iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 7 **Issue:** X **Month of publication:** October 2019

DOI: http://doi.org/10.22214/ijraset.2019.10076

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.177
Volume 7 Issue X, Oct 2019- Available at www.ijraset.com

Network Intrusion Detection using Machine Learning Techniques

Prajeeth Kumar M.J[1], Sharvan A. J.[2], Aarthan.A[3], Manjula R[4]
[1,2,3,4] *School of Computer Science and Engineering, Vellore institute of Technology, Vellore, Tamilnadu, India.*

Abstract: ***The modern era deals with a different type of structured and unstructured information. The information can be accessed, processed and retrieved from different major or minor resources. One of the biggest storage of information is the internet. So, in order to deal with the resources of information, we need to take a holistic approach with security and integrity for the networks. The common approaches to security deal with firewall-based mechanisms. However, beyond a certain extent, this approach fails as it works on the principle of definition and signature. In this paper, a novel method is proposed for the protection of the network-based resources from the outside world intruders by means of the machine learning techniques. One of the major pros of using a machine learning-based approach is the self-capability of the machine in separation, regression, and classification of the intruder patterns.***

Keywords: Firewall, Internet Security, Intruder, Machine Learning, naïve Bayes

I. INTRODUCTION

In today's era, where almost every association is completely reliant on the Internet to survive, so need to survive to certification the security and security of the data is an important task. This issue has been brought further into a closer view by pushing to internet security. The main objective of a system interruption discovery framework is to locate wrong, inaccurate, and irregular action on a system or on the hosts having a place with a nearby system by observing the content.

As dependence upon the utilization of transmitted information over machine systems, for example, the Internet has expanded, so has the need for securing these systems from vindictive clients. Numerous routines for distinguishing malignant interlopers e.g., firewalls, secret key ensured frameworks presently exist. Notwithstanding, these customary techniques are getting to be progressively defenseless and wasteful because of their inalienable issues. Thus, new strategies for interruption location that are not hampered by defencelessness and wastefulness must be created.

Intrusion detection systems exiting in now require human input for attack methods or to determine effective models for normal behaviour. Support for learning algorithms enhances a better alternate to expensive human input which more efficient than existing methodology. One of the major objectives of the learning algorithm is to make machines enhance their efficiencies and accuracy in determining the nature of classification. It is extremely important that the security components of a framework are planned in order to avert unapproved access to framework assets and information. We can be that as it may, attempt to catch these interruption endeavours so that move may be made to repair the harm later. This field of exploration is called Intrusion Detection.

As reliance upon the use of digitally transmitted data over computer networks such as the Internet has increased, so has the need for protecting these networks from malicious users (commonly called "hackers" or "crackers"). Many methods for detecting malicious intruders e.g., firewalls, password-protected systems currently exist. However, these traditional methods are becoming increasingly vulnerable and inefficient due to their inherent problems. As a result, new methods for intrusion detection that are not hampered by vulnerability and inefficiency must be developed.

One of the biggest challenges in network-based intrusion detection is the extensive amount of data collected from the network. Therefore, before feeding the data to a machine learning algorithm, raw network traffic should be summarized into higher-level events such as connection records. Each higher-level event is described with a set of features. Selecting good features is a crucial activity and requires extensive domain knowledge.

A. Denial-of-Service

Dos assaults are likely to be most troublesome to address. These are the nastiest, on the basis that they're not difficult to launch, troublesome to track, and it isn't difficult to deny the service of the user, without denying real authentication demands procedure. The reason for a Dos assault is basic: send a larger number of appeals to the machine than it can deal with.

B. Unauthorized Access

It is an abnormal amount term that can that deals with potentially different types of attacks. The objective, of these kinds, is to get to some asset that your machine. Case in point, a host could be a web server and request to anybody with asked for website pages. There are commonly two types of unauthorized access:

1) Executing Commands Illicitly
2) Confidentiality Breaches

C. Destructive Behavior

Among the destructive sorts of attacks, there are two major classifieds with

1) *Data Diddling:* Tampering with user documents with different access such as read and writes to modify the user's information.
2) *Data Destruction:* Deleting the users' file information by gaining access rights.

II. RELATED WORK

Most researchers focus on using only one algorithm for the detection of multiple attack categories. Therefore, the performance evaluated is dismal in some cases. Firewalls can provide security against unauthorized access but it cannot detect the network attacks. One method is to use two signature-based network intrusion detection systems as proposed by Shiri et al in [1], but it requires encryption of data in a better way. One other way of providing such a method is proposed by Tian et al in [2] which is based on a Radial basis function neural network which uses the K-nearest algorithm for providing protection against the attack but it works well only with local networks.

Tian et al in [3] proposed intruder detection based on high speed and precise genetic algorithm which is a better way of intruder detection. Bao et al in [4] proposed a method based on the Vector machine which is also one of the ways of implementing intruder detention but suffers from data-related issues.

Sommer et al in [5] proposed based on machine learning provides a knowledge base for storing the information. Hu et al in [6] proposed two algorithms for intrusion detection based on Adaboost parameterized methods. Decision stumps form the basis of weak classifiers in the first algorithm. And, a mixture of Gaussian techniques forms the basis to the improvised second algorithm. The limitation of this design is those varieties if attacks are not taken into account for simulation.

Wonghirunsombat et al in [7] proposed a centralized architecture to detect and prevent network intrusion using a web application, monitored by a network administrator. Since the architecture is centralized, the administrator node is prone to failure. As a result, the reliability offered using this design architecture is significantly low. Ying et al in [8] proposed a performance evaluation technique for a set of large data files distributed in parallel operating processors forming a mesh, tree or hypercube topology.

Yang et al in [9] proposed architecture to detect network intrusion using a parameterized based approach. Many cyber-attacks are detected using this approach. The limitation of this design is the constrained understanding of SCADA applications and rules. Kang et al in [10] proposed a survey that discusses the opportunities and challenges in preventing network intrusion and attacks for SCADA systems. Fuzzy based intrusion detection framework for an attack where intruder drops packets from mobile-based ad-hoc networks. The limitation for this design is constrained to only attacks where the intruder drops packets. However, other attacks are not detected.

III.PROPOSED WORK

One of the biggest challenges in network-based intrusion detection is the extensive amount of data collected from the network. Therefore, before feeding the data to a machine learning algorithm, raw network traffic should be summarized into higher-level events such as connection records. Each higher-level event is described with a set of features. Selecting good features is a crucial activity and requires extensive domain knowledge. For achieving this goal we are implementing a NIDS algorithm for network intrusion detection with adaptive and dynamic updating of the knowledge base in the networking devices.

A. NIDS Algorithm

As shown in Fig. 1 the supervised train model used by using Machine Learning which enhances the input to the probabilistic approach of naive Bayes. The different parameters associated with the kernel of the SVM classifier are set. This gives to the way of predication by selecting appropriate classifiers by desired by type of attack. After this step training of machine completed successfully and it gives rise to the benchmark model for the generation of classifier and separator. This can be applied to network-based packet for regeneration and classification.

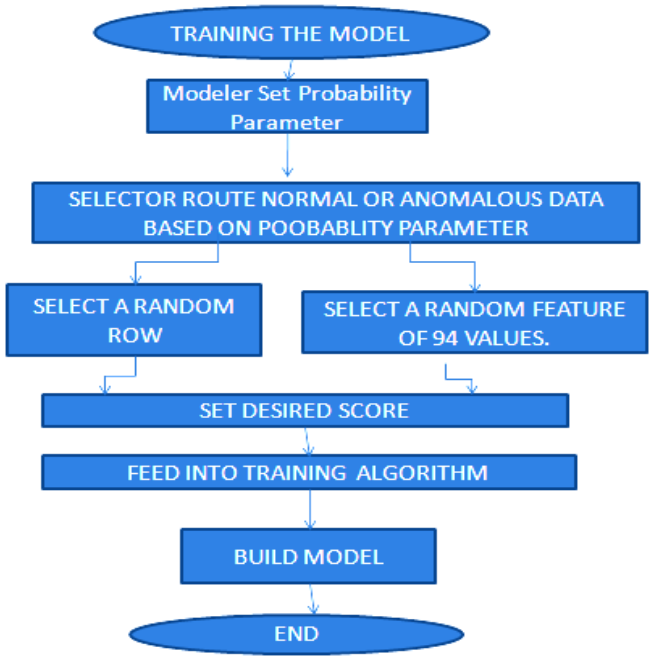


Fig 1: NIDS Flow chart

The working of the proposed work is described below.

1) *Training the Model:* The training module is responsible for providing the basis of the classification. By using the training module one can consider a set of support vector which has been used in this paper as an implemented algorithm for providing classification. In the implementation part, we need to analyse and observe the operational view of the simulation. We will initialize the status, passed and failed bit and then we need to select the SVM type, kernel type, and training file in order to start the simulation. Training module algorithm is as follows:

status=0;passed=0;failed=1;
Begin
Step 1: Select SVM type, kernel type, and import training file then simulate.
Step 2: Notify Alert message to import train file if not imported.
Step 3: Select the SVM kernel type but do not select kernel type and simulate the training module.
Step 4: Train the file with the default kernel and its parameters.
Step 5: Verify the validation result for the train simulation.
Step 6: If no error message the stop otherwise go to step 1.
End

2) *Predict Module:* This is a process of binary classification/prediction a label is attached to each record. This label can be either '+1' or '-1'. We can assign which label will indicate what type of attack. The process of predicting the module is shown below by using the prediction algorithm.

Step 1: Do not any predict the file before the simulation of the training module.
Step 2: import test file and compute predict and display alert message to import the model file if the file is not imported.
Step 3: Import model file and go to predict module simulation.
Step 4: import both test file and model file then simulate predict module.
Step 5: Verify the validation result for the train simulation.
Step 6: If no error message the stop otherwise go to step 1.
End

B. NIDS DFD Model

A data-flow diagram (DFD) is a graphical representation of the "flow" of data through an information system. DFDs can also be used for the visualization of data processing. Fig. 2 shows the NIDS DFD diagram. As shown in Fig. 2 The central part consist of NIDS is the central processing system and accepts the request from the administrator the specific input for it is raw data that is classified into 39 attacks which act as a dataset for the NIDS the output specified is based on modular classification of binary classification, Predication of accuracy based on arithmetic. In processing raw data the machine import files for the purpose of machine learning. The training file used to train the classifier to update the machine in train prediction. This training module will act as input to accuracy prediction as shown in fig. 2.

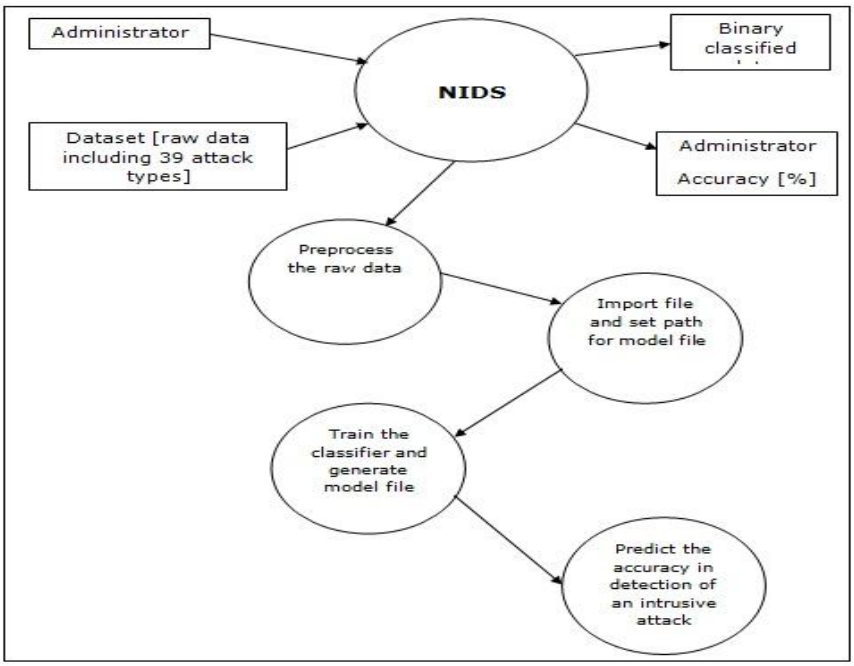


Fig 2: NIDS DFD Diagram

C. Implementing Dynamic updation Algorithm with SVM

The Support Dynamic Updation algorithm deals with the training issue of the machine to behave system intelligently this method plays a very crucial role in pattern separation. This model acts on space representation of coordinates and also with hyperbolic and linear planes on separable or inseparable points. The points for which Lagrange multipliers is non-zero and which lie on the margin of the separation are called Support Vectors.

D. Accuracy and Attack Classification

Accuracy Classification indicates the complete overall picture of the accuracy which is calculated based on the Bayes model used as a separate classifier for detecting accuracy. The process of the total packet transferred and separation of the packet into normal and the detected type of classification.

IV. IMPLEMENTATION

The proposed model deals with a different component of SVM, Dataset from KDD CUP and LIBSVM as major library resources. The processor can be of typically P4 and above with a minimum 32 MB of RAM. The proposed work can be implemented on a different version of the operating system ranging from Windows XP and Windows 7.

A. Dataset details

Attacks fall into four main categories:

1) *DOS:* denial-of-service, e.g. syn flood;
2) *R2L:* unauthorized access from a remote machine, e.g. guessing password;
3) *U2R:* unauthorized access to local superuser (root) privileges, e.g., various ``buffer overflow'' attacks;
4) *Probing:* surveillance and another probing, e.g., port scanning.

We have actually divided the dataset into 5 meta-classes' viz., NORMAL, DOS, PROBE, R2L, and U2R. This is shown in Table 1. After dividing the dataset into 5 meta-classes, we compiled records of two classes and classified them successfully using binary classification method.

DOS	R2L	U2R	Probe
Smurf	Snmpgetattack	buffer overflow	Ipsweep
Pod	Named	Perl	Saint
apache2	Xlock	Xterm	Portsweep
Udpstorm	Multihop	Ps	Satan
process_table	Xsnoop	Rootkit	Mscan
Neptune	Sendmail	Loadmodule	Nmap
Back	guess_passwd	Httptunnel	
Mailbomb	Phf	Worm	
Teardrop	warez_master	Sqlattack	
Land	Imap	Snmpguess	
	ftp_write		

Table 1: Classification of attacks in different classes

B. Training

In the process of training, we'll get some support vectors (SV) that are responsible for the entire classification. Fig. 3 shows the screenshot of the training module Operational View.

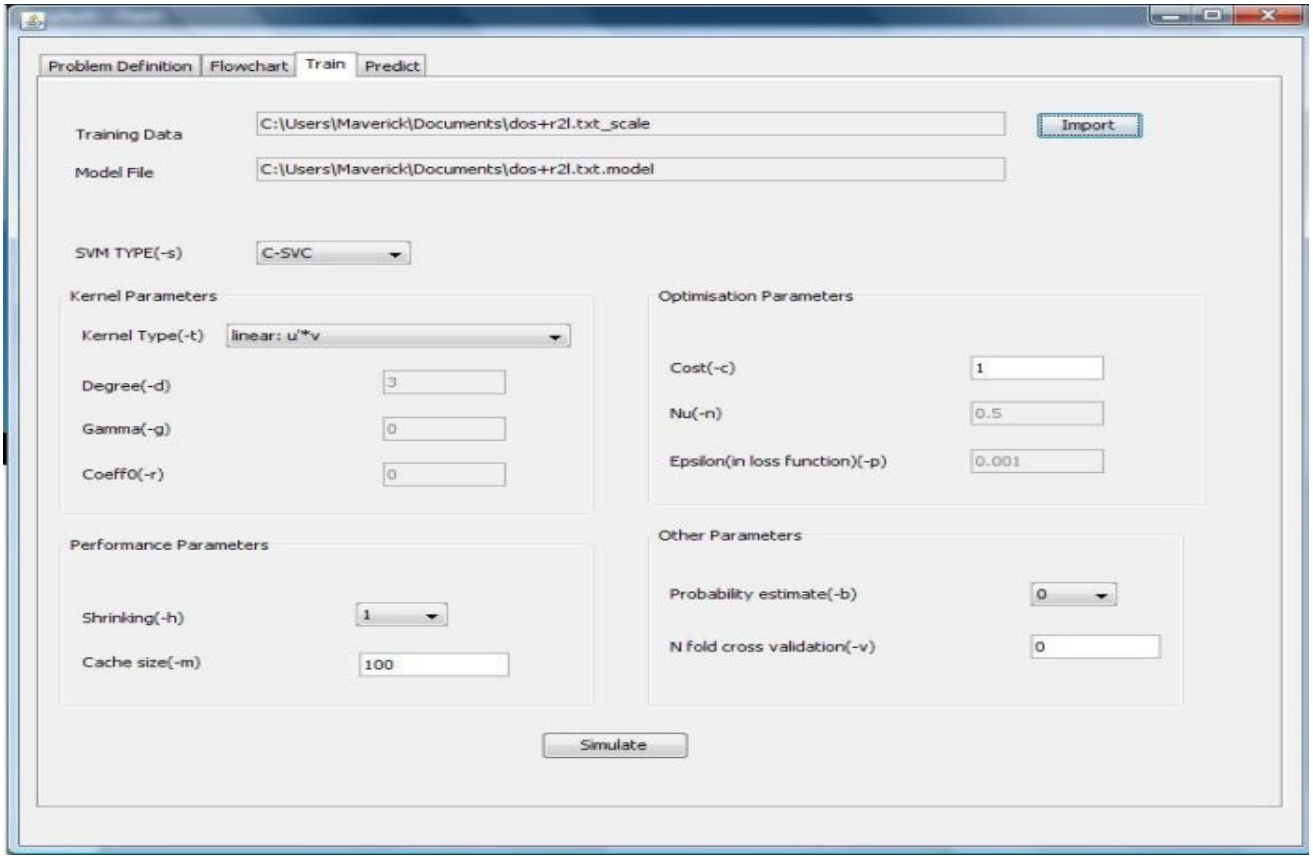


Fig.3: Train Module Operational View

Fig. 4 Shows the Train Module after simulation. In this module, the figure shows after the training giving the machine and message indicate training completed successfully. In the process of training, we'll get some support vectors (SV) which are responsible for the entire classification

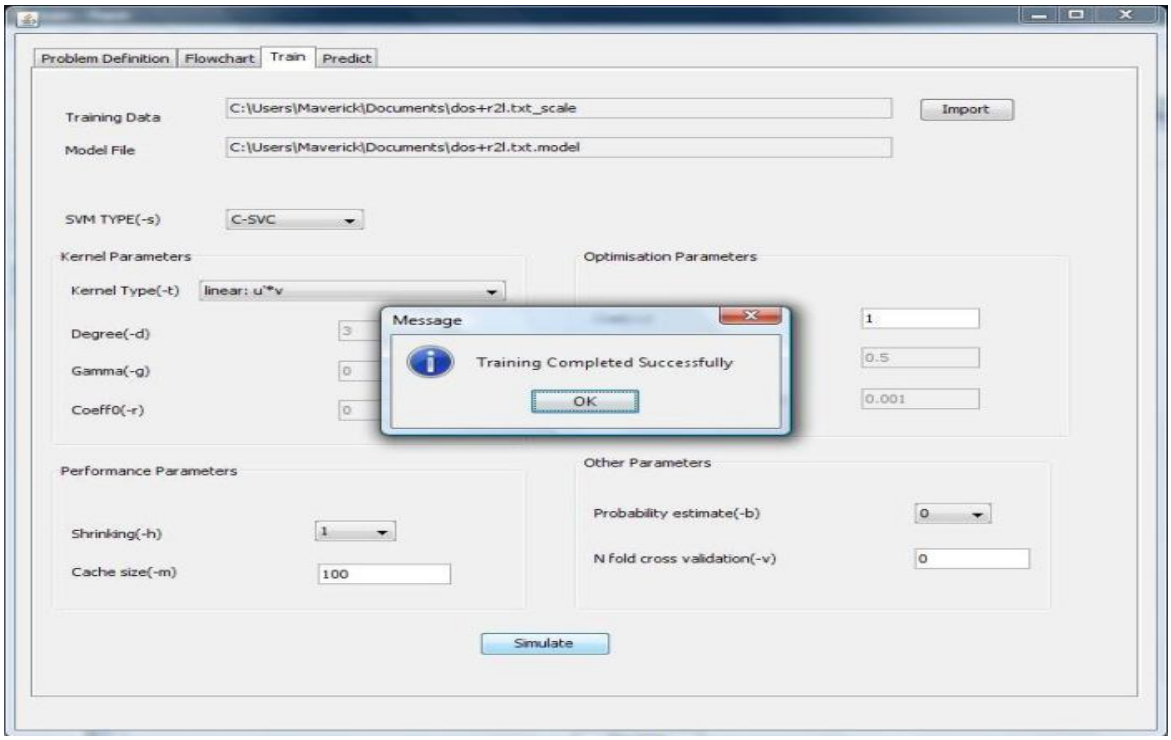


Fig. 4: Train Module Operational View after Simulation

V. RESULT AND ANALYSIS

A. Predict

This is a process of binary classification/prediction a label is attached to each record. This label can be either '+1' or '-1'. We can assign which label will indicate what type of attack. Fig. 5 shows the screenshot of predict module Operational View and classify the DOS and R2R types of attack and the number of packets in the figure.

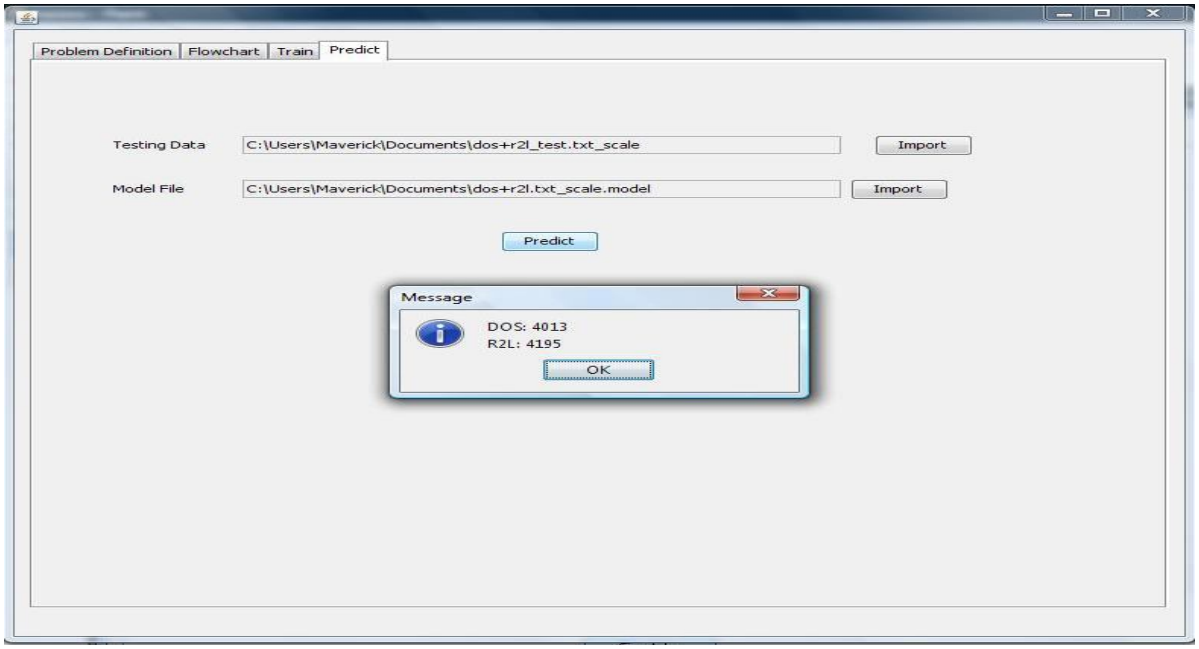


Fig. 5 Predict Module Operational View

B. Accuracy and Attack Classification

Fig. 6 and Fig. 7 shows the screenshot of Accuracy Classification this indicates the complete overall picture of the accuracy which is calculated based on the Bayes model used as a separate classifier for detecting accuracy.

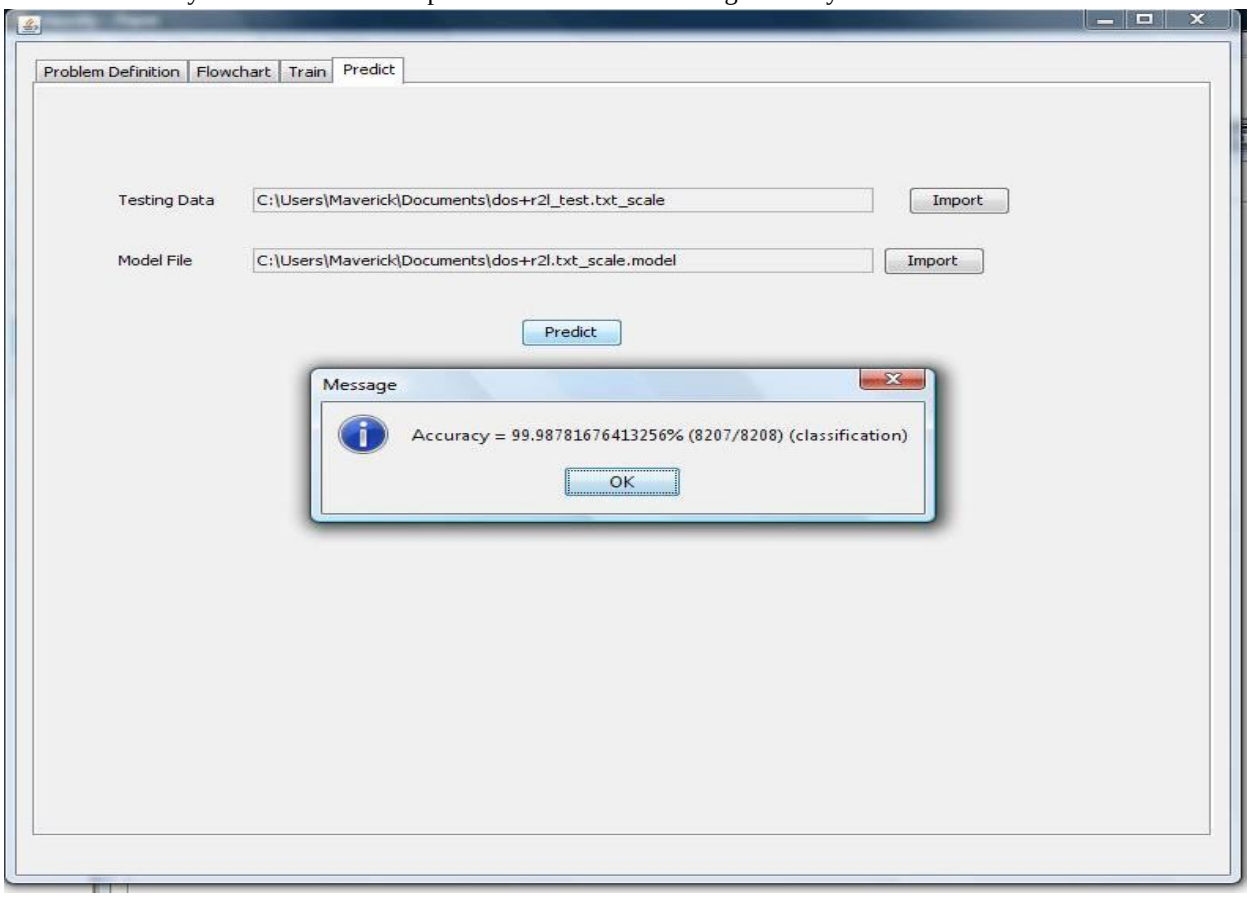


Fig. 6 Predict Module Operational View after Simulation

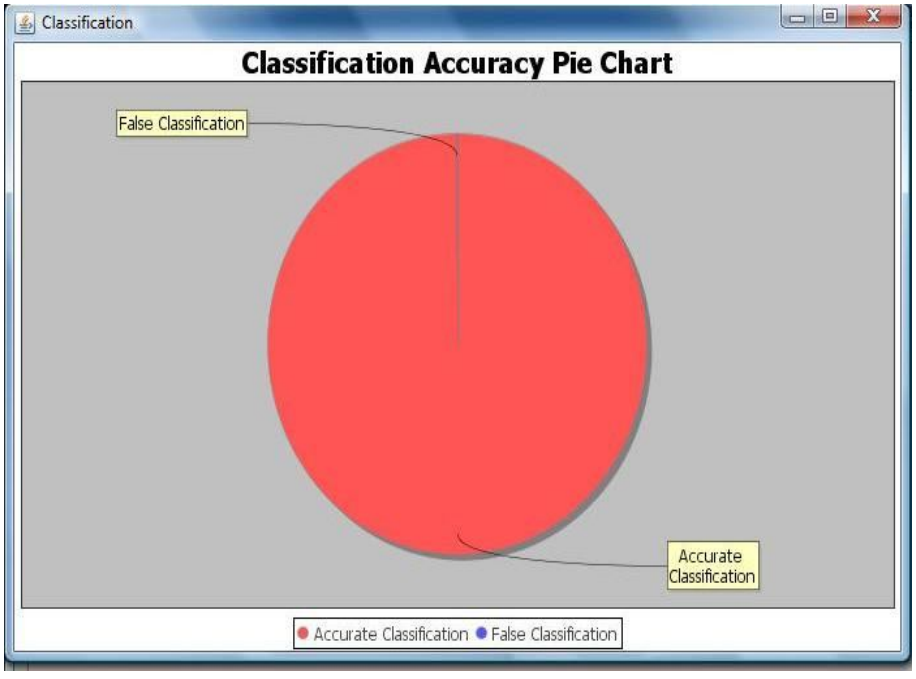


Fig. 7: Accuracy Classification

Fig. 8 shows attack classification under the process of the total packet transferred and separation of the packet into normal and the detected type of classification.

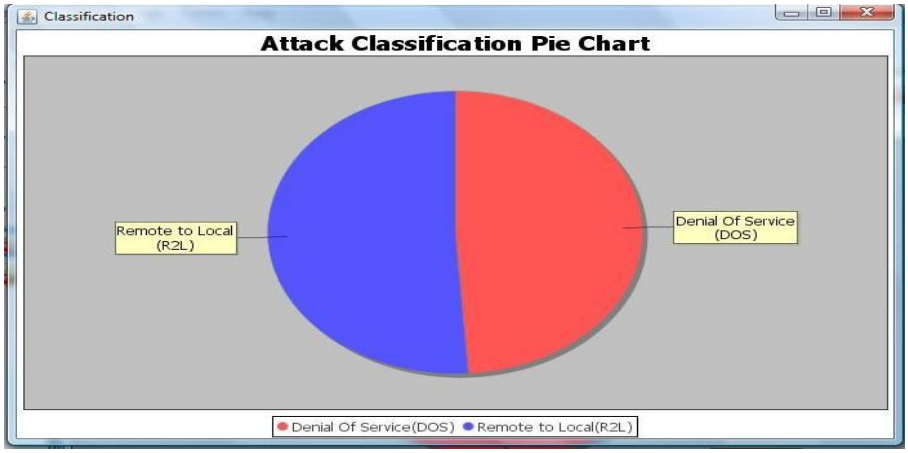


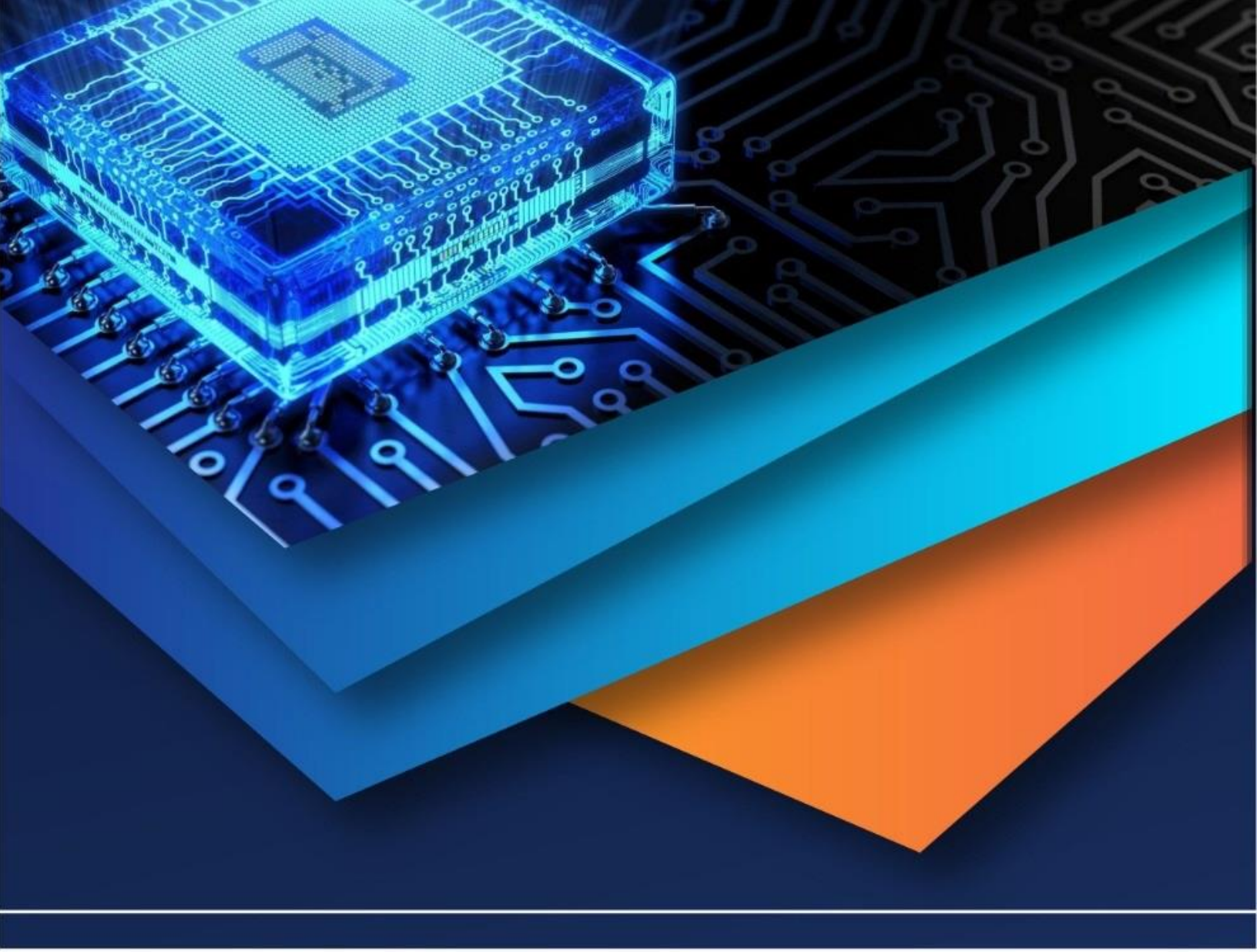
Fig. 8 Attack Classification

VI. CONCLUSION AND FUTURE WORK

This work depicts a novel methodology for arranging and discovering the meddling assaults with correctness more than 90% arranged as DOS, R2l, U2r, and Probe, utilizing the technique for paired arrangement and SVM. Notwithstanding our work goes past customary NIDS concerning the processing time/many-sided quality utilizing characteristic choice. With characteristic determination, we can attain the outcomes speedier keeping the same correctness in interruption recognition. As future work, we have considered the identification of a meddling strike utilizing the strategy for multi-class order. The grouping will be (typical example vs. [dos+probe+r2l+u2r]) or (strike design vs. other strike designs). Moreover, diminishment in expense for every test case was additionally accomplished utilizing the multi-classifier model. However, none of the machine taking in classifier calculations assessed was capable to perform identification of client to-root and remote-to-nearby ambush classes altogether (no more than 30% identification for U2r and 10% for remote-to-nearby classification). In determination, it will be sensible to declare that machine taking in calculations utilized as classifiers for the KDD 1999 Cup information sets don't offer any guarantees for identifying U2r and R2l ambushes inside the abuse identification setting.

REFERENCES

[1] Ali MH, Bahaa Abbas Dawood Al Mohammed, Alyani Ismail & Mohamad Fadli Zolkipli. A new intrusion detection system based on Fast learning network and swarm optimization. IEEE, 2018; 6:20255–61.
[2] Jamadar RA, Himani Gupta, Ankit Baghel & Rituraj. Study and analysis of hadoop based Network Intrusion Detection System, International Journal of Engineering and Science Invention. Dec 2017; 6(12):1−4.
[3] Divyatmika & Manasa Sreekesh. A Two-tier Network based Intrusion Detection System Architecture using Machine Learning Approach. 2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT), 3-5 March 2016. IEEE, Chennai, India; Mar. 2016. p. 42−47.
[4] FarzanehIzakShiri, BharnidharanShanmugam, NorbikBashahIdris “A parallel technique for improving the performance of Signature-based network intrusion detection system” IEEE Symposium, vol. 29, no. 12, pp. 3022-3025, 2008.
[5] Jigwen Tian and MeijuanGao “Network intrusion detection based on radial basis function neural network” IEEE Computer Society, vol. 27, no.16, pp. 1569-1584, 2009.
[6] JigwenTian and MeijuanGao “Network intrusion detection based on high speed and precise genetic algorithm” IEEE Trans vol. 32, no.25, pp. 700-752, 2010.
[7] XiaohuiBao and TianqiHou “Network intrusion detection based on support vector machine”. IEEE, vol.73, no.15, pp. 1356-1584, 2009.
[8] Robbinsommer and Vern Paxson “Network intrusion detection based on machine learning”. IEEE vol. 25, no. 1, pp. 69-75, 2007.
[9] Weiming Hu, JunGao, Yanguo Wang, Ou Wu “Online AdaBoost based parameterized methods for dynamic distributed network intrusion detection”.IEEE transactions on cybernetic vol 32 no. 2, pp. 210-215, 2014.
[10] Wonghirunsombat and Charnsripinyo “A centralized management framework of networking based intrusion detection and prevention system”.vol.44 SE-13, No. 2, pp.222-232. 2013.
[11] Zhongwenying and thomus “Signature searching in a networked collection of files” IEEE transactions on cybernetics Vol. 15, No. 8, pp.687-704 2014.
[12] Yang, Sezer, Litter, Pragonno” Multiattribute SCADA-Specific intrusion detection system for power networks” IEEE Transactions on Power Delivery vol. 29, pp. 1092 - 1102, 2014.
[13] Kang and Kim “Cyber threads and defense approaches in SCADA system”. IEEE vol. 27, no.16, pp. 1569- 1584, 2004.
[14] Choudhary, Tiwari and Kumar “Design and anomaly-based fuzzy intrusion detection system for packet dropping attack in mobile ad hoc networks” IEEE vol. 35, no.42, pp. 932-976, 2004.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY