ISSN 2320-9984 03 >
9 772320 998015

Volume 1, Issue 3, October 2013 ISSN: 2320-9984 (Online)

International Journal of Modern Engineering & Management Research

Website: www.ijmemr.org

Intrusion Detection System Using Semi-Supervised Machine Learning By DBSCAN

Virendra Singh Thakur
M. Tech. Scholar
Department of Computer Science & Engineering
Madhav Prodyogiki Mahavidyalaya
Bhopal (M.P.) [INDIA]
Email :veeru_jb@yahoo.com

Dr. Gireesh Kumar Dixit
Assistant Professor
Department of Computer Science & Engineering
Madhav Prodyogiki Mahavidyalaya
Bhopal (M.P.) [INDIA]
Email : gireeshdixit15@rediffmail.com

Abstract**—***A lot of research work has been done in the area of internet traffic classification by application type and several classifier is suggested. In this paper, we apply both technique i.e Supervised and unsupervised learning approach, known as sem- supervised classification based on DBSCAN clustering algorithm. It classify network flows by using only flow statistics. This methodology is based on machine learning principle, consists of two components : clustering and classification. The goal of clustering is to partitions the training data set in to disjoint groups. After making clusters classification is performed in which labeled data are used for assigning class label to the clusters. A NSL KDD data set are used for training and testing this approach. Which includes four kinds of attack and normal data. Experimental result shows that DBSCAN has better effectiveness and efficiency.*

Keywords**—***Clustering, Classification, Machine learning, DBSCAN, Traffic classification.*

1. INTRODUCTION

Network traffic classification is the process of identifying traffic flows and associating them to different categories of network application, and it represents an essential task in the whole chain of network management. The aim of network traffic classification is to find out what type of application are run by end users, and what is the share of the traffic generated by different applications in the total traffic mix. All activities related to network are linked to traffic. Network traffic is an important carrier to record and reflect the internet and end user activities; it is also an important composition of network behavior, through the analysis of network traffic statistics, we can master the network statistical behavior indirectly. Network traffic classification plays an important role in network activities such as network management, planning and network design,. It also includes the allocation, control and management of resources in TCP/IP networks. Network classification is also essential for bandwidth management, traffic shaping, intrusion detection and abnormality. The above activities need the capability of accurately classifying and identifying internet traffic.

The rest of this paper is organized is written as followed. Section 2 represents back ground and related work about internet traffic classification. Section 3 introduces DBSCAN algorithm. Section 4 and 5 presents our methodology data set and experimental results. Section 6 represents our conclusion.

2. BACK GROUND AND RELATED WORK

There are many approaches suggested to classify the internet traffic. The classical or traditional approach "Port based classification" approach[1] classify simply inspect the port number of packet and identify the application according to the IANA'S list of well known ports and register ports. This method is no longer effective when dealing with P2P applications, real video stream. So the port-based approach is currently combined with other approaches for traffic classification.

An alternate approach for well-known port number is pay load based analysis, where packet pay loads are searched for characteristics signatures of known application [9]. It also determine whether they contain the given characteristics strings.[6] presents a statistical machine learning algorithm to automatically extract application signatures from IP traffic payload. Although packet payload analysis has high accuracy but it having some demerits. (1) It fail to detect encrypted traffic (2) It required increased processing and storage capacity. (3) It useless if Payload is not available. (4) It is unable to identify the unknown application. (5) Signature must be obtain in advance, and it may be changed along with the evolvement of application.

Another promising approach to traffic classification is the use of machine learning. This approach is fundamentally different from traditional traffic classification approaches. It has many advantages: (1) It does not rely on port number. Further, we presume no prior knowledge about port- application mapping. (2) It does not require the inspection of traffic payload. Basically there are two steps in this technique Learning and Testing. The learning phase examines the provided data and construct the classification model. In testing phase the model that has been built in the training phase is used to classify new unseen instances. Machine learning technique [5] can be divided into categories of unsupervised and supervised. The supervised approach requires the training data to be labeled before the model is built. The goal of this method is how to improve the accuracy of classification. Their exist a number of supervised learning classification algorithm eg. C4.5, Decision tree, Naïve Bayes. While the unsupervised approach do not need to handle labeled traces, they just based on the inner similarity among all flows with in a training set to group several cluster. Their exist a number of unsupervised learning classification algorithm eg. K-means, Auto class, DBSAN.

A semi-supervised methodology is the combination of both supervised and unsupervised learning to improve the performance of classifier. While building the model, the supervised approach should classify the train data in advance, but the unsupervised not. The semi-supervised technique is based on machine learning principles, consists of two components a learner and a classifier. The goal of the learner is to discern a mapping between flows and traffic class from a training data set. Subsequently this learned mapping is used to obtain the classifier. There are many advantages in semi supervised approach: (1) Fast and accurate classifier is obtained by training with a small number of labeled flows mix with a large number of unlabeled flows. (2) This approach is robust and can handle previously unseen flows. Further more this approach allows iterative development of the classifier by allowing flexibility of adding unlabeled flows to enhance the classifier's performance.

In this paper we proposed semi-supervised technique using DBSCAN algorithm which classifies network flows by using only flow statistics which is analyzed and implemented. This technique is based on machine learning principle consists of two components clustering and classification. Clustering is used to partitions the training data set into disjoint group (cluster). After making cluster, classification is performed in which labeled data are used for assigning class labels to the clusters. Labeled data means the input set for which the class to which it belong is

known. Unlabeled data set is one for which class to which it belongs is unknown and is to be properly classified. This technique will enable to built a traffic classifier using flow statistic from both labeled and un labeled flow. Our method consist of two step clustering and classification. The details of this steps are as follows.

A. Clustering

We first employ a clustering algorithm to partition a training data set that consist of labeled flows combined with unlabeled flows. Clustering data is method by which the large set of data are grouped into clusters of small sets of similar data. We propose DBSCAN algorithm for clustering purpose.

B. Classification

After clustering of training data, we use the available labeled flows to obtain a mapping from the clusters to the different known classes the result of the learning is a set of cluster. Some mapped to the different flow types. This method referred to as semi-supervised learning. The input data for classification task is collection of numbers of records. Each records, also known as instance, is characterized by a tuple (x, y) where x is the attribute set and y is class attribute.

3. DBSCAN ALGORITHM

In this paper, we apply semi-supervised machine learning approach for internet traffic classification, which use DBSCAN (Density Based Spatial Clustering of Application with noise) algorithm to cluster train data for model building. This clustering algorithm has an advantage over partition-based algorithms because they are not limited to finding spherical shaped clusters but can find clusters of arbitrary shapes. The DBSCAN algorithm has three merits: (1) Minimal requirements of domain knowledge to determine the input parameters. (2) Discovery of clusters with arbitrary shapes. (3) Good efficiency on large data set. Experiment results show that DBSCAN has better effectiveness and efficiency.

There are two important objects : clusters and noise, for DBSCAN algorithm. All points in data set are divided into points of clusters and noise. The key idea of DBSCAN is that for each point of a cluster the neighborhood of a given radius has to contain at least a minimum number of points, i.e. the density in the neighborhood has to exceed some threshold. The shape of neighborhood is determined by the distance function for two points p and q, denoted by dist(p,q). Euclidean distance function is using for DBSCAN in this paper as formula

$$Dist(p,q) = \sqrt{\sum_{i=1}^{n} (P_i - q_i)^2}$$

Where n is the number of the features for point object p and q, p_i and q_i are the i^{th} feature of point object p and q.

The DBSCAN algorithm is based on the concepts of density reach ability and density-connectivity. These concepts depend on two input parameters: epsilon (eps) and minimum number of points (minpts). Epsilion is the distance around an object that defines its eps-neighborhood. For a given object q, when the number of objects with in the eps-neighborhood is at least minpts, then q is de-fined as a core object. All objects within its eps -neighborhood are said to directly density reachable from q. In addition, an object p is said to density reachable it is with in the eps-neighborhood of an object that is directly density-reachable or density reachable from q. further more, objects p and q are said to be density connected if an object o exists that both p and q are density reachable.

These notations of density – reach ability and density connectivity are used to define what the DBSCAN algorithm consider as a cluster. A cluster is defined ad the set of objects in a data set that are density- connected to particular core object. Any object that is not

part of a cluster is categorized as a noise. This is in contrast to K-means, and Auto Class, which assign every object to a cluster.

The DBSCAN algorithm works as follows. Initially, all objects in the data set are assumed to be unassigned. DBSCAN then chooses an arbitrary unassigned object p from the data set. If DBSCAN finds p is a core object, it finds all the density connected objects based on eps and minpts. It assigns all these objects to a new cluster. If DBSCAN finds p is not a core object, then p is considered to be noise and DBSCAN moves onto the next unassigned object. Once every object is assigned, the algorithm stops.

4. DATA SET, TOOLS AND SYSTEM ARCHETCHTURE

Before applying clustering, we need to follow few prerequisites. Real word data tend to be dirty, incomplete and inconsistent. Data preprocessing technique can improve the quality of the data, thereby helping to improve the accuracy and efficiency of the subsequent mining process.

Data set can also be in varying from, e.g. one attribute varies in range of 100 and other attribute varies in the range of 10000. So, a proper normalization of data set is done in which each attribute come in the range of 100 or whatever user selects. This normalization technique is known as Min-max normalization.

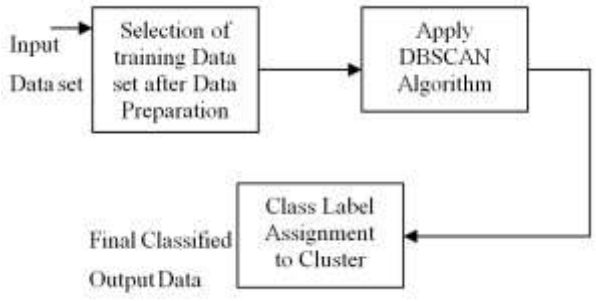

Figure 1. Architecture of Proposed System

The input data set is the real data which captured in the real network. It includes many kinds of attack data, also includes the normal data. The classification model is built is based on semi-supervised machine learning approach, thus both labeled and unlabeled data record are present. The output will basically would be the classification that will specify the class to which the data set is belong irrespective of the data input is labeled or unlabeled.

The figure 1. denotes the sequential execution of various phases the output of the first phase act as input to the second phase and so on. Initially the input is taken as the data set which acts as input to phase one in which both labeled and unlabeled data are used. It will partition a whole data space into small number of disjoint regions (cluster). Finally, it labels the cluster, for each cluster formed; perform the probabilistic assignment to find the mapping from cluster to labels. If the maximum priority belongs to same class, then all the labeled and unlabeled data samples with in the cluster are assigned the same class label. The result parameter after each phase can be viewed by the user. These results are displayed in the form of graphical tables. The entire classification task terminate once the class are assigned to all the data samples. This classification technique helps in classifying data and also making the system to learn how to classify a new coming data.

5. EXPERIMENTAL WORK

This section describe the data set form the basis of our work as well as the analysis tool used in the evaluation. The basic input to any system is the data. In the proposed system the input is the real world data set. Real data set are the set that a data analyst may encounter while dealing with real world application where the attribute are real valued. Furthermore, the data set may contain binary or numeric values. Since the system is deigned for semi-supervised classification thus we consider both labeled and unlabeled data for classification. The classifier is tested on KDD Cup 99 data set.

The aim of implementation or programming phase is to translate the design of the system produced during the design phase in

to code in a given programming language, so that the code is simple, easy to test and easy to understand and modify. For given design aim is to implement the design in best possible manner. The system will be implementing using MATLAB software, which is user friendly with graphical interface that can be easily implement in MATLAB as compared to other GUI development tools. Moreover MATLAB is high performance language for technical computing. It integrates computation, visualization, and programming in an easy to use environment where problems and solutions are expressed in familiar mathematical notation. Hardware requirements of the system are as follows:

- Processor : PENTIUM III or onwards
- RAM : 512 MB SDRAM
- Hard disk : 200 MB of disk space
- Operating System : windows, Mac, UNIX etc.

Experimenting with KDD CUP 99 data set

The KDD Cup 1999 Intrusion detection data is used in our experiments. This data was prepared by the 1998 DARPA Intrusion Detection Evaluation program by MIT Lincoln Labs. Lincoln labs acquired nine weeks of raw TCP dump data. The raw data was processed into connection records, which consist of about 5 million connection records. The data set for our experiments contained 10000 records. This data set is divided into training dataset which contained 8000 records and test dataset which contained 2000 records. Training dataset consists of 2400 labeled records and 5600 unlabeled records. As the data set has five different classes like normal data belongs to class1, probe belongs to class2, denial of service (DOS) belongs to class3, user to root (U2R) belongs to class4 and remote to local (R2L) belongs to class5. This dataset has been experimented on the designed classifier. The results obtained are displayed in next section, followed by discussion on the obtained results.

Following is the detailed description of the dataset:

- Number of Instances: 8000
- Number of Attributes: 41 plus the class attribute
- Attribute Information: duration, protocol_type,service, src_bytes, dst_bytes, urgent, and, wrong_fragment, flag, hot, num_failed_logins, logged_in,etc.

B) Performance Evaluation

The performance accuracy of the system is evaluated based on the generalization error. Generalization Error: generalization error can be obtained by testing the classifier using testing samples from the dataset. Testing samples are the remaining samples of the dataset after selecting samples for training. These samples are being tested for correct classification, thus find out that how much the system is generalized. The number of samples misclassified out of the total testing samples gives the generalization error. It is necessary to evaluate the performance of the system being designed. To do so generalization accuracy of the system is computed. The generalization accuracy is evaluated on the testing samples. Amongst all the testing samples it is determined how many samples are wrongly classified. Then accuracy percentage is given by

$$\text{Accuracy} = \frac{\text{Total Testing Sample- Misclassified Sample}*100}{\text{Total Testing Samples}}$$

Classifier gives 90.65 % accuracy respectively for KDD cup 99 dataset.

6. CONCLUSION

The aim of the project is to design and implement a semi supervised learning approach for network traffic classification and it has been achieved successfully. A semi supervised approach to design a Network Traffic Classifiers is implemented successfully.

Algorithm permits both labeled and unlabeled data to be used in training the network. While performing training and testing of the classifier for a dataset, it is observed that a test error rate depends on the number of clusters which is randomly used in training phase. It is observed that the range of classifiers accuracy lies between 70% to 96 % for various datasets.

REFERENCES :

[1] "Network Traffic Classification Using Semi Supervised Approach" Amita Shrivastav and Aruna Tiwari

[2] (Basic Book) H. Margaret, S. S. Dynham, "Data Mining Introductory and Advanced topics".

[3] J. Erman, A. Mahanti, M. Arlitt, I. Cohen, and C. Williamson," Traffic Classification Using Clustering Algorithms", University of Calgary, *SIGCOMM'06 Workshops* September 1115, 2006, Pisa, Italy. Copyright 2006 ACM 1595934170/06/0009.

[4] "Design and Implementation of Semi-Supervised Classification using Support Vector Machine" GSITS M.E. Thesis, J. Thomas,2008

[5] L. Yingqiu, Li Wei, L. Yunchun," Network Traffic Classification Using K-means Clustering", *School of Computer Science and Engineering, Beihang University, Beijing 100083, China,* 2007 IEEE

[6] J. Erman, A. Mahanti, M. Arlitt, I. Cohen, and C. Williamson, "Offline/ Online Traffic Classification Using Semi-Supervised Learning", Technical report, University of Calgary, 2007.

[7] J. Erman, A. Mahanti, M. Arlitt, I. Cohen, and C. Williamson, "Semi-Supervised Network Traffic Classification", *SIGMETRICS'07*, June 12.16, 2007, San Diego, California, USA. ACM 978-1-59593-639-4/07/0006.

[8] G. Munz, Sa Li, G. Carle,". Traffic Anomaly Detection Using K-Means Clustering ", Computer Networks and Internet, Wilhelm Schickard Institute for Computer Science, University of Tuebingen, Germany

[9] E. Spafford, D. Zamboni, "Data collection mechanisms for intrusion detection systems", Center for Education and Research in Information Assurance and Security, CERIAS Technical Report (2000)

[10] (Basic Book) M. Kamber, J. Han, "Data Mining Concepts and Techniques", (2nd ed.) [9] KDD Cup 99 Intrusion Detection Datasets. Available at: http:// kdd.ics.uci.edu/databases/kddcup99/ kddcup99.html.