International Journal of Engineering Research-Online
A Peer Reviewed International Journal
Articles available online http://www.ijoer.in

Vol.3., Issue.4., 2015
(July-Aug)

RESEARCH ARTICLE

ISSN: 2321-7758

INTRUSION DETECTION SYSTEM USING DATA MINING TECHNIQUES

KRISHAN KANT [1], SARITA BHAN[2], KAPIL SHRIVASTAVA [3]
[1]M.Tech Student, Department of CSE, Satya Group of Institutions,Palwal, India
[2]Assistant Professor, Department of CSE, Satya Group of Institutions,Palwal, India
[3]Assistant Professor, Department of CSE, Hindustan College of Science & Technology
Mathura, India

ISSN:2321-7758
International Journal of Engineering Research-Online
A peer-reviewed, open access International scientific journal
www.ijoer.in

ABSTRACT
Intrusion Detection Systems are challenging task for finding the user as normal user or attack user in any organizational information systems or IT Industry. The Intrusion Detection System is an effective method to deal with the kinds of problem in networks. Different classifiers are used to detect the different kinds of attacks in networks. In this paper, the performance of intrusion detection is compared with various neural network classifiers. In the proposed research the four types of classifiers used are Feed Forward Neural Network (FFNN), Generalized Regression Neural Network (GRNN), Probabilistic Neural Network (PNN) and Radial Basis Neural Network (RBNN). The performance of the full featured KDD Cup 1999 dataset is compared with that of the reduced featured KDD Cup 1999 dataset. The JAVA software is used to train and test the dataset and the efficiency and False Alarm Rate is measured. It is proved that the reduced dataset is performing better than the full featured dataset.

Keywords— Intrusion Detection, Neural Networks, KDD Dataset

INTRODUCTION

Over the Internet, the users are sharing their valuable information all over the world. Internet has also created numerous ways to compromise the stability and security of the systems connected with each other. The two kinds of mechanisms are static and dynamic. The static mechanisms such as firewalls and software updates provide a reasonable level of security and dynamic mechanisms such as intrusion detection systems. In the previous century, there was less number of intruders so the user can manage them easily from the known or unknown attacks. In present years the security is the most serious issue for securing the valuable information. Therefore either static mechanism or dynamic mechanism is required for protecting individual information despite the prevention techniques. The intrusion detection system is useful not only in detecting successful intrusions, but also in monitoring or preventing the attacks for timely countermeasures [1].

Intrusion detection can be defined as the process of identifying malicious behavior that targets a network and its resources. Malicious behavior is defined as a system or individual action which tries to use or access to computer system authorization (i.e., crackers) and the privilege excess of those who have legitimate access to the system (i.e., the insider threat).

The proliferation of heterogeneous computer networks serious implications for the intrusion detection problem. Foremost among these implications is the increased opportunity for unauthorized access that is provided by the network's connectivity. Intrusion detection is not an easy task due to the vastness of the network activity data and the need to regularly update the IDS to be adapted to unknown attack methods. Nowadays, completely protect a network from attacks is being a very hard task. Even heavily protected networks sometimes penetrated, and an Adaptive Intrusion Detection System seems to be essential and is a key component in computer and network security.

Intrusion detection attacks can be classified into two groups: Misuse or Signature based and Anomaly based Intrusion Detection. The misuse or signature based intrusion detection system detects the intrusion by comparing with its existing signatures in the database.

If the detecting attacks and signatures match, it is an intrusion. The signature based intrusions are called known attacks whenever the users are detecting the intrusion by matching with the signatures log files. The log file contains the list of known attacks detected from the computer system or networks. The anomaly based intrusion detection is called as unknown attacks and this attack is observed from network as it deviates from the normal attacks.

The intrusion detection systems are classified as Network based or Host based attacks. The network based attacks may be either misuse or anomaly based attacks. The network based attacks are detected from the interconnection of computer systems. Since the system communicates with each other, the attack is sent from one computer system to another computer system by the way of routers and switches.

The host based attacks are detected only from a single computer system and is easy to prevent the attacks. These attacks mainly occur from some external devices which are connected. The web based attacks are possible when systems are connected over the internet and the attacks can be spread into different systems through the email, chatting, downloading the materials etc. Nowadays many computer systems are affected from web based dangerous attacks.

PRELIMINARY INVESTIGATION

There are two general methods of detecting intrusions into computer and network systems: anomaly detection and signature recognition. Anomaly detection techniques establish a profile of the subject's normal behavior (norm profile), compare the observed behavior of the subject with its norm profile, and signal intrusions when the subject's observed behavior differs significantly from its norm profile.

Signature recognition techniques recognize signatures of known attacks, match the observed behavior with those known signatures, and signal intrusions when there is a match.

An IDS installed on a network is like a burglar alarm system installed in a house. Through various methods, both detect when an intruder/burglar is present. Both systems issue some type of warning in case of detection of presence of intrusion/burglar. Systems which use misuse-based techniques contain a number of attack descriptions, or 'signatures', that are matched against a stream of audit data looking for evidence of the modeled attacks. The audit data can be gathered from the network, from the operating system, or from application log files. Experimentation conducted in this research work is based on DARPA KDD'99 data set.

Many classifications exist in literature about intrusion detection [13][25]. The basic types of intrusion detection are host-based and network-based. Host-based systems were the first type of intrusion detection systems to be developed and implemented. These systems collect and analyze data that originate in a computer that hosts a service, such as a Web server. Once this data is aggregated for a given computer, it can either be analyzed locally or sent to a separate/central analysis machine. Instead of monitoring the activities that take place on a particular network, network-based intrusion detection analyzes data packets that travel over the actual network.

These packets are examined and sometimes compared with empirical data to verify their nature: malicious or benign. Because they are responsible for monitoring a network, rather than a single host, network-based intrusion detection systems tend to be more distributed than host-based intrusion detection system.

The two types of intrusion detection systems differ significantly from each other, but complement one another well. The network architecture of host-based is agent-based, which means that a software agent resides on each of the hosts that will be governed by the system. In addition, more efficient host based intrusion detection systems are capable of monitoring and collecting system audit trails in real time as well as on a scheduled basis, thus distributing both CPU utilization and network overhead and providing for a flexible means of security administration.

Intrusion Detection System (IDS)

Firewalls have been used for security in the networks since long but they can be easily deceived as a lot of techniques for deceiving the firewall have been developed. Tunneling is one of the techniques used to bypass the firewall; one can envelop message for a protocol inside some other message format. Another method is to route the illegitimate traffic through some other unauthorized route. We know that firewalls sniff the packets at the border i.e. between the networks, and have nothing to do with the traffic flowing inside the network. So it is not useful for the attacks generated from inside the network. An IDS identifies the attacks and protects the system like misuse and DoS attacks etc. An ID maintains an information system about the attacks after they had occurred in order to prevent such attacks in the future. The information gathered from the network and the network devices is compared with the predefined attack patterns for detecting misuse. IDS have the following three major components, as shown in figure 1.

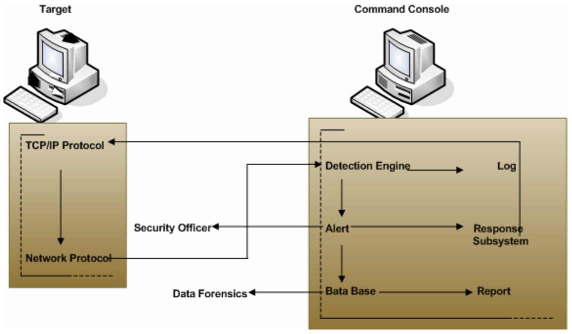


Fig. 1: Standard Network Intrusion Detection Architecture

- Sensors: sniff the network and detect the intrusion.
- Console: controls the sensors and monitors if some event is generated.
- Central Engine: records events generated by the sensor for future use and also compare detected events with the database and generate alarms.

Network based IDS have stronger detection mechanism to detect external intruders by comparing current behavior with already observed behavior. Host based system have weak real time response, because they need signatures of attacks for detection. This degrades their performance for real time attacks but at the same time they can perform well for long term attack (signatures of already occurred attack are generated which can detect similar attack for the next time of occurrence). On the other hand, network based systems just compare the current system state with observed state to detect real time attacks. Host based systems are excellent to calculate overall damages done by the attack, because it has the capability of event or kernel log analysis. This may give the idea of alteration in the system. Whereas network based IDS have weak damage assessment. Host based IDS have enough ability to detect suspicious behavior pattern due to log analysis. Whereas network based cannot do such analysis because it is mainly focused on the network activities (analyze network packets for anomalies).

Techniques in IDS

There are generally four techniques used in IDS. Each has its own limitation, as discussed below [8]:

Misuse Detection

All known attacks can be represented by some patterns. This detection scheme compares the patterns and also detects the similar patterns (different variants of the same pattern). It is different from virus detection because it detects similar patterns as well. It is a plus point of this scheme that it is strong for detecting the known patterns and also it can detect some of the unknown attacks, but still it cannot cope with all of the unknown attacks which may occur. Attack detection scheme may be categorized as follows:

a) Rule-Based Languages

In misuse detection, rule based approach is widely used. In this technique attacks are represented in different sets of rules. Rule sets are created and later compared with different attacks to detect presence of intrusion.

b) State Transition Analysis

Rule based system [14] requires highly skilled programming techniques to update the rules. Thus, state transition based scheme originated to overcome the drawbacks of rule based system. In state transition, attacks are symbolized as a series of events that are lead by the attacker having some initial state to the final state.

The states correspond to the targeted system that represents all the memory locations of the system as shown in figure 2. In this scenario it is assumed that the attacker must have some permission to access the network and all penetration guide to the acquisition of some ability that the attacker does not have prior to the attacks.

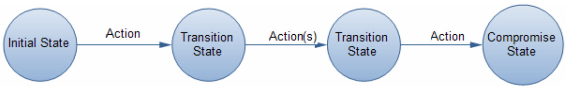


Fig. 2: The State Transition Diagram

c) Abstraction-Based Intrusion Detection

The major and common drawback of misuse detection approaches is that they all are written for their own specific environment and cannot work well for others. To address this problem abstraction based algorithms were introduced. The first attempt of abstraction based is adaptable real-time misuse detection system (ARMD). It is host based misuse detection system that provides language platform for signatures and methods that translate these signatures into monitoring program.

B. Anomaly detection

Anomaly detection is the comparison of a behavior with some observed behaviors in order to detect intrusion. It is stronger than misuse detection; because it has ability to detect unseen attack. Anomaly detection schemes are as follows:

a) Statistical Models

Statistical models are one of the earliest methods which are used for intrusion detection. In this model it is assumed that attacker behavior is different from the normal user, their statistical methods can be used to distinguish normal behavior to abnormal one. There are two statistical models which are used in intrusion detection. First one is the real time IDS having statistical component based on expert system.

It analyzes behavior of the network in normal mode and shortlists nodes whose behavior is found varying. The significant change or deviation from the expected behavior is flagged and treated as a potential intrusion.

Haystack on the other hand analyzes user activities according to four steps. Initially, it generates statistics based on user sessions namely session vectors. Next, it generates Bernoulli vector to characterize attributes which are not meant for that specific session. After that it assigns weights to intrusions types based on occurred frequency. Lastly, it generates suspicion quotient to represent how that session is suspicious as compared to other sessions for specific intrusion types.

b) Machine Learning Techniques

Machine learning based techniques help in independent identification and amalgamation of gather information based on models, either implicit or explicit to identify pattern analysis. The said information is marked to train the behavioral model accordingly for applying strict inquiry on resources so that misbehavior is identified dynamically.

c) Knowledge-based

Expert system approaches are widely used examples of knowledge based system. Expert system classifies audit data according to their rule sets. It involves multiple steps. Initially it identifies different classes and attributes from the trained data on the basis of which a set of classification rules is generated and parameters and functions are figured out. Lastly, the audit data is classified accordingly.

Proposed Work

The objective of the paper is to determine whether the given data set of captured packets belongs to the normal class of packets or to the anomaly class of packets. This is done by comparing the testing data set with the training data set. The system is generalized in a way that the user itself can decide and train the system based on his exclusive requirements of features. The system is taking a training data set as an input to train the system followed by a testing data set which will be classified as normal or anomaly.

The intrusion detection system has a critical role in detecting the intrusion in the real world. A number of methods and techniques have been proposed as many systems have been affected by a variety of intrusions. The various techniques used to detect the intrusions are data mining, neural network and statistical methods. In this related work, the various methods and techniques for detecting intrusion detection systems are discussed.

The Multivariate Statistical Analysis methods are used to determine the anomaly detection. The statistical methods are used to compare the performance of the system. The Hidden Markov Model is used to implement and determine the system call based anomaly intrusion detection. Conditional Random Fields and Layered Approach are addressed by the two issues of Accuracy and Efficiency. This approach demonstrates the high attack detection accuracy and high efficiency using Conditional Random Fields and Layered Approach. This approach uses KDD Cup ’99 intrusion detection data set for detecting the attacks.

Recurrent Neural Network model used with four groups of input features has been proposed as misuse-based IDS and the experimental results have shown that the reduced-size neural classifier has improved classification rates, especially for R2L attack.

The Genetic Algorithm is used to detect the intrusions in networks. It considers both temporal and spatial information of network connections during the encoding of the problem using Genetic Algorithm. The Genetic Algorithm is more helpful for identification of network anomalous behaviors.[20] and[31]

The Rough Set Neural Network Algorithm is used to reduce a number of computer resources required to detect an attack. The KDD Cup’99 data set is used to test the data and gives the better and robust result.[12] The various feature reduction techniques such as Independent Component Analysis, Linear Discriminate Analysis and Principal Component Analysis are used to reduce the computational intensity. KDD Cup 99 data set is used to reduce computation time and improve the accuracy of the systems.[33]

The Hierarchical Gaussian Mixture Model detects network based attacks as anomalies using statistical classification techniques. This model is evaluated by well known KDD99 data set. There are six classification techniques used to verify the feasibility and effectiveness. This technique is used to reduce the missing alarm and accuracy of the attack in Intrusion Detection System.[27]

Anomaly detection and analysis are based on the methods which describe the normal and abnormal traffic and accurately detect and classify various anomaly behaviors based on Correlation Coefficient Matrix. [7]The KDD 99 data set is used for training and testing the data.

Implementation Work

KDD Data Set

Software to detect network intrusions protects a computer network from unauthorized users, including perhaps insiders. The intrusion detector learning task is to build a predictive model (i.e. a classifier) capable of distinguishing between “bad” connections, called intrusions or attacks, and “good” normal connections. During the last decade, anomaly detection has attracted the attention of many researchers to overcome the weakness of signature-based IDSs in detecting novel attacks, and KDDCUP’99 [10] is the mostly widely used data set for the evaluation of these systems.

The KDD training dataset consist of 10% dataset that is approximately 494,020 single connection vectors each of which contains 41 features and is labeled with exact one specific attack type i.e, .either normal or an attack. Each vector is labeled as either normal or an attack, with exactly one specific attack type.

Deviations from ’normal behavior’, everything that is not ’normal’, is considered attacks. Attacks labeled as normal are records with normal behavior. A smaller version 10% training dataset is also provided for memory constrained machine learning methods. The training dataset has 19.69% normal and 80.31% attack connections.

KDD CUP 99 has been most widely used in attacks on network. The simulated attack falls in one of the following four categories [23]:

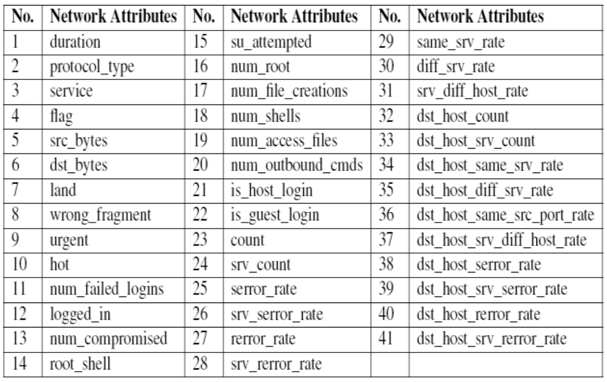
a) Denial of Service Attack (DOS): In this category the attacker makes some computing or memory resources too busy or too full to handle legitimate request, or deny legitimate users access to machine. DOS contains the attacks: ’neptune’, ’back’, ’smurf’, ’pod’, ’land’, and ’teardrop’.

b) Users to Root Attack (U2R): In this category the attacker starts out with access to a normal user account on the system and is able to exploit some vulnerability to obtain root access to the system. U2R contains the attacks:’ buffer_overflow’, ’load module’,’root kit and ’perl’.

c) Remote to Local Attack (R2L): In this category the attacker sends packets to machine over a network but who does not have an account on that machine and exploits some vulnerability to gain local access as a user of that machine. R2L contain the attacks:’warezclient’, ’ multi hop’, ’ ftp_write’, ’imap’, ’guess_passwd’, ’warez master’, ’spy’ and ’phf’

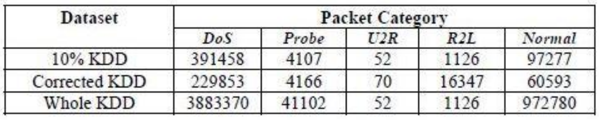
d) Probing Attack (PROBE): In this category the attacker attempt to gather information about network of computers for the apparent purpose of circumventing its security. PROBE contains the attacks:’portsweep’, ’satan’, ’nmap’, and ’ipsweep.

Fig. 3: Attributes of KDD Dataset



No.	Network Attributes	No.	Network Attributes	No.	Network Attributes
1	duration	15	su_attempted	29	same_srv_rate
2	protocol_type	16	num_root	30	diff_srv_rate
3	service	17	num_file_creations	31	srv_diff_host_rate
4	flag	18	num_shells	32	dst_host_count
5	src_bytes	19	num_access_files	33	dst_host_srv_count
6	dst_bytes	20	num_outbound_cmds	34	dst_host_same_srv_rate
7	land	21	is_host_login	35	dst_host_diff_srv_rate
8	wrong_fragment	22	is_guest_login	36	dst_host_same_src_port_rate
9	urgent	23	count	37	dst_host_srv_diff_host_rate
10	hot	24	srv_count	38	dst_host_serror_rate
11	num_failed_logins	25	serror_rate	39	dst_host_srv_serror_rate
12	logged_in	26	srv_serror_rate	40	dst_host_rerror_rate
13	num_compromised	27	rerror_rate	41	dst_host_srv_rerror_rate
14	root_shell	28	srv_rerror_rate		

Fig. 4: Sample Distribution in each type of KDD 99 Dataset Version attacks: ’neptune’, ’back’, ’smurf’, ’pod’, ’land’, and ’teardrop’.

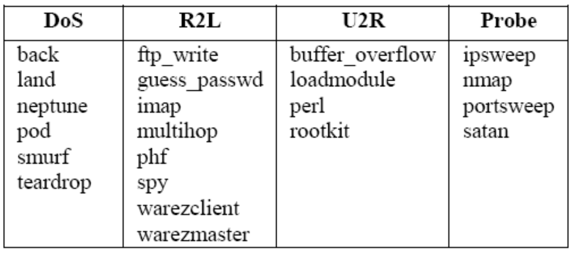


Dataset	Packet Category				
	DoS	*Probe*	*U2R*	*R2L*	*Normal*
10% KDD	391458	4107	52	1126	97277
Corrected KDD	229853	4166	70	16347	60593
Whole KDD	3883370	41102	52	1126	972780

Attribute-Relation File Format

The input that has been provided to our computer network intrusion detection system is in the form of attribute relation file format (ARFF). An ARFF (Attribute-Relation File Format) file is an ASCII text file that describes a list of instances sharing a set of attributes. ARFF files were developed by the Machine Learning Project at the Department of Computer Science of The University of Waikato for use with the Weka machine learning software.

Fig. 5: List of Attacks- Category wise



DoS	R2L	U2R	Probe
back land neptune pod smurf teardrop	ftp_write guess_passwd imap multihop phf spy warezclient warezmaster	buffer_overflow loadmodule perl rootkit	ipsweep nmap portsweep satan

ARFF files have two distinct sections- The first section is the Header information, which is followed the Data information. The Header of the ARFF file contains the name of the relation, a list of the attributes (the columns in the data), and their types. The ARFF Header section of the file contains the relation declaration and attributes declarations.

- The @relation Declaration: The relation name is defined as the first line in the ARFF file. The format is: @relation <relation-name>
- The @attribute Declarations: Attribute declarations take the form of an ordered sequence of @attribute statements. Each attribute in the data set has its own @attribute statement which uniquely defines the name of that attribute and it’s data type. The order the attributes are declared indicates the column position in the data section of the file. The format for the @attribute statement is: @attribute <attribute name><datatype>

The ARFF Data section of the file contains the data declaration line and the actual instance lines.

- The @data Declaration: The @data declaration is a single line denoting the start of the data segment in the file. The format is: @data
- The instance data: Each instance is represented on a single line, with carriage returns denoting the end of the instance.

K2 Algorithm

K2 learning algorithm showed high performance in many research works. The principle of K2 algorithm, proposed by Cooper and Herskovits, is to define a database of variables: (x1,...,xn), and to build an acyclic graph directed (DAG) based on the calculation of local score [10]. Variables constitute network nodes. Arcs represent "causal" relationships between - variables.

Algorithm K2 used in learning step needs:

- A given order between variables
- The number of parents, u of the node.

K2 algorithm proceeds by starting with a single node (the first variable in the defined order) and then incrementally adds connection with other nodes which can increase the whole probability of network structure, calculated using the g() function. A requested new parent which does not increase node probability cannot be added to the node parent.[6]

Where, for each variable x_i; r_i is the number of possible instantiations; N is the number of cases in the database; w_{ij} is the j_{th} instantiation of pa_i in the database; q_i is the number of possible instantiations for pai; N_{ijk} is the number of cases in D for which xi takes the value x_{ik} with pai instantiated to w_{ij}; N_{ij} is the sum of N_{ijk} for all values of k. Execution time is in the order $O(Nu^2n^2r)$ with r being the maximum value for r_i [10].

Naive-Bayes Classification Algorithm

The Bayesian Classification represents a supervised learning method as well as a statistical method for classification. Assumes an underlying probabilistic model and it allows us to capture uncertainty about the model in a principled way by determining probabilities of the outcomes. It can solve diagnostic and predictive problems.[30] This Classification is named after Thomas Bayes (1702-1761), who proposed the Bayes Theorem .

- Bayesian classification provides practical learning algorithms and prior knowledge and observed data can be combined.
- Bayesian Classification provides a useful perspective for understanding and evaluating many learning algorithms.
- It calculates explicit probabilities for hypothesis and it is robust to noise in input data.
- Naive Bayes classifiers are highly scalable, requiring a number of parameters linear in the number of variables (features/predictors) in a learning problem.

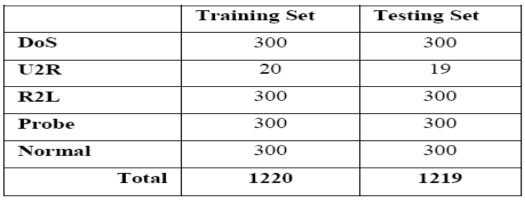
Maximum-likelihood training can be done by evaluating a closed-form expression [26] ,which takes linear time, rather than by expensive iterative approximation as used for many other types of classifiers. In the statistics and computer science literature, Naive Bayes models are known under a variety of names, including simple Bayes and independence Bayes[15].

All these names reference the use of Bayes' theorem in the classifier's decision rule, but naive Bayes is not (necessarily) a Bayesian method[15];Russell and Norvig note that "[naive Bayes] is sometimes called a Bayesian classifier, a somewhat careless usage that has prompted true Bayesians to call it the idiot Bayes model. Despite their naive design and apparently oversimplified assumptions, naive Bayes classifiers have worked quite well in many complex real world situations. In 2004, an analysis of the Bayesian classification problem showed that there are sound theoretical reasons for the apparently implausible efficacy of naïve Bayes classifiers. Still, a comprehensive comparison with other classification algorithms in 2006 showed that Bayes classification is outperformed by other approaches, such as boosted trees or random forests.

Experimental Results

The following sets of data can be used for training and testing the data from KDD Cup 1999 dataset. The Intrusion Detection techniques are used to detect the intrusions based on the KDD Cup 1999 dataset. These dataset contains 41 features in various types of attacks. By reducing 41 features into 13 features the accuracy has improved by 96.23% using the Probabilistic Neural Network. These Dataset can be applied using JAVA software and comparing these five Neural Network classifiers, the Probabilistic Neural Network proves the best accuracy.

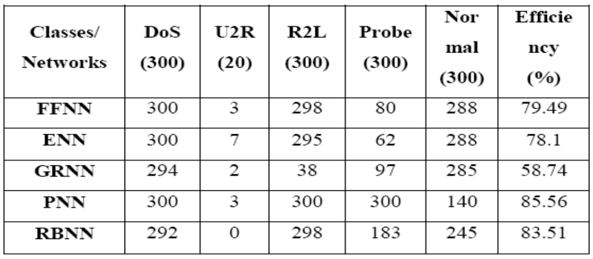
Fig. 5: Training and Testing Data Set



	Training Set	Testing Set
DoS	300	300
U2R	20	19
R2L	300	300
Probe	300	300
Normal	300	300
Total	1220	1219

The following table contains the five types of classes, five types of neural network classifiers used and the efficiency is measured. Figure 6 shows the classification of 41 featured dataset.

Fig. 6: Results for 41 Features Dataset



Classes/ Networks	DoS (300)	U2R (20)	R2L (300)	Probe (300)	Normal (300)	Efficiency (%)
FFNN	300	3	298	80	288	79.49
ENN	300	7	295	62	288	78.1
GRNN	294	2	38	97	285	58.74
PNN	300	3	300	300	140	85.56
RBNN	292	0	298	183	245	83.51

Conclusion and future scope

In today's scenario, to completely protect a network from attacks is a very hard task. Even heavily protected networks sometimes penetrated, and an Adaptive

Intrusion Detection System seems to be essential and is a key component in computer and network security. The difficulty in developing an intrusion detection system is to select features which are different for different organizations.

Thus, this intrusion detection project aims at dealing with this issue. The system will determine whether the given data set of captured packets belongs to the normal class of packets or to the anomaly class of packets. This is done by comparing the testing data set with the training data set. The system is generalized in a way that the user itself can decide and train the system based on his exclusive requirements of features. The system is taking a training data set as an input to train the system followed by a testing data set which will be classified as normal or anomaly. This intrusion detection system allows the user to input his desired selected features to train the system for building a model which detects the intruders. With the increasing amounts of traffic through our networks, performance is an important factor in any decision that is made regarding an organization's network.

It has been addressed, the problem of increased traffic through networks which was a factor affecting the performance of the IDS. It is safe to assume that hardware and software capabilities will match the increased throughput that we've been seeing lately, albeit at a higher price. Devices have been designed to circumvent the problem faced by NIDS in switched networks - they "sit invisibly between two networks and monitor all traffic exchanged, regardless of switches or hubs, while remaining immune to attack attempts.

REFERENCES

[1] AMMANN, P., AND OFFUTT, J. Introduction to software testing. Cambridge University Press, 2008.

[2] AYDIN, M. A., ZAIM, A. H., AND CEYLAN, K. G. A hybrid intrusion detection system design for computer network security. Computers & Electrical Engineering 35, 3 (2009), 517–526.

[3] BACH, J. Risk and requirements-based testing. Computer 32, 6 (1999), 113–114.

[4] BEIZER, B. Software testing techniques. Dreamtech Press, 2002.

[5] BINDER, R. V. Testing object-oriented systems: Objects, patterns, and tools, 1999.

[6] BOUHAMED, H., MASMOUDI, A., LECROQ, T., AND REBAÏ, A. A new approach for bayesian classifier learning structure via k2 algorithm. In Emerging Intelligent Computing Technology and Applications. Springer, 2012, pp. 387–393.

[7] CHEN, N., CHEN, X.-S., XIONG, B., AND LU, H.-W. An anomaly detection and analysis method for network traffic based on correlation coefficient matrix. In Scalable Computing and Communications; Eighth International Conference on Embedded Computing, 2009. SCALCOM-EMBEDDEDCOM'09. International Conference on (2009), IEEE, pp. 238–244.

[8] CLAPP, J. A. Software quality control, error analysis, and testing.William Andrew, 1995.

[9] DEBAR, H., BECKER, M., AND SIBONI, D. A neural network component for an intrusion detection system. In Research in Security and Privacy, 1992. Proceedings., 1992 IEEE Computer Society Symposium on (1992), IEEE, pp. 240–250.

[10] DEVARAJU, S., AND RAMAKRISHNAN, S. Detection of accuracy for intrusion detection system using neural network classifier. International Journal of Emerging Technology and Advanced Engineering 3 (2013).

[11] GERACI, A., KATKI, F., MCMONEGAL, L., MEYER, B., LANE, J., WILSON, P., RADATZ, J., YEE, M., PORTEOUS, H., AND SPRINGSTEEL, F. IEEE standard computer dictionary: Compilation of

IEEE standard computer glossaries. IEEE Press, 1991.

[12] GHALI, N. I. Feature selection for effective anomaly-based intrusion detection. International Journal of Computer Science and Network Security 9, 3 (2009), 285–289.

[13] GUPTA, K. K. Robust and efficient intrusion detection systems.

[14] GUPTA, K. K., NATH, B., AND KOTAGIRI, R. Layered approach using conditional random fields for intrusion detection. Dependable and Secure Computing, IEEE Transactions on 7, 1 (2010), 35–49.

[15] HAND, D. J., AND YU, K. Idiot's bayesU° not so stupid after all? International statistical review 69, 3 (2001), 385–398.

[16] HU, J., YU, X., QIU, D., AND CHEN, H.-H. A simple and efficient hidden markov model scheme for host-based anomaly intrusion detection. Network, IEEE 23, 1 (2009), 42–47.

[17] JAVITZ, H. S., AND VALDES, A. The sri ides statistical anomaly detector. In Research in Security and Privacy, 1991. Proceedings., 1991 IEEE Computer Society Symposium on (1991), IEEE, pp. 316–326.

[18] KAYA, F. Discretizing continuous features for naive bayes and c4. 5 classifiers. University of Maryland publications (2008).

[19] KIM, J., AND BENTLEY, P. The artificial immune model for network intrusion detection. In 7th European Conference on Intelligent Techniques and Soft Computing (EUFITŠ99), Aachen, Germany (1999), vol. 158.

[20] LI, W. Using genetic algorithm for network intrusion detection. Proceedings of the United States Department of Energy Cyber Security Group (2004), 1–8.

[21] MATHUR, A. P. Foundations of Software Testing, 2/e. Pearson Education India, 2008.

[22] NOEL, S., WIJESEKERA, D., AND YOUMAN, C. Modern intrusion detection, data mining, and degrees of attack guilt. In Applications of data mining in computer security. Springer, 2002, pp. 1–31.

[23] PANDA, M., AND PATRA, M. R. Network intrusion detection using naïve bayes. International journal of computer science and network security 7, 12 (2007), 258–263.

[24] PATTON, R. Software testing. Sams Pub., 2006.

[25] PUKETZA, N. J., ZHANG, K., CHUNG, M., MUKHERJEE, B., AND OLSSON, R. A. A methodology for testing intrusion detection systems. Software Engineering, IEEE Transactions on 22, 10 (1996), 719–729.

[26] RUSSELL, S., NORVIG, P., AND INTELLIGENCE, A. A modern approach. Artificial Intelligence. Prentice-Hall, Egnlewood Cliffs 25 (1995).

[27] SARASAMMA, S. T., ZHU, Q. A., AND HUFF, J. Hierarchical kohonenen net for anomaly detection in network security. Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on 35, 2 (2005), 302–312.

[28] SAVENKOV, R. How to become a software tester. Roman Savenkov, 2008.

[29] SHEIKHAN, M., JADIDI, Z., AND FARROKHI, A. Intrusion detection using reduced-size rnn based on feature grouping. Neural Computing and Applications 21, 6 (2012), 1185–1190.

[30] SHYARA TARUNA, R., AND HIRANWAL, M. S. Enhanced naïve bayes algorithm for intrusion detection in data mining.

[31] STEIN, G., CHEN, B., WU, A. S., AND HUA, K. A. Decision tree classifier for network intrusion detection with ga-based feature selection. In Proceedings of the 43rd annual Southeast regional conference-Volume 2 (2005), ACM, pp. 136–141.

[32] TAN, K. The application of neural networks to unix computer security. In Neural Networks, 1995. Proceedings., IEEE International Conference on (1995), vol. 1, IEEE, pp. 476–481.

[33] VENKATACHALAM, V., AND SELVAN, S. Intrusion detection using an improved competitive learning lamstar neural network. IJCSNS International Journal of Computer Science and Network Security 7, 2 (2007), 255–263.

[34] XIE, Y., AND YU, S.-Z. A large-scale hidden semi-markov model for anomaly detection on user browsing behaviors. Networking, IEEE/ACM Transactions on 17, 1 (2009), 54–65.

[35] YE, N., EMRAN, S. M., CHEN, Q., AND VILBERT, S. Multivariate statistical analysis of audit trails for host-based intrusion detection. Computers,IEEE Transactions on 51, 7 (2002), 810–820.