Asian Journal of Information Technology 15 (18): 3605-3610, 2016
ISSN: 1682-3915

Comparative Study of Multiple Intrusion Detection Systems

[1]S. Saravanan and [2]M. Ramakrishnan
[1]RMK College of Engineering and Technology, Chennai, India
[2]School of Information Technology, Madurai Kamaraj University, Madurai, India

Abstract: Network security is of paramount importance in the present data communication environment. Network security is a critical problem because a single attack can inflict catastrophic damages to the computers and network systems. The various hackers and intruders can create multiple successful attempts to cause the crash of the networks and web services by unauthorized intrusion. New threats and associated solutions to detect and prevent these threats are emerging together with the secured system evolution. The best solution to solve this issue is Intrusion Detection Systems (IDS). The important function of Intrusion Detection System (IDS) is to secure the resources from threats. In this study, we have presented a brief study about characteristics of ad hoc network, how they are problematic in ad hoc network security, attacks in ad hoc network and a description of some existing intrusion detection system. We have also justified why multiple intrusion detection is much better for ad hoc network with comparative study of existing intrusion detections in ad hoc network. This research proposed a new approach called Multiple Intrusion detection systems where the anomaly dataset is measured by the Neighborhood Outlier Factor (NOF). Here, trained model consists of big datasets with distributed storage environment for improving the performance of the proposed Intrusion Detection system. The experimental results proved that the proposed system identifies the anomalies very effectively than any other approaches. The experimental results proved that the proposed system identifies the anomalies very effectively than any other approaches.

Key words: Ad hoc network, multiple intrusion detection, neighborhood outlier factor, anomalies, approaches

INTRODUCTION

Network security has recently received an enormous attention due to the mounting security concerns in today's networks. A wide variety of algorithms have been proposed for detecting and to combat with these security threats. Among all these proposals, signature based Network Intrusion Detection Systems (NIDS) have been a commercial success and have seen a widespread adoption. A NIDS aims at detecting possible intrusions such as a malicious activity, computer attack and/or computer misuse, spread of a virus, etc and alerting the proper individuals upon detection. A NIDS monitors and analyzes the data packets that travel over a network looking for such suspicious activities (Spafford, 2008). A large NIDS server can be set up on the links of a backbone network to monitor all traffic or smaller systems can be set up to monitor traffic directed to a particular server, switch, gateway or router (Gu *et al*., 2007). Another class of NIDS can be setup at a centralized server which will scan the system files, looking for unauthorized activity and to maintain data integrity.

There are basically two primary approaches to NIDS implementation: signature based and anomaly detection based. The first approach has become a commercial success. A signature based NIDS maintains a collection of signatures, each of which characterizes the profile of a known security threat (e.g., a virus or a DoS attack). These signatures are used to parse the data streams of various flows traversing through the network link; when a flow matches a signature, appropriate action is taken (e.g., block the flow or rate limit it). Traditionally, security signatures have been specified as a string signature, port signature and header condition signature.

Anomaly based NIDS monitors network traffic and compares it against an established baseline of normal traffic profile. The baseline characterizes what is "normal" for the network such as the normal bandwidth usage, the common protocols used, correct combinations of ports numbers and devices and alerts the administrator or user anomalous traffic is detected which is significantly different from the baseline. It is highly subjective to decide what can be considered normal and what an anomaly but a widely accepted rule of thumb is that any

Corresponding Author: S. Saravanan, RMK College of Engineering and Technology,Chennai, India

incident which occurs on a frequency greater than two standard deviations from the statistical norm should be considered suspicious. An example of such behavior would be if a normal user logs on and off of a machine 20 times a day instead of the normal course of 1 or 2 times. At another level, a NIDS can also investigate the user patterns such as profiling the programs that are often executed, etc. If a user in the administrative department suddenly starts to execute programs from the engineering division or begins to compile a code, then the system can promptly alert the administrators.

Clearly, such anomaly based intrusion detection may lead to a high rate of false detection which is called as false positives (Gaikwad *et al*., 2012). It is generally considered difficult to keep low false positives in any system that sets aggressive policies for detecting anomalies. For example, it may be difficult to distinguish flash crowd from a Distributed Denial of Service attack (DDoS), thus a system may raise false alarm during a flash crowd event assuming that it is a DDoS attack. Similarly network reconfigurations and transient failures might abruptly change the traffic profile falsely raising the alarm. The second challenge concerns with the assumption made by these systems that attacks are always anomalous which may not necessarily be true. An intelligent attacker may develop intrusion techniques which will cause minimal disruption in the underlying traffic, thus may go undetected (Rafsanjani *et al*., 2008).

The final challenge in designing these intrusion detection systems concerns with the availability of dataset that is representative of normal traffic (Rocke and DeMara, 2006). To be realistic, the assumption that there exists attack-free data for training a detector outside of simulated data is not a realistic assumption. Typical network traffic contains a large number of scans, denial-of-service attacks and backscatter and worm activity. If not careful, this activity will become part of the normal state for an anomaly detector.

MATERIALS AND METHODS

Intrusion detection system: An intrusion is any unwanted activity either in the form of passive attacks or active attacks which are used by the attackers in order to create undesired situation and harmful consequences for the user's confidentiality, network integrity or network resources availability. In simple words, any set of actions that try to compromise the data integrity, user's confidentiality or service availability can be termed as intrusion while a system that attempts to detect such malicious actions of network or compromised nodes is known as IDS (Eskin *et al*., 2002). However, the security level of wireless networks can be enhanced up to certain limit by implementing IDS. The primary functions of IDS are to monitor users' activities, network behavior and different layers. A single perfect defense is neither feasible nor possible in wireless networks as there always exist some architectural weaknesses, software bugs or design flaws which may be compromised by the intruders. The best practice to secure the wireless networks is to implement multi lines of security mechanisms that is why, IDS is more critical in wireless networks which is viewed as a passive defense as it is not intended to prevent attacks, instead it alert network administrator about possible attacks well in time to stop or reduce the impact of the attack. The accuracy of intrusion detection is generally measured in terms of false positives (false alarms) and false negatives (attacks not detected) where an ideal IDSs attempt to minimize both these approaches (Kartit *et al*., 2012).

There are several different methods of approaches to actually detecting intrusions (Jaiganesh *et al*., 2013). These include statistical anomaly detection, rule-based anomaly detection and rule-based penetration identification. Statistical anomaly detection uses statistics formed from audit logs to detect anomalies from the behavior of normal user. Most statistical anomaly systems rely on "learning" about past behavior of users. Analysis of audit logs over time determine what behavior is normal for users. Any deviations generate alerts. Tests for determining normal behavior include mean and standard deviation. This test examines data from logs to see if they fall into the range of average behavior and how much the data points vary from one another. The multivariate test looks at correlation between two or more variables such as login frequency and time between sessions. If these two variables taken together exceed what is normal, then an alert will be generated. The Markov process examines transition probabilities between certain states. For example, it can look at the transitions between commands to see if they fit normal usage. The time series test determines whether something happens too quickly or too slowly. Finally, the operational test suspects intrusion if the number of occurrences of an event exceeds a predetermined limit. These tests can be used together to determine deviations because each test measures different aspects of a single event.

Another approach is rule-based anomaly detection. In this type of detection, the system analyzes data from audit logs and automatically develops a set of rules to describe normal behavior. While statistical anomaly detection inputs data into statistical tests to see

whether this data falls into previously learned statistics, rule-based anomaly detection relies on the rules generated from previous statistics. Hence, the data about each new event is tested against the rules to see whether it is normal. Because rules are generated from statistics, a large database of rules is needed for rule-based anomaly systems to work well. The number of rules could reach 10000 or even 1 million. Nevertheless, the rule based anomaly approach is as effective and strong as the statistical anomaly approach (Puttini *et al*., 2003). Despite the large volume of rules. Intrusion Detection Expert System (IDES) is an example of a statistical anomaly system and Wisdom and Sense (W&S) is an example of a system with rule-based anomaly detection.

Components and classification of IDS

Components of IDS: Broadly speaking, IDS has two main components (Eskin *et al*., 2002), i.e., the features and the modeling algorithm. Some Features may include attributes or measures which are mostly concern with the functionalities the IDS would provide. Algorithm is the core component and the efficiency and accuracy of detecting and responding intrusion is totally dependent on the underlying algorithm. IDS may have many components depending on the nature and characteristics of the network and possible intrusions. Most of the IDS have some common components as follows:

- Monitoring Component which is used for local events monitoring as well as neighbors monitoring
- Intrusion database which contains the records of recent misbehaviors and reputation value for the neighbors
- Response component which is used to respond in case of intrusion is detected. The response may be used to raise an alarm to alert the administrator or to broadcast the information to its neighbor nodes about the misbehaving node

Classification of IDS: Two distinct types of intrusion detection systems exist. Pattern-based intrusion detection system has the capability to identify all the known intrusions, while anomaly-based intrusion detection mechanisms have the intelligence in identifying and responding to new intrusions which are not known. IDS are further classified as Stand-alone IDS, Distributed and Cooperative IDS and Hierarchical IDS (Gu *et al*., 2014).

Stand-alone IDS operates on each node independently to determine intrusions by monitoring the internal events which are recorded in the system logs. In distributed and cooperative IDS, every node will participate in intrusion detection and response while in hierarchical IDS, the cluster-heads monitor all of its child nodes and respond in case of detection of the intrusion (Gu *et al*., 2014). Classification of Intrusion Detection Systems (IDS) is as follows.

Classification is one of the best known solution approaches. National Institute of Standards and Technology (NIST) organization provides guidance document on Intrusion Detection Systems. Intrusion Detection System briefly classified into three different categories:

- Host-based IDS
- Network-based IDS
- Vulnerability-assessment IDS

There are two basic models used to analyze the events and discover attacks:

- Misuse detection model intrusion detection system detect intrusions by looking for similar activities such as vulnerabilities or known intrusion signatures
- Anomaly detection model IDS detect intrusions by searching abnormal network traffic

The misuse detection model is commonly referred as IDS commercial tool (Puttini *et al*., 2003); vendors must update intrusion signatures. Anomaly detection based IDS model generally have the capability to detect attack symptoms without specifying the various attack models but these models are very sensitive to false alarms. In the present study we have utilized the proposed IDS approach's based on the anomaly detection model.

General architecture and diagram of the proposed system: The proposed Multiple intrusion detection system is composed of multiple local IDSs agents. Each IDS agent (Fig. 1) is responsible for detecting the possible intrusions locally. The collection of all the independent IDS agents forms the IDS system for the mobile wireless ad hoc network. Each local IDS agent is composed of the following components:

- Data collector: Responsible for selecting local audit data and activity logs
- Detection engine: Responsible for detecting local anomalies using local audit data. The local anomaly detection is performed using the eSOM classification algorithm. The procedure that is followed in the local detection engine is the one described below

- Select labeled audit data and perform the appropriate transformations
- Compute the classifier using training data and the eSOM algorithm
- Apply the classifier to test local audit data in order to classify it as normal or abnormal

Figure 2 summarizes the major steps in our system based on security policy at three levels. We need to collect event logs from three different levels, then we can group, filter alerts chronic and finally, we can correlate our data to reduce their volumes for ease of analysis and optimization of processing time in search of some intrusions. In the case of an intrusion of level 2 or 3, the administrator can group data together to know exactly how events unfolded. This method is called "event reconstruction" and it is really useful for administrators because they can:

- Have a better understanding of the needs of their networks
- Identify weaknesses in the system and improve safety policies
- Preventing abuse of these weaknesses by malicious internal and external
- Update the knowledge base of level 1
- Solving the problem of false positive and negative to reduce their numbers, thus reducing the number of alerts and speed up the processing thereafter
- Improve, continuously, the performance of our system

As shown in the proposed architecture in Fig. 2, where the traffic packet arrives, it passes through the first level where the IDS is installed. If a packet is intrusive and his script is included in the IDS database, the packet will

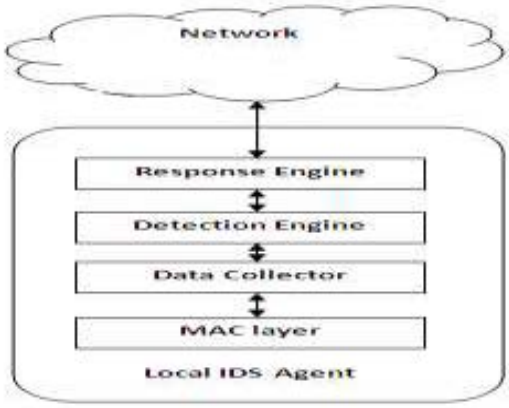

Fig. 1: IDS with multiple local IDS

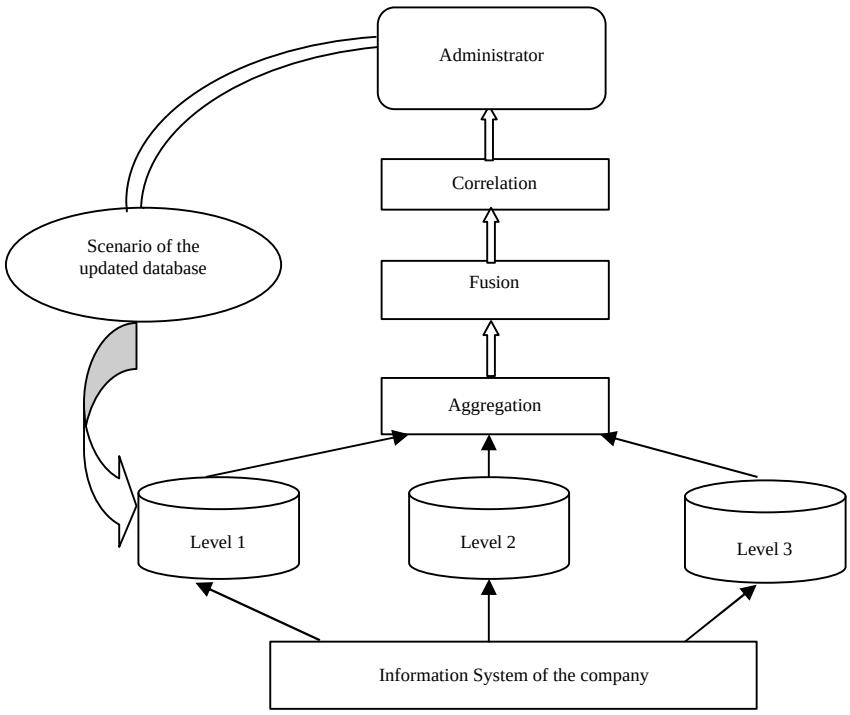

Fig. 2: Proposed multiple IDS architecture

be rejected, if it is not the case, it passes through the second level where we check the type of service performed or requested by the user behind the machine, if it is allowed to use the requested service or not. If it is not allowed to access the services requested and/or resources, the application will be rejected and the network administrator will be notified by an alert to start the diagnostics, if so, the packet passes through the third level. In this level, we check if the user is present in the company or not. If he is present, hence the user will have full access to services and/or resources required. If he is absent, it will not be allowed to access it remotely, the packet will be rejected and the network administrator will be notified by an alert to run diagnostics. We present the analysis of the intrusive packet provided to the network administrator in order to determine the origin of the attack with the reconstruction of events to highlight what exactly happened and implement measures to against this new type of attack and subsequently update the IDS's database of level 1.

RESULTS AND DISCUSSION

Comparison of proposed approach and existing approach (execution time vs dataset size): Figure 3 shows the overview of various execution times with various size of dataset. The Proposed Intrusion Detection System takes less execution time at every level rather than other existing machine learning approaches. The cause is less trained datasets thus the distance computation is easy between the trained and testing dataset, respectively.

Comparison of proposed approach and existing approach (anomaly detection rate vs dataset size): Figure 4 shows the anomaly detection rate in the computer network. The proposed Intrusion Detection System identifies almost all type of attacks such as Probe, DoS, U2R and R2L. The anomaly detection rate depends on the outlier values of the testing data. If the outlier value increase then the dataset assumed acts as intrusion dataset.

Experiment results: Experimental results are shown in Table 1 with Identification Rules (IR) where DR = Detection Rate, PDR = Partial Detection Rate, MR = Misclassification rate, FAR = False Alarm Rate.

We can see that in the proposed new scheme, the overall detection rate, i.e., the sum of the first three columns is always the same as the detection rate of anomaly detection model alone (Table 1). This is not surprising because the rules are used to (further) identify the attack type only after an anomaly is detected. The overall detection rate is not changed because no additional attacks will be detected. We can see that most of the well-known attacks have been detected except very few.

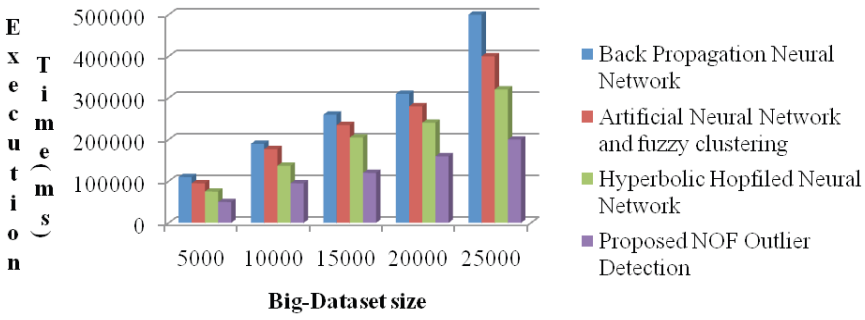

Fig. 3: Big-dataset size vs. execution time

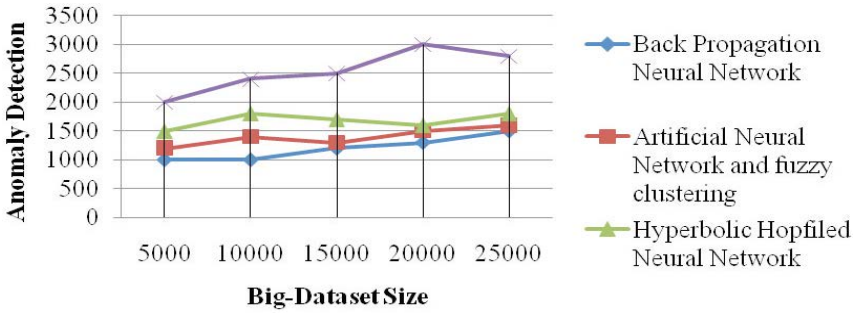

Fig. 4: Big-dataset size vs. anomaly detection

CONCLUSION

In this study, we have presented the details of a new approach called Multiple Intrusion Detection approach to detect the intrusion in the computer network. Our training model consists of big datasets with distributed environment that improves the performance of the proposed Multiple Intrusion detection system. The proposed system is also been tested with the KDD datasets that are received from real world. The existing approaches are capable of detecting the intrusion in the computer network with huge execution time and storage to predict the dataset when compared to the proposed IDS system which takes less execution time and storage to test the dataset .Here in our study, the performance of proposed IDS is much better than that of other existing approaches and is significantly capable of detecting almost all anomaly data in the computer network. In future, the proposed work can be possibly used for various distance computation function between the trained model and testing data. Our research work can be considered to improve the efficiency of IDS in a better manner which is shown in the experiment results.

REFERENCES

Eskin, E., A. Arnold, M. Prerau, L. Portnoy and S. Stolfo, 2002. A Geometric Framework for Unsupervised Anomaly Detection. In: Applications of Data Mining in Computer Security. Daniel, B. and S. Jajodia (Eds.). Springer US, Berlin, Germany, ISBN: 978-1-4613-5321-8, pp: 77-101.

Gaikwad, D.P., S. Jagtap, K. Thakare and V. Budhawant, 2012. Anomaly based intrusion detection system using artificial neural network and fuzzy clustering. Int. J. Eng. Res. Technol., Vol. 1,

Gu, G., P. Porras, V. Yegneswaran, M. Fong and W. Lee, 2007. BotHunter: Detecting malware infection through IDS-driven dialog correlation. Proceedings of the 16th USENIX Security Symposium, August 6-10, 2007, Boston, MA., USA -.

Gu, L., D. Zeng, P. Li and S. Guo, 2014. Cost minimization for big data processing in geo-distributed data centers. IEEE. Trans. Emerg. Topics Comput., 2: 314-323.

Jaiganesh, V., P. Sumathi and S. Mangayarkarasi, 2013. An analysis of intrusion detection system using back propagation neural network. Proceedings of the 2013 International Conference on Information Communication and Embedded Systems (ICICES), February 21-22, 2013, IEEE, Chennai, India, ISBN: 978-1-4673-5786-9, pp: 232-236.

Kartit, A., A. Saidi, F. Bezzazi, E.M. Marraki and A. Radi, 2012. A new approach to intrusion detection system. J. Theor. Appl. Inf. Technol., 36: 284-290.

Puttini, R.S., J.M. Percher, L. Me, O. Camp and D.J.R. Sousa *et al*., 2003. A Modular Architecture for Distributed IDS in MANET. In: Computational Science and its Applications. Vipin, K., L.G. Marina, C.J.T. Kenneth and P.L. Ecuyer (Eds.). Springer Berlin Heidelberg, Berlin, Germany, ISBN: 978-3-540-40156-8, pp: 91-113.

Rafsanjani, M.K., A. Movaghar and F. Koroupi, 2008. Investigating intrusion detection systems in MANET and comparing IDSs for detecting misbehaving nodes. World Acad. Sci. Eng. Technol., 20: 351-355.

Rocke, A.J. and R.F. DeMara, 2006. CONFIDANT: Collaborative object notification framework for insider defense using autonomous network transactions. Auton. Agents Multi Agent Syst., 12: 93-114.

Spafford, E., 2008. James P. Anderson: An information security pioneer. IEEE. Secur. Privacy, 6: 9-9.