mini batch Kmeans for intrusion detection system over Big Data. IEEE Access. 2018.

[9] Peng K. et al. Intrusion detection system based on decision tree over Big Data in fog environment. Wireless Commun Mob Comput. 2018. https://doi.org/10.1155/2018/4680867.

[10] Belouch M, El Hadaj S, Idhammad M. Performance evaluation of intrusion detection based on machine learning using Apache Spark. Procedia Comput Sci. 2018;127:1–6.