Volume 4, Issue 1, January 2014 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Anomaly-Based Intrusion Detection Systems Using Neural Network

Sheikh Kashif Ahmed, Prof. Sriram Yadav, Rajat Singh

Computer Science and Engineering,

RGPV, Bhopal (M.P.), India

Abstract—To improve network security different steps has been taken as size and importance of the network has increases day by day. Then chances of a network attacks increases Network is mainly attacked by some intrusions that are identified by network intrusion detection system. These intrusions are mainly present in data packets and each packet has to scan for its detection. This paper works to develop a intrusion detection system which utilizes the identity and signature of the intrusion for identifying different kinds of intrusions. As network intrusion detection system need to be efficient enough that chance of false alarm generation should be less, which means identifying as a intrusion but actually it is not an intrusion. Result obtained after analyzing this system is quite good enough that nearly 90% of true alarms are generated. It detect intrusion for various services like Dos, SSH, etc by neural network

Keywords—Anomaly Detection, Computer Networks, Intrusion Detection, Network Security.

1. INTRODUCTION

Providing network security for different web services on the internet, different network infrastructures, communications network many steps has been taken like encryption, firewall, and virtual private network etc. network Intrusion detection system is a major step among those. Intrusion detection field emerges from last few years and developed a lot which utilizes the collected information from different type of intrusion attacks and on the basis of those different commercial and open source software products come into existence to harden your network to improve network security of the different communication, service providing networks. As the number of network users and machine are increasing day by day to provide different kind of services and easiness for the smoothness of the world. But some unauthorized users or activities from different types of attackers which may internal attackers or external attackers in order to harm the running system, which are known as hackers or intruders, come into existence. The main motive of such kind of hacker and intruders is to bring down bulky networks and web services[1]. Due to increase in interest of network security of different types of attacks, many researchers has involved their interest in their field and wide variety of protocols as well as algorithm has been developed by them, In order to provide secure services to the end users. Among different type of attack intrusions is a type of attack that develop a commercial interest. Intrusion detection system is introduced for the protection from intrusion attacks.

From the above discussion we can conclude the main aim of the network Intrusion detection system is to detect all possible intrusion which perform malicious activity, computer attack, spread of viruses, computer misuse, etc. so a network intrusion detection system not only analyses different data packets but also monitor them that travel over the internet for such kind of malicious activity[2]. So the smooth running of overall network different server has to settle on the whole network which act as network intrusion detection system that monitor all the packets movements and identify their behavior with the malicious activities. One more kind of network Intrusion detection system is developed that can be installed in a centralized server which also work in the similar fashion of analyzing and monitoring the different packet data units for their network intrusion behavior. Network Intrusion detection system can be developed by two different approaches which can be named as signature based and anomaly based. In case of signature based Network Intrusion detection system it develops a collection of security threat signature. So according to the profile of each threat the data stream of different packets in the network are identified and the most matching profile is assigned to that particular packets. If the profile is malicious then that data packet comes under intrusion and it has to remove from the network in order to stop his unfair activities

2. RELATED WORK

Lakhina et al. proposed a detection method that detects and diagnoses anomalies in large scale networks. First, their approach monitors the traffic using a matrix in which each cell represents the traffic volume of a link of the network at a certain time interval. Second, the main behavior of the traffic is extracted from the matrix with the principal component analysis (PCA) and anomalies are detected in residual traffic. Finally, the origin and destination nodes of the network that are affected by the anomalous traffic are identified and reported.

The KDD'99 has been the most wildly used data set for the evaluation of anomaly detection methods is prepared by Stolfo et al, based on the data captured in DARPA'98 IDS evaluation program [11]. Agarwal and Joshi [12] proposed a Two stage general to specific framework for learning a rule based model (PNrule) to learn classifier models on a data set that has widely different class distributions in the training data. The proposed PN rule evaluated on KDD dataset reports high detection rate. Yeung and Chow proposed a novelty detection approach using no parametric density estimation based on Parzen window estimators with Gaussian kernels to build an intrusion detection system using normal

data. This novelty detection approach was employed to detect attack categories in the KDD dataset. In 2006, XinXu et al. presented a framework for adaptive intrusion detection based on machine learning.

Lee et al., introduced data mining approaches for detecting intrusions. Data mining approaches for intrusion detection include association rules that based on discovering relevant patterns of program and user behaviour. Association rules , are used to learn the record patterns that describe user behaviour. These methods can deal with symbolic data and the features can be defined in the form of packet and connection record details. However, mining of features is limited to entry level of the packet and requires the number of records to be large and low diversity in data; otherwise they tend to produce a large number of rules which increases the complexity of the system[3]. Data clustering methods such as the kmeans and the fuzzy cmeans have also been applied extensively for intrusion detection. One of the main drawbacks of clustering technique is that it is based on calculating numeric distance between the observations and hence the observations must be numeric.

Observations with symbolic features cannot be easily used for clustering, resulting in inaccuracy. In addition, the clustering methods consider the features independently and are unable to capture the relationship between different features of a single record which further degrades attack detection accuracy. Naive Bayes classifiers have also been used for intrusion detection. However, they make stark independence assumption between the features in an observation resulting in lower attack detection accuracy to detect intrusions when the features are correlated, which is often the case for intrusion detection..

3. BACKGROUND

a) Attack types

The easy and common criterion for describing all computer network attacks and intrusions in the respective literature is to the attack types [1]. In this chapter, we categorize all computer attacks into the following classes:

- **Denial of Service (DoS) attacks:**

Denial of Service (DoS) attacks mainly attempt to "shutdown a whole network, computer system, any process or restrict the services to authorized users" [2]. There are mainly two types of Denial of Service (DoS) attacks:

- operating system attacks
- networking attacks

In denial of service attack, operating system attacks targets bugs in specific operating system and then can be fixed with patch by patch, on the other hand networking attacks exploits internal limitation of particular networking protocols and specific infrastructure.

- **SSH**

Secure Shell is a protocol that provides authentication, encryption and data integrity to secure network communications. Implementations of Secure Shell offer the following capabilities: a secure command-shell, secure file transfer, and remote access to a variety of TCP/IP applications via a secure tunnel. Secure Shell client and server applications are widely available for most popular operating systems.

The secure shell protocol allows users to log in remote terminals in a secure fashion[4]. It does this by performing authentication using a passphrase and a public keyring, and subsequently encrypts all information transmitted or received, guaranteeing its confidentiality and integrity.

- **Probing (surveillance, scanning):**

Probing (surveillance, scanning) attacks scan the networks to identify valid IP addresses and to collect information about them (e.g. what services they offer, operating system used). Very often, this information provides a tacker with the list of potential vulnerabilities that can later be used to perform an attack against selected machines and services.

These attacks use known vulnerabilities such as buffer overflows [8] and weak security points for breaking into the system and gaining privileged access to hosts. Depending upon the source of the attack (outside attack vs. inside attack), the compromises can be further split into the following two categories:

- **R2L (Remote to Local)**:

Attacks, where an attacker who has the ability to send packets to a machine over a network (but does not have an account on that machine), gains access (either as a user or as the root) to the machine. In most R2L attacks, the attacker breaks into the computer system via the Internet[5]. Typical examples of R2L attacks include guessing passwords (e.g. guest and dictionary attacks) and gaining access to computers by exploiting software vulnerability (e.g. phf attack, which exploits the vulnerability of the phf program that allows remote users to run arbitrary commands on the server).

- **U2R (User to Root):**

Attacks, where an attacker who has an account on a computer system is able to misuse/elevate her or his privileges by exploiting vulnerability in computer mechanisms, a bug in the operating system or in a program that is installed on the system. Unlike R2L attacks, where the hacker breaks into the system from the outside, in U2R compromise, the local user/attacker is already in the system and typically becomes a root or a user with higher privileges. The most common U2R attack is buffer overflow, in which the attacker exploits the programming error and attempts to store more data into a buffer that is located on an execution stack.

b) KDD' 99 Dataset

KDD'99 Dataset The KDD'99 dataset includes a set of 41 features derived from each connection and a label which specifies the status of connection records as either normal or specific attack type. The list of these features can be found in [21]. These features had all forms of continuous, discrete with significantly varying ranges falling in four categories:

1**. Basic Features**: Basic features can be derived from packet headers without inspecting the payload[6].

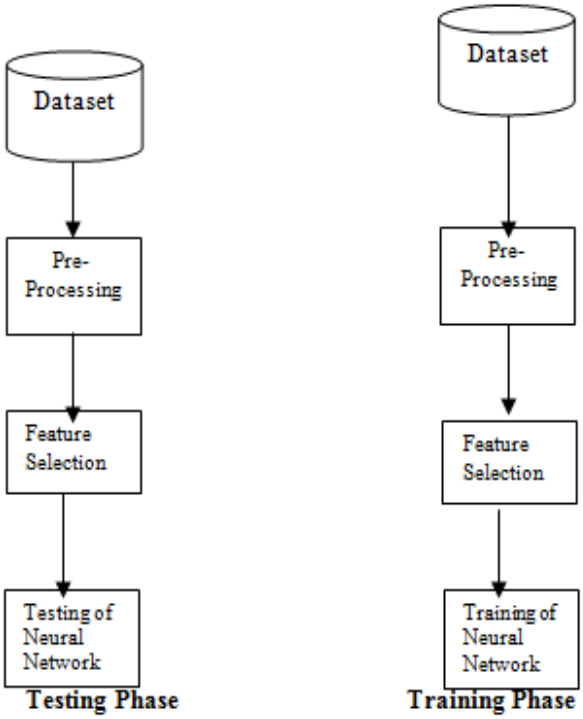
2.**Content Features**: Domain knowledge is used to assess the payload of the original TCP packets. This includes features such as the number of failed login attempts.

3. **Time4based Traffic Features**: These features are designed to capture properties that mature over a 2 second temporal window. One example of such a feature would be the number of connections to the same host over the 2 second interval.

4. **Host4based Traffic Features**: Utilize a historical window estimated over the number of connections. Time based and Host based traffic described as a Traffic features in KDD'99. Likewise, attacks fall into four main categories: DoS, R2L, U2R, Probe.

4. METHODOLOGY

Whole work is divide into three module, first is Pre-processing, second is training, third is testing. In order to make the analysis better feature vector of each class is prepare for training the neural network of the current updated dataset sessions.



Module 1. Pre-Processing

In order to increase the efficiency of the work dataset should be pre-process as the Preprocessing of Raw Dataset Instead of direct input of raw dataset to selected classifiers; raw dataset is preprocessed in different ways to overcome different issues like training overhead, classifier confusion, false alarms and detection rate ratios. Separating feature space from one another is very necessary and arrange in vector. Let us consider single vector of the dataset
{0,tcp,ftp_data,SF,491,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.00,0.00,0.00,150,25,0.17,0.03,0.17,0.00,0.00,0.00,0.05,0.00,normal,20}

In above vector presence of comma ',' and discarding symbolic characters that are of three kind s of symbolic features (tcp, ftp_data and SF etc.) in feature space of 41 features. As symbolic values are not of interest to our research, these three feature vectors are discarded to get the feature space.

So after the preprocessing the obtain vector contain only the numeric numbers[7]. While the last word represent the class which specify that a session is either a intrusion or not.

Module 2. Training

Feature selection is an important factor in NID. Since, the large numbers of features that can be monitored taking into account the large variety of possible values especially for continuous feature even for a small network[8]. For ID purpose, which is truly useful and reliable, which are significant features or less significant features and which may be useless?. The questions are relevant because the elimination of insignificant and useless features from audit data will enhance the accuracy of detection while speeding up the computation, thus will improve the overall performance of our proposed work for detecting intrusions. So, the main concentration is on selecting significant features.

Now the obtain vector is contain two important feature for selecting the features, first is the pattern of the different type of class in numeric formsuch as {491 , 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.00,0.00,0.00,150,25,0.17,0.03,0.17,0.00,0.00,0.00,0.05,0.00} and other is the class name such as {normal}. In the similar fashion different pattern of same class are collect in the single vector and use them to decide the kind of attack or normal network.

In order to efficiently detect anomaly in the network for intrusion detection following algorithm is implemented:

Algorithm start with the following inputs DataSet (Ds) number of vector space (n), Number of iteration for neural (N) Network.

```
Vs←Load_dataset(Ds, n)
// For Creating the feature vector
Pv←Pre-Process (Vs)
Loop I = 1: Pv
Group Pvclasswise C
Fv{j} ←Pv(I)
End Loop
Tn←Neural_network(Fv, N)
```

In above algorithm

Vs: Raw feature Vector, Pv: Pre-Processed Vector

Fv: Feature Vector, Ci :Class index Vector for different attack class, Tn: Trained Neural Network

For Training the neural network proper dataSet feature is required as the different class has different pattern set which have different value set. On the basis of these values neurons of the network will adjust there weight. Fv the feature vector is grouped during the feature collection steps of the different type of class which is matched, in the network. Finally Tn (Trained neural network) is obtained.

Module 3. Testing

For testing following are the parameter to be pass: Testing Dataset size Ds and Trained neural network Tn.

```
Testing(Ds, Tn)
Pv←Pre-Process (Ds)
Loop I = 1: Pv
Fv(I) ←Pv(I)  // Collect numeric feature
End Loop
Rc←Tn(Fv)  // Pass feature in Trained network
```

In above Testing Algorithm Rc : Resulting Class

As for testing the trained network dataset is again required with different vector, of different or may be of same pattern of the classes. Here it also need to make the feature vector of all the vector for testing from the neural network, but only numeric feature is collect in the Fv then as per training the values of the network is obtained that the input vector is belong to which class. feature is give as input which will specify the corresponding class[9]. At the end in order to evaluate the results it is necessary to check that the specified class is correct or not so each Rc resulting class is compare with the attach class of the numeric feature like {normal}.

5. EXPERIMENT AND RESULTS

In order to implement above algorithm for intrusion detection system MATLAB is use, where KDD99 dataset is use of different size.

Evaluation Parameter

To test our result this work use following measures the evaluation parameter which specify that either the trained neural network effectively identify the intrusion session from the normal one. One more important work of this network is to specifically identify the intrusion type. So the parameter are Precision, Recall and F-score.

Precision = true positives / (true positives+ false positives)

Recall = true positives / (true positives +false negatives)

F-score = 2 * Precision * Recall / (Precision + Recall)

Table a. Different size dataset and type of intruder obtained.

DataSet Size	Network Type
1,000	Normal , U2R
5,000	Normal , U2R
10,000	Normal , U2R, DOS
15,000	Normal , U2R, DOS, R2L
25,000	Normal , U2R, DOS, R2L, SSH

By evaluating the system at different size of dataset it was obtained that different type of attack has been identified which are intruder for the system.

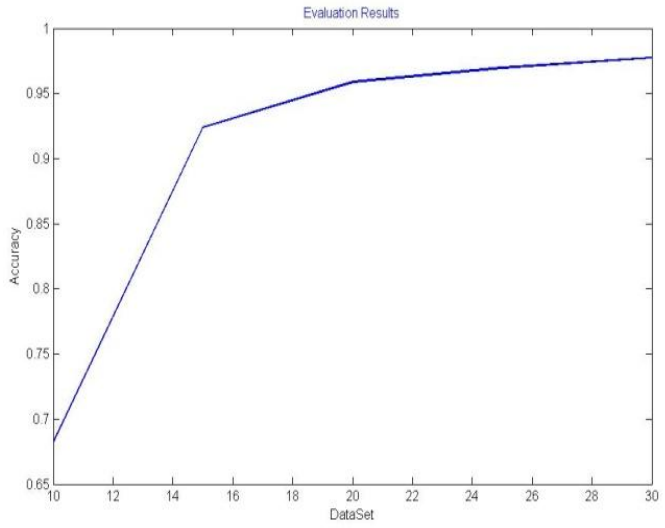
DataSet Size	Precision	Recall	F-score
1000	0.996	0.998	0.997
5,000	0.9984	0.9992	0.9992
15,000	0.9632	0.7535	0.7533
25,000	0.9387	0.8441	0.8407
30,000	0.8813	0.879	0.88015

Table b. Different dataset and corresponding values.

It is clear from above table that evaluation of Algorithm for different Dataset Size has effective values as the precision is always above 93% which shows that how effective the system for intrusion detection from the normal traffic. It is seen in the table that all the values of F-Measure is increasing by the when the type of intruder get sufficiently trained by the dataset. So after 15000 session of dataset one can get the results which has more number of attack and they are also identifiable.

In order to make the better evaluation for this work one more parameter has introduced that is accuracy of the class of the intrusion. Accuracy of the work is calculate by:

Accuracy = (true positives +false negatives)/(Total_Normal + Total_Intrusion)



observed that accuracy values continuously increase as the data Size for training is increases. It has seen that at smaller data size for training some time results of accuracy is nearly 0.99 and above. But that was not true for all as it not cover all type if intrusion attacks. So testing with small size may produce unexpected result.

6. CONCLUSION

Network security is one of the most important non-functional requirements in a system. Over the years, many software solutions have been developed to enhance network security and this paper provides an efficient system which has been a promising one for detecting intrusion of different kind where, one can get the detail of the class of attack as well. Results shows that all type of attack are accurately identified by the system as the accuracy value is above 96%[10]. In future it need to be improved by putting data on the unsupervised network, so it automatically update the new behavior of the intruder.

REFERENCES

.[1] K. Kendall, A Database of Computer Attacks for the Evaluation of Intrusion Detection Systems, Massachusetts Institute of Technology Master's Thesis, 1998.

[2] D. Marchette, Computer Intrusion Detection and Network Monitoring, A Statistical Viewpoint. New York, Springer, 2001.

[3] J. Mirkovic, G. Prier and P. Reiher, Attacking DDoS at the Source, 10th IEEE International Conference on Network Protocols, November 2002

[4] C. Cheng, H.T. Kung and K. Tan, Use of Spectral Analysis in Defense Against DoS Attacks, In Proceedings of the IEEE GLOBECOM , Taipei, Taiwan, 2002

[5] H. Burch and B. Cheswick, Tracing Anonymous Packets to Their Approximate Source, In Proceedings of the USENIX Large Installation Systems Administration Conference, New Orleans, LA, 319-327, December 2000.

[6] A.D. Keromytis, V. Misra and D. Rubenstein, SoS: Secure Overlay Services, In Proceedings of the ACM SIGCOMM Conference, Pittsburgh, PA, 61-72, August 2002

[7] S. Robertson, E. Siegel, M. Miller and S. Stolfo, Surveillance Detection in High Bandwidth Environments, In Proceedings of the 3rd DARPA Information Survivability Conference and Exposition (DISCEX 2003) , Washington DC, April 2003.

[8] CERT® Advisory CA-2003-25 Buffer Overflow in Sendmail, http://www.cert.org/ advisories/CA-2003-25.html, September, 2003.

[9] C. Cowan, C. Pu, D. Maier, H. Hinton, J. Walpole, P. Bakke, S. Beattie, A. Grier and P. Zhang, StackGuard: Automatic Adaptive Detection and Prevention of Buffer Overflow Attacks, In Proceedings of the 7th USENIX Security Symposium, San Antonio, TX, 63-77

[10] CERT® Advisory CA-2000-14 Microsoft Outlook and Outlook Express Cache Bypass Vulnerability, http://www.cert.org/advisories/CA-2000-14.html, July 2000

[11] Leonid Portnoy ,EleazarEskin and Stolfo, "Intrusion Detection with Unlabeled Data Using Clustering" Department of Computer Science, Columbia University, Newyork, NY 10027

[12] R. Agarwal, and M. V. Joshi, "PNrule: A New Framework for Learning Classifier Models in Data Mining", Technical Report TR 00-015, Department of Computer Science, University of Minnesota, 2000.

