JCSE International Journal of Computer Sciences and Engineering Open Access

Review Paper Volume-4, Issue-3 E-ISSN: 2347-2693

Hybrid Intrusion Detection System Using K-Means Algorithm

Darshan K. Dagly[1*], Rohan V. Gori[2], Rohan R. Kamath[3] and Deepak H. Sharma[4]

[1*,2,3,4] Department of Computer Engineering,
K. J. Somaiya College of Engineering, India

www.ijcseonline.org

Received: Feb/22/2016 Revised: Mar/01/2016 Accepted: Mar/14/2016 Published: Mar/31/ 2016

Abstract— Today in the age of computers and internet, identity theft, data theft, privacy and confidentiality infringement are some of the major issues faced by organizations as well as individuals. Network and System Security can be provided with the help of firewalls and Intrusion Detection Systems. An Intrusion Detection System (IDS) investigates all incoming and outgoing network traffic to identify malicious behavior that may pose a threat to the confidentiality, integrity or availability of a network or a system. IDS can be signature-detection (misuse) based or anomaly detection based. Misuse detection technique can be used to detect only known attacks whereas anomaly detection can be used to detect novel attacks (Unknown Attacks).This paper focuses on Hybrid Intrusion Detection System which combines both Misuse and Anomaly Detection modules. Various data mining techniques have been developed and implemented to be used with Intrusion Detection Systems. We use K-Means Clustering Algorithm to cluster and classify the incoming data into normal and anomalous connections. Clustering is an unsupervised learning technique for finding patterns in collection of unsupervised data. Prototype testing shows that K-Means algorithm can be successfully used to detect unknown attacks in real live data.

Keywords— K-Means, Intrusion Detection system, Data Mining, Clustering

I. INTRODUCTION

In today's world, computers and networks play a major role in almost every aspect of life. A deliberate attempt to try and break into a secure network or any type of attack which denies service to the network compromises the integrity, availability and confidentiality of a network. Thus security related to networks and computers is a major issue which needs to be tackled. In this research, we aim to build a host based hybrid intrusion detection system which will be capable of detecting attacks on a system. The proposed system will detect specific types of attacks or anomalous behavior and inform the user of such activity. The system maintains a log of all incoming traffic which is classified as either normal or abnormal. Java is used as programming language to develop this system. We make use of Jpcap and Winpcap in order to sniff or capture incoming packets. These incoming packets are then analyzed and pre-processed. The packets first go through misuse detection, where they are checked against a set of rules. If there is a match against any of the rules, the packet is deemed as malicious and the user is informed of a potential or evident attack. If the packet does not match against any of the rules, it is then forwarded to the anomaly detection module. In the anomaly module we use K-means clustering algorithm. The system is trained using KDD-99 data set. The incoming traffic is converted to KDD-99 format, and classification is done based of clusters formed during the training. If the packet is classified as an anomaly, the user is informed and a signature of this malicious packet is created and stored in the signature database.

II. SYSTEM DESCRIPTION

A. System Architecture

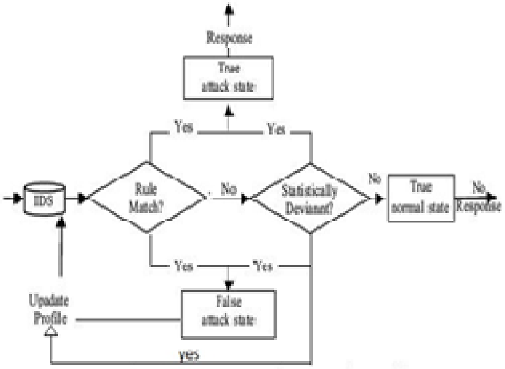

Figure 1: HIDS Architecture

The above diagram shows the system architecture for our proposed HIDS system. The system consists of a rule matching module which matches the signature of the incoming signal with the predefined database. If the rule is matched then the signal classified as an intrusion. If the rule is not matched then the threshold of the signal is compared with the predefined threshold by the anomaly module. If the signal doesn't cross the threshold then the signal is classified as a safe signal or else as an anomaly.

B. Hybrid Intrusion Detection System

In Hybrid Intrusion Detection System (HIDS), both the signature based and anomaly based system are integrated. In the training phase, a database of known attack signatures is constructed for the signature based system, and normal attack-free traffic is passed through the feature extractor of ADS to generate the episode rule database. When the traffic data passes through a Signature Matching Engine, known attacks are detected but novel attacks can bypass it. When the traffic cannot find any match with the normal connection rules in the database then an anomaly is detected. Hence both the systems are used in series to improve the effectiveness of the IDS. Once an anomaly is detected, a signature generation mechanism generates a unique signature which can be used to match the activity with a signature based system in the future.

C. Capturing Live Packets

1) JPCAP

Jpcap is a Java based API that enables java applications to capture network packets as well as send custom made packets over the network. Jpcap utilizes Winpcap/Libpcap libraries to provide the different functionalities as stated below:

- Live Capturing of raw packets from the network.
- Storing of captured packets to a file.
- Reading the captured packets from file.
- Filtering of packets as per the rules stated by the user and displaying them.
- Sending custom packets over the network.

2) WINPCAP

Winpcap is an open source industry standard tool for packet capture and network analysis for Windows. Winpcap allows the Network Interface Card (NIC) to run in promiscuous mode. A NIC in promiscuous mode captures traffic that is addressed to another machine and then passed it to the sniffer tool. Winpcap bypasses the protocol stack and allows application to transmit and captures network data packets.

Winpcap is mainly used by network sniffing tools, traffic loggers, traffic generators and network security tools to capture and read incoming and outgoing traffic.

D. Misuse Detection

Misuse detection is basically based on having a base of well-known attacks. Here patters or signatures are stored for these known attacks in the signature base. The incoming packets are checked against these signatures to find any type of intrusion. An advantage to this type of detection is that it is very fast as compared to anomaly detection. However signature detection demands that the system should already know about the attack, that is, it would not be able to detect previously unknown attacks. Hence this type of detection cannot be used to detect zero day attacks[4].

E. Anomaly Detection

Anomaly detection is based on the normal and abnormal behavior of a system. Here basically the intrusion detection system is trained so as to learn the “normal behavior” or normal traffic of a system. Any activity or traffic that deviates from the normal behavior is considered as an anomaly [7]. Anomaly detection methods presume that all malicious activities are necessarily anomalous. This means that if it could establish a "normal behavior profile" for a system, we would be able to detect any type of anomalous behavior and in turn detect intrusions. However, there arise two possibilities:

1. Anomalous activities that in reality are not malicious are classified as intrusive. These are known as false positives.
2. Intrusive activities that are not detected as anomalous. These are termed as false negatives.

In our research, we have used the KDD CUP ’99 dataset which has over 40000 connections, for training of the system. We use K-means clustering algorithm to form clusters and perform actual learning of the system[5].

F. Knowledge Discovery in Databases(KDD)

The KDD ‘99 intrusion detection datasets are based on the 1998 DARPA initiative to provide network security experts a benchmark to evaluate the different functionalities of their working model. The KDD 99 dataset uses a version of DARPA98 dataset. In KDD99 dataset, each instance represents attribute values of a network connection and classifies it into two classes i.e. normal or abnormal and labels it thereby in the dataset[2].

In KDD’99 dataset, here are a total of 41 attributes for each of the network connection that have either continuous or discrete values and are divided into four categories viz:

Basic Features: The Basic Features can be acquired from the header of the packet without analyzing the packet payload. It contains attributes like duration, protocol type, source bytes and destination bytes.

Content Features: Here the TCP packet payload is analyzed and dissected to extract attributes like no. of failed login attempts, etc.

Time-based Traffic Features: These attributes contain the properties of a connection which is captured over a time window of 2 seconds. Attributes in the dataset like number of connections for the same host for 2 sec time window can be considered as a time based feature.

Host-based Traffic Features: Unlike the time based window, host based traffic features have a historical connection based window which deliberates over the number of connections which in our case is 100. Thus, host based features can determine the attacks which last over an interval of 2 seconds.

The network attacks considered in the dataset can be classified into these four groups:

1) Denial of Service Attack (DOS): DOS is any type of attack where the hacker (attacker) prevents the legitimate user to access the system by making the computing resources and memory resources too busy to handle any legitimate requests.

2) User to Root Attack (U2R): U2R is an attack in which the attacker has gained normal access to the system on the network by sniffing passwords alongside social engineering and then exploits several weaknesses in the system to gain root access.

3) Remote to Local Attack (R2L): R2L is an attack in which the attacker tries to gain user access to the machine from a remote location by sending packets over the network and finding weaknesses the system or machine.

4) Probing Attack: Probing is an attack in which the attacker collects information about a network device or a system to determine vulnerabilities and weaknesses in the system which can be used for orchestrating future attacks on the system.

Hence we classify the dataset into 5 clusters: 1 cluster for normal data and remaining 4 clusters for the above 4 types of attacks

G. Preprocessing

KDD'99 Cup Dataset contains 41 attributes which can be classified into 4 groups' i.e. basic features, content features, time based features & host based features. We can extract basic features like flags, protocols directly from the incoming packets however we cannot get the attributes like failed login attempts, duration of connection, etc. directly. Thus pre-processing is required for converting packet level data to connection oriented data. After the implementation of pre-processing module, anomaly module can run on Real Time data on live network. However, all the attributes of the dataset cannot be extracted from the incoming packets. The attributes that can be derived are:

- Duration
- Protocol Type
- Land
- Source Bytes
- Destination Bytes

This required that the original KDD '99 dataset to be reduced to these five features, which was done using java program. After the pre-processing module, this connection data is provided to the anomaly module for classification of the traffic[3][6][8].

H. K-Means Algorithm

K-Means Algorithm falls under the category of unsupervised learning, which divides the given data set into various clusters. It categorizes data in such a way that the members of the same clusters are very similar to each other and dissimilar to the members of the other clusters[1].

- The **k-means algorithm** clusters *n* objects based on their attributes into *k* distinct partitions, where *k* is less than *n*.
- It partitions N data points into K disjoint subsets S_j $\mathbf{S} = \{S_1, S_2, \ldots, S_k\}$ such that data points of a particular subset minimize the sum of squares criteria.

$$\underset{\mathbf{S}}{\arg\min} \sum_{i=1}^{k} \sum_{\mathbf{x} \in S_i} \left\| \mathbf{x} - \boldsymbol{\mu}_i \right\|^2$$

Where $\boldsymbol{\mu}_i$ is the geometric centroid of points in S_i.

- In simple words, k-means clustering algorithm groups the objects into K number of disjoint groups based on their attributes.
- The grouping is done in such a way so as to minimize the sum of squares of distances between data points and the corresponding clusters' geometric mean

Steps of Algorithm:

1) Initialization: Decide the number of clusters to be made (k) and then randomly choose k instances from the dataset X and make them the cluster centers.

2) Assignment: Assign each instance from the dataset to its closest cluster center i.e. if $d_{i,j}(x_i,k_j) < d_{i,p}(x_i,k_p)$ then assign the instance to the cluster with the cluster center k_i.
3) Updating: The clusters' centroids are recalculated based on the elements in the respective clusters.
4) Repeat steps 2 & 3 until convergence is achieved, i.e. there is no movement of objects between the clusters.

Distance measure will determine how the *similarity* of two elements is calculated. They include:

1) Euclidean distance:

$$d_{xy} = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

2) Manhattan Distance:

$$D = \sum_{i=1}^{n} |x_i - y_i|$$

III. CONCLUSION

The paper proposes a hybrid intrusion detection system which uses K-means clustering algorithm. Here first the signature module was developed. In this module we checked each live packet against a base of well-known signatures. If there was a match it was deemed as an attack. Next was the anomaly module. The KDD cup 99 data set was used to train the system. The data set was divided into 5 clusters to perform unsupervised learning and based on this learning the system was able to classify incoming traffic as normal or abnormal. Heavy preprocessing of live packets was required for extracting the KDD attributes from live connections. The current system works on live network without becoming a bottleneck to the network bandwidth. A lot of work still has to be done in order to improve the efficiency of the system.

IV. ACKNOWLEDGEMENT

We sincerely thank Prof. Deepak Sharma, Associate Professor, Department of Computer Engineering, K. J. Somaiya College of Engineering for his constant support and guidance.

V. REFERENCES

[1] M. Jianliang, S. Haikun and B. Ling, "The Application on Intrusion Detection Based on K-means Cluster Algorithm," Information Technology and Applications, 2009. IFITA '09. International Forum on, Chengdu, 2009, pp. 150-152. Doi: 10.1109/IFITA.2009.34

[2] Ms. Urvashi Modi, Prof. Anurag Jain. A survey of IDS classification using KDD CUP 99 dataset in WEKA, International Journal of Scientific & Engineering Research, Volume 6, Issue 11, November-2015

[3] L.Dhanabal, Dr. S.P. Shantharajah. A Study on NSL-KDD Dataset for Intrusion Detection System Based on Classification Algorithms. International Journal of Advanced Research in Computer and Communication Engineering Vol. 4, Issue 6, June 2015

[4] N. T. Tran, S. Tomiyama, S. Kittitornkun and Tran Huy Vu, "TCP reassembly for signature-based Network Intrusion Detection systems," EEE, Computer, Telecommunications and Information Technology (ECTI-CON), 2012 9th International Conference on, Phetchaburi, 2012, pp. 1-4. doi: 10.1109/ECTICon.2012.6254336.

[5] Monowar Hussain Bhuyan, D K Bhattacharyya and J K Kalita. Survey on Incremental Approaches for Network Anomaly Detection. International Journal of Communication Networks and Information Security (IJCNIS) Vol. 3, No. 3, December 2011

[6] Sachin Baghel, Prof. Anurag Jain, Dr. J. L. Rana. A Review of Various Intrusion Detection Techniques on KDD Cup99 Dataset. International Journal of Emerging Technology and Advanced Engineering Volume 5, Issue 8, August 2015

[7] Nguyen Ha Duong, Hoang Dang Hai. A Model for Network TrafficAnomaly Detection. ICACT Transactions on Advanced Communications Technology (TACT) Vol. 4, Issue 4, July 2015.

[8] H. Günes Kayacık, A. Nur Zincir-Heywood, Malcolm I. Heywood. Selecting Features for Intrusion Detection: A Feature Relevance Analysis on KDD 99 Intrusion Detection Datasets. DOI: 17.01.16
https://web.cs.dal.ca/~zincir/bildiri/pst05-gnm.pdf