ANOMALY-BASED NETWORK INTRUSION DETECTION SYSTEMS BASED ON NEURAL NETWORK

RACHNA NAGDEV
Radharaman Institute of Technology Bhopal, Computer Science dept.Bhopal, India
(rachna.nagdev@gmail.com)

ANURAG JAIN
Radharaman Institute of Technology Bhopal, HOD Computer Science dept Bhopal , India
(anurag.akjain@gmail.com)

ABSTRACT: As a dimension and importance of the network has increases day by day. Then chances of a network attacks as also increases. So to enhance network security different steps has been taken. Network is mainly attacked by some intrusions which can be identified by network intrusion detection system. Many types of network intrusion detection system which utilizes the identity and signature of the intrusion. These intrusions are mainly contained in data packets and each packet has to scan for its detection. This paper works to develop a intrusion detection system in the similar fashion of identifying signature or patterns of different types of intrusions. As anomaly detection system has to face different problem of false alarm generation 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 85% of true alarms are generated**.**

KEYWORDS: Computer Networks, Network Security, Anomaly Detection, Intrusion Detection.

INTRODUCTION

As the amount of network users and machine are increasing daily to offer different kind of services and easiness for the smoothness of the entire 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. 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.

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. 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. 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. An additional 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 different packet data units for his or her 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.

RELARED WORK

The KDD'99 has been probably 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 principle 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 [13] proposed an uniqueness detection approach using no parametric density estimation predicated on Parzen window estimators with Gaussian kernels to construct an intrusion detection system using normal data. This novelty detection approach was employed to detect attack categories in the KDD dataset. In 2006, Xin Xu et al. [14] presented a construction for adaptive intrusion detection predicated on machine learning. Lee et al. [15], introduced data mining approaches for detecting intrusions. Data mining approaches for intrusion detection include association rules that centered on discovering relevant patterns of program and user behavior. Association rules [16], are used to learn the record patterns that describe user behavior. These methods can cope with symbolic data and the features can be defined in the form of packet and connection record details. However, mining of features is limited by entry degree of the packet and requires the number of records to be large and low diversity in data; otherwise they tend to generate a large amount of rules which escalates the complexity of the machine [17]. Data clustering methods including the kmeans and the fuzzy cmeans have already been applied extensively for intrusion detection. One of the main drawbacks of clustering technique is that it is based on calculating numeric distance involving the observations and hence the observations must certanly be numeric**.**

Observations with symbolic features can't be easily useful for clustering, causing inaccuracy. Additionally, the clustering methods consider the features independently and cannot capture the partnership between different features of a single record which further degrades attack detection accuracy. Naive Bayes classifiers have been useful for intrusion detection [18]. However, they make stark independence assumption involving the features in a declaration causing lower attack detection accuracy to detect intrusions once the features are correlated, which will be the case for intrusion detection.

Decision trees have already been useful for intrusion detection [18]. Your decision trees select the most effective features for every single decision node throughout the construction of the tree centered on some well defined criteria. One particular criterion is by using the information gain ratio that is used in C4.5. Decision trees generally have very top speed of operation and high attack DR. The investigation ers in discussed the usage of ANNs for NID. Though, the neural networks could work effectively with noisy data, they might need massive amount data for training and it's often hard to pick the perfect architecture for a neural network. Support vector machines have already been useful for detecting intrusions. Support vector machines map real valued input feature vector to a higher diversity in feature space through nonlinear mapping and can provide realtime detection capability, deal with large diversity of data. Sen. [19] designed of a distributed IDS is proposed that consists of a small grouping of autonomous and cooperating agents. The machine is capable of identifying and isolating compromised nodes in the network thereby introducing.

BACKGROUND

A). ATTACK TYPEThe easy and common criterion for describing all computer network attacks and intrusions in the respective literature is always 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 an entire network, computer system, any process or restrict the services to authorized users” [2]. 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 may be fixed with patch by patch, on the other hand networking attacks exploits internal limitation of particular networking protocols and specific infrastructure.

Probing (Surveillance,Scanning):

Probing (surveillance, scanning) attacks scan the networks to identify valid IP addresses and to get information about them (e.g. what services they offer, operating system used). Often, these records supplies a tacker with the list of potential vulnerabilities that will later be used to execute an attack against selected machines and services.

These attacks use known vulnerabilities such as for example buffer overflows [8] and weak security points for breaking into the system and gaining privileged access to hosts. Dependant on the origin of the attack (outside attack vs. inside attack), the compromises could be further split into the next two categories:

R2L(Remote To Local)

Attacks, where an attacker who has the capability to send packets to a device over a network (but does not need an account on that machine), gains access (either as an individual or while the root) to the machine. Generally in most R2L attacks, the attacker breaks into the computer system via the Internet. Typical samples 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 which allows remote users to operate arbitrary commands on the server).

U2R (User To Root)

Attacks, where an attacker who has an account on some type of computer system can misuse/elevate her or his privileges by exploiting a vulnerability in computer mechanisms, an insect in the os or in an application that is installed on the system. Unlike R2L attacks, where the hacker breaks into the machine from the surface, in U2R compromise, the area user/attacker has already been in the machine and typically becomes a root or a consumer with higher privileges. The most frequent U2R attack is buffer overflow, in that your attacker exploits the programming error and attempts to store more data into a buffer that is situated on an execution stack.

B). KDD' 99 DATASET

KDD'99 Dataset The KDD'99 dataset includes a couple of 41 features produced from each connection and a brand 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 types of continuous, discrete with significantly varying ranges falling in four categories:

1. Basic Features: Basic features could be produced from packet headers without inspecting the payload.
2. Content Features: Domain knowledge is used to gauge the payload of the initial TCPpackets. Including features such as for instance how many failed login attempts.
3. Time4based Traffic Features: These features are designed to capture properties that mature over a 2 second temporal window. An example of this kind of feature will be the number of connections to exactly the same host over the 2 second interval.
4. Host4based Traffic Features: Start using a historical window estimated over how many connections. Time based and Host based traffic referred to as a Traffic features in KDD'99. Likewise, attacks fall under four main categories: DoS, R2L, U2R, Probe.

Table 1: KDD dataset was employed here and this sample distributed

Type	Quantity of Samples
Normal	97227
DoS	39145
Probe	4107
R2L	1126
U2R	52

c). PRE-PROCESSING

To be able to increase the efficiency of the work dataset should really be pre-process because the Preprocessing of Raw Dataset As opposed to 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 each other is quite necessary and arrange in vector. Let's 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 d**iscarding symbolic characters which can be of** three kind s of symbolic features (tcp, ftp_data and SF etc.) in feature space of 41 features. As symbolic values aren't of interest to the research, these three feature vectors are discarded to obtain the feature space.So after the preprocessing the obtain vector is {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} where all element are require for dataset analysys.

d). FEATURES SELETION

Feature selection is an important element in NID. Since, the large numbers of features which can be monitored considering the large variety of possible values particularly for continuous feature even for a small network. For ID purpose, which will be truly useful and reliable, which are significant features or less significant features and which might be useless ?.The questions are relevant as the elimination of insignificant and useless features from audit data will boost the accuracy of detection while speeding up the computation, thus will improve the entire performance of our proposed benefit 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.

e). TRAINING ALGORITHM

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.

```
Training(Ds, N, n)
Vs←Load_dataset(Ds, n)
// For Creating the feature vector
Pv ←Pre-Process (Vs)
Loop I = 1: Pv
    Loop J = 1:Ci
        If Isequal( Pv(I), Ci(J))
            Fv{j} ← Pv(I)
        End If
    End Loop
End Loop
Tn←Feedforward_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 36 different values. On the basis of this 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.

Testing Algorithm

For testing following are the parameter to be pass: Dataset size Ds, number of vector to be use for testing (n) and Trained neural network Tn.

```
Testing(Ds, Tn, n)
Vs🡐Load_dataset(Ds)

Pv 🡐Pre-Process (Vs)
Loop I = 1: Pv
    Fv(I) 🡐 Pv(I)  // Collect numeric feature
End Loop
Rc🡐Tn(Fv)  // Pass feature in Trained network
Loop I = 1: Pv
    If Isequal( Pv(I), Rc(I))
        TP = TP + 1;
    Otherwise
        TN = TN + 1
    End If
End Loop
```

In above Testing Algorithm
Rc : Resulting Class
TP : True Positive
TN : True Negative

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. Such as {491,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.00,0.00,0.00,150,25,0.17,0.03,0.17, 0.00,0.00,0.00,0.05,0.00}feature is give as input which will specify the corresponding class. At the conclusion to be able to evaluate the results it is necessary to check on that the specified class is correct or not too each Rc resulting class is match up against the attach class of the numeric feature like normal.

EXPERIMENT AND RESULT

In order to implement above algorithm for intrusion detection system MATLAB is use, where dataset is use of different size. It was found that as the data size increase numbers of different class also increase as during 1000 to 5000 only two classes were found in dataset 'normal' 'u2r'.

While increasing the size will increase the different class such as by working on 25,000 data size following classes of attack were found 'normal' 'dos' 'probe' 'r2l' 'u2r'.

To test our result this work use following measures the accuracy of the writing mining approach, that's to state 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 2: Different dataset and corresponding values

DataSet Size	Precision	Recall	F-score
10,000	0.8870	0.7889	0.7736
15,000	0.9672	0.7545	0.7563
20,000	0.8528	0.8678	0.8083
25,000	0.9387	0.8041	0.8437

Evaluation of Algorithm for different Data Size from above table (b) it has observed that F-Score 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 F-score was above 0.9 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.

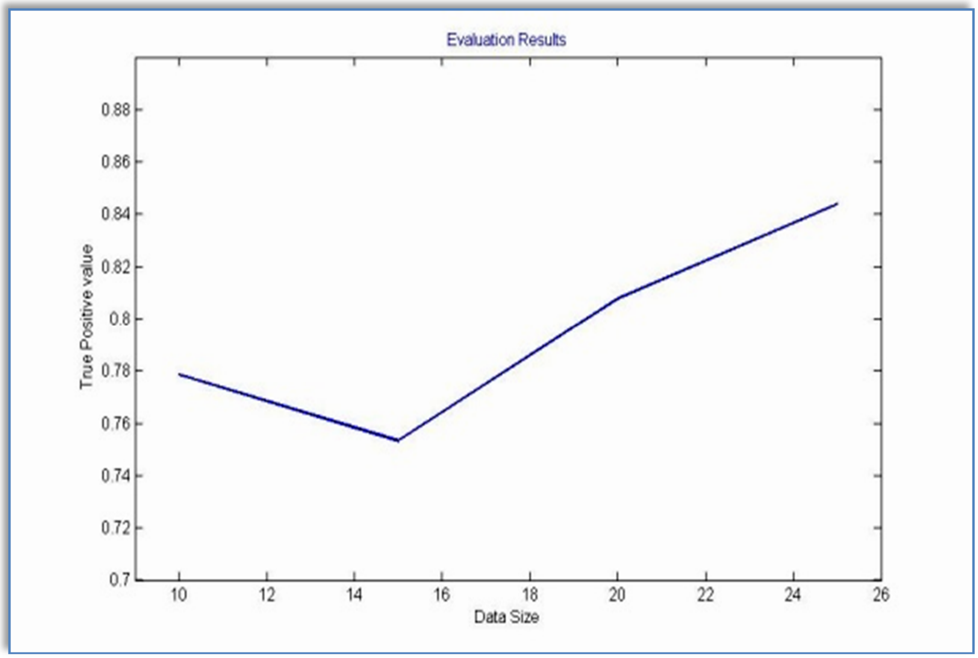

Fig 1: Data size (in thousand scales) Vs True positive values

From above table (b) and graph fig(a) it has found that as the training data size increase the true positive values is also increase so after 15000 training session a continuous growing graph is obtain

which tends towards one. As shown in figure 0.844 true positive values are obtain against 25000. So overall detection is good enough as it cover almost each class of different attack.

CONCLUSION

In this paper, IDS tool is develop for effectively identify the different intrusion of any class. Here a neural network is trained by learning the behavior of the different intrusion feature vector, it is obtained after testing that this system can efficiently detect attacks with 85 percent accuracy. One more valuable information is obtain from the system is that network works better for training vector of more then 25000 vector space. In future as this work utilizes only KDD’99 dataset, while there are other dataset as well for learning the feature and detect different intrusion.

REFERENCES

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

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

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

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

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.

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

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.

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

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

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

Leonid Portnoy ,Eleazar Eskin and Stolfo, “Intrusion Detection with Unlabeled Data Using Clustering” Department of Computer Science, Columbia University, Newyork, NY 10027

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.

Dit-Yan Yeung, Calvin Chow, "Parzen-Window Network Intrusion Detectors," icpr, vol. 4, pp.40385, 16th International Conference on Pattern Recognition (ICPR'02) - Volume 4, 2002

Xin Xu, Adaptive Intrusion Detection Based on Machine Learning: Feature Extraction, Classifier Construction and Sequential Pattern Prediction, Institute of Automation, College of Mechantronics Engineering and Automation, National University of Defense Technology, Changsha, 410073, P.R.China, International Journal of Web Services Practices, Vol.2, No.1-2 (2006), pp. 49-58

Lee W., Stolfo S., and Mok K., "A Data Mining Framework for Building Intrusion Detection Model," in Proceedings of IEEE Symposium on Security and Privacy , Oakland, pp. 120132, 1999.

Agrawal R., Imielinski T., and Swami A., "Mining Association Rules between Sets of Items in Large Databases," in Proceedings of the International Conference on Management of Data , USA, vol. 22, pp. 207216, 1993.

Abraham T., "IDDM: Intrusion Detection using Data Mining Techniques," available at: http://www.dsto.defence.gov.au/publications/234 5/DSTOGD0286.pdf, last visited 2008.

Amor N., Benferhat S., and Elouedi Z., "Naive Bayes vs Decision Trees in Intrusion Detection Systems," in Proceedings of the ACM Symposium on Applied Computing , USA, pp. 420424, 2004.

Sen J., "An AgentBased Intrusion Detection System for Local Area Networks," International Journal of Communication Networks and Information Security , vol. 2, no. 2, pp. 128140, 2010.

Chimphlee W., Abdulla A., Sap M., Chimphlee S., and Srinoy S., "A Rough Fuzzy Hybrid Algorithm for Computer Intrusion Detection," The International Arab Journal of Information Technology , vol. 4, no. 3, pp. 247254, 2007.

KDDCUP 1999 Data, available at: http://kdd.ics.uci.edu/databases/kddcup99/kddcu p99.html, last visited 2013.