International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2017 IJSRCSEIT | Volume 2 | Issue 5 | ISSN : 2456-3307

Traffic Flooding Attack Detection Using SNMP MIB Variables and Decision Tree Classifier

J. Princy Juliet M.Sc (CS)*[1], Mrs. R.Carolene Praveena M.Sc, M.Phil[2]

*[1]MPhil Research Scholar, Department of Information Technology, Sri Jayendra Saraswathy Maha Vidyalaya CAS, Coimbatore, Tamil Nadu, India

[2]Assistant Professor, Department of Information Technology, Sri Jayendra Saraswathy Maha Vidyalaya CAS, Coimbatore, Tamil Nadu, India

ABSTRACT

In emerging technology of Internet, security issues are becoming more challenging. The Internet has become an important source for information, entertainment, and a major means of communication at home and at work. With connectivity to the Internet, however, comes certain security threat. Unauthorized access, modifiers, denial of service, or complete control of machines by malicious users are all examples of security threats encountered on the Internet. Therefore, there is need for an approach, which will efficiently detect the flooding attacks in the network. The proposed system deals with Simple network management protocol based detection system to detect TCP and UDP flooding attacks effectively.

Keywords: SNMP, MIB, Decision Tree, TCP flooding, UDP flooding.

I. INTRODUCTION

With the ever more rapid development of the Internet, the Internet is currently an infrastructure for all kinds of network service. For example, the Web and VoIP service became the most popular and general services for the Internet. Many other new services such as IPTV service are emerging in the Internet. The significant increase of our dependency on Internet-based services in everyday life has intensified the survivability of networks. Because of extensive public availability, the Internet has become the main target of malicious attacks. Both the systems connected to the Internet and the network devices comprising the Internet, can all be severely compromised by intrusions. Recently, network flooding attacks such as DoS/DDoS and Internet Worm have posed devastating threats to network services [1]. The recent security forum reported that the DoS/DDoS attack is the main threat to the entire Internet, and the majority of them (90–94%) are deployed by using TCP. As a result, rapid detection and fast response mechanisms are the major concern for secure and reliable network services [2].

Distributed denial of service (DDoS) TCP and UDP flooding attacks are DoS attacks in which attackers flood a victim machine with packets in order to exhaust its resources or consume bandwidth. As the attack may be distributed over multiple machines, it will be very hard to differentiate authentic users from attackers [3]. In fact, a DDoS flood attack is not only a widespread attack; it is the second most common cybercrime attack to cause financial losses according to the United States Federal Bureau of Investigation (FBI).

SNMP provides a universal method of exchanging data for purposes of monitoring systems that reside on a network [4]. The use of SNMP is most dominant in the modern industry. SNMP is basically a management protocol, installed on network devices and is used to watch the activities of the network. An SNMP network management system consists of managed devices, agents and network monitoring systems (NMS). NMS monitors and manages the devices and may usually exists within them. NMS usually requests for the data and managed devices respond with the collected data from an agent. There are 66 MIB variables and proper selection of MIB variables plays a vital role in the detection of flooding attacks.

The audit data collected through the Management Information Base of Simple Network Management Protocol (SNMP-MIB) becomes an important entity since it provides information about the change in the network parameters directly [5]. For example, SNMP MIB-II, an IETF standard MIB supported by all the SNMP agents, provides a large number of traffic performance information on different layers and protocols such as IP, TCP, UDP, ICMP, etc. SNMP agents have already been implemented in most current network elements, thus, MIB statistical data available are easily collected for security analysis. This can be extended to collect additional data pertinent to network activities and is independent of the operating systems. By enlisting the MIB data, IDS promises a lower processing overhead for analysis, and high flexibility of deployment.

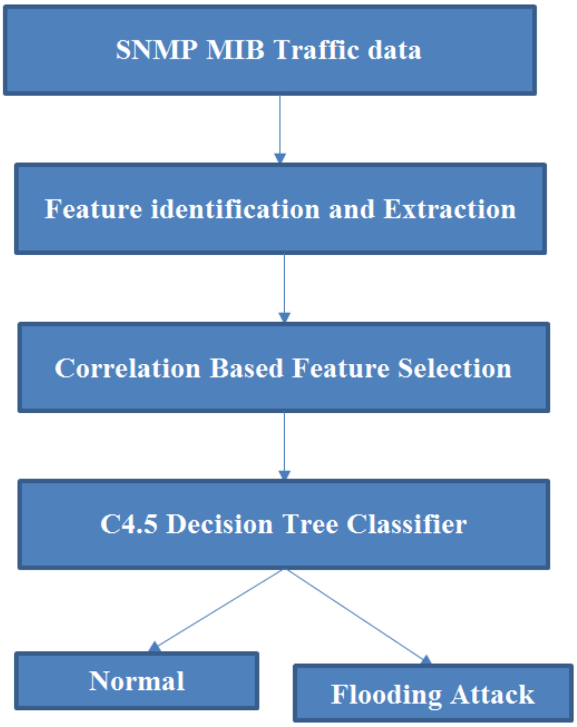
II. RELATED WORKS

Some studies used SNMP MIB data for intrusion detection [6]. A system is developed with a name MAID which uses SNMP MIB-II data for anomaly detection. They periodically collected 27 MIB variables from 4 MIB-II groups (Interface, IP, TCP, and UDP), and converted them into a probability density function (PDF) to calculate statistical similarity metrics which is the input data of the attack classifier. For the detection mechanism, they used a neural network classifier, a typical back propagation (BP) network, other than SVM. In [7] developed an anomaly detection system using periodic SNMP data collection which is derived from a PCA (Principle Component Analysis) based unsupervised anomaly detection.

In [8] used Simple Network Management Protocols Management Information Base (SNMP MIB) to detect intrusion. SNMP is a protocol used in TCP/IP network management and the idea to use it as a security monitoring tool is intriguing. SNMP logs are generated in network devices in any case and by using the already available logs do not add new requirements to the network infrastructure. By using SNMP MIB, some of the challenges in network intrusion detection can be avoided. There are no privacy concerns as user confidential information is not needed for the analysis. Also the data rates are low compared to network traffic amounts. SNMP MIB does not require any new hardware as the SNMP is widely supported. In their work they used 12 features from SNMP MIB in intrusion detection. Traffic on interfaces is estimated by analysing the correlation between IP group objects and interface group objects of SNMP MIB.

III. PROPOSED METHOD

An overview of the complete process of flooding attack detection is shown in the Figure 1, each of whose steps are explained in this section.


SNMP MIB Traffic data
Feature identification and Extraction
Correlation Based Feature Selection
C4.5 Decision Tree Classifier
Normal
Flooding Attack

SNMP MIB Traffic data:

The simple network Management Protocol provides a universal method for data exchange between nodes and elements in a network for the purpose of network maintenance and monitoring. SNMP includes a Network Management Service, which in turn runs slave agents in every system in the network that requires monitoring [9, 10]. The agent, which is a process called “snmpd” running in ever network element, collects network statistical data from that node at various layers and different protocols like TCP, UDP, ICMP, FTP, and HTTP. MIB stands for Management Information Base and is a group of objects that is used to store that information that is gathered by the “snmpd” process. The MIBs are nothing but a group of objects that are structured in a tree-like fashion. The roots of the trees are objects and the leaves are variables that hold that actual network parameter values.

Feature Identification and Extraction

For effectiveness of the data gathered from the MIB, it is necessary to ensure that the variables are queried out of the SNMP agent of the victim system, every time the agent itself updates the variables. This can be done by continuously querying out a variable and checking its value. The following are the features considered for detection of TCP and UDP flooding attacks.

tcpActiveOpens: The number of times that TCP connections have made a direct transition to the SYN-SENT state from the CLOSED state.

tcpPassiveOpens: The number of times TCP connections have made a direct transition to the SYN-RCVD state from the LISTEN state.

tcpAttemptFails: The number of times that TCP connections have made a direct transition to the CLOSED state from either the SYN-SENT state or the SYN-RCVD state, plus the number of times that TCP connections have made a direct transition to the LISTEN state from the SYN-RCVD state.

tcpEstabResets: The number of times that TCP connections have made a direct transition to the CLOSED state from either the ESTABLISHED state or the CLOSE-WAIT state.

tcpCurrEstab: The number of TCP connections for which the current state is either ESTABLISHED or CLOSE-WAIT.

tcpInSegs: The total number of segments received, including those received in error. This count includes segments received on currently established connections.

tcpOutSegs: The total number of segments sent, including those on current connections but excluding those containing only retransmitted octets.

udpInDatagrams : The total number of UDP datagrams delivered to UDP users.

udpNoPorts: The total number of received UDP datagrams for which there was no application at the destination port.

udpInErrors: The number of received UDP datagrams that could not be delivered for reasons other than the lack of an application at the destination port

udpOutDatagrams: The total number of UDP datagrams sent from this entity.

Correlation based Feature Selection (CFS)

Correlation based Feature Selection is an algorithm that couples this evaluation formula with an appropriate correlation measure and a heuristic search strategy. A correlation-based feature selection algorithm is proposed for selecting a subset of most informative features. We utilized the popular CFS algorithm whose performance is widely accepted and is implemented as a component of the WEKA [13]. Out of 11 features CFS selects 6 features for training and testing of decision tree classifier.

C4.5 Decision Tree Classifier

C4.5 Decision tree is an entropy based approach for generating a pruned or unpruned decision tree which examines the information gain ratio that results from choosing an attribute for splitting the data. The attribute with the highest information gain ratio is the one used to make the decision [11].

A decision tree is built top-down from a root node and involves partitioning the data into subsets that contain instances with similar values (homogenous). Decision tree algorithm uses entropy to calculate the homogeneity of a sample. If the sample is completely homogeneous the entropy is zero and if the sample is an equally divided it has entropy of one.

The estimation criterion in the decision tree algorithm is the selection of an attribute to test at each decision node in the tree. The goal is to select the attribute that is most useful for classifying examples. A good quantitative measure of the worth of an attribute is a statistical property called information gain that measures how well a given attribute separates the training examples according to their target classification [12]. This measure is used to select among the candidate attributes at each step while growing the tree. The information gain is based on the decrease in entropy after a dataset is split on an attribute. Constructing a decision tree is all about finding attribute that returns the highest information gain.

IV. EXPERIMENTAL RESULT

The experiment is done using Weka [13], which had been one of the standard tools in data mining and machine learning. It contains various classification and clustering algorithms like Naïve Bayes, J-48, Random Tree, Random Forest, etc. We also use accuracy, precision, F1-Measure as criteria to evaluate the classification performance which are described below.

Accuracy (ACC) is defined as the number of malicious over the total number of events. The more efficient the accuracy values, the higher the rate of correctly detected incidents.

$$ACC = \frac{TP + TN}{TP + FP + FN + TN}$$

Where, True Positive (TP) =Number of samples correctly predicted as malicious.
False Positive (FP) = Number of samples incorrectly predicted as malicious.
True Negative (TN) = Number of samples correctly predicted as normal.
False Negative (FN) = Number of samples incorrectly predicted as normal.

Precision = TP / (TP +FP)

$$Fmeasure = 2 * \frac{precision * recall}{precision + recall}$$

The experimental result is shown in the table 1.

Measures	Experimental results
Accuracy	95.8
Precision	0.93
F1-Measure	0.82

Table 1. Experimental results

Table 1 shows the performance of C4.5 decision tree classifier. We can see that the C4.5 decision tree classifier performs with accuracy of 95.8%. That means the classifier gained better performance in the detection of TCP and UDP flooding attacks using SNMP MIB variables.

V. CONCLUSION

This paper introduce TCP and UDP flooding attack detection system based on SNMP MIB data, which selects effective MIB variables and applied C4.5 decision tree classification algorithms based on chosen variables. Finally, the proposed system, models detection mechanism, is using the algorithm with the highest accuracy. The advantage of this system is its ability to learn. System's detection model will be optimized after receiving the new data. While the behavior of attack changes, the system will be adapted easily. Since SNMP manages the traffic flow in a network it is expected to achieve security in a using MIBs from SNMP with the aid of SNMP agents.

VI. REFERENCES

[1] Braga, R., Mota, E., & Passito, A. (2010, October). Lightweight DDoS flooding attack detection using NOX/OpenFlow. In *Local Computer Networks (LCN), 2010 IEEE 35th Conference on* (pp. 408-415). IEEE.

[2] Afanasyev, A., Mahadevan, P., Moiseenko, I., Uzun, E., & Zhang, L. (2013, May). Interest flooding attack and countermeasures in Named Data Networking. In *IFIP Networking Conference, 2013*(pp. 1-9). IEEE.

[3] Xiao, B., Chen, W., He, Y., & Sha, E. M. (2005, July). An active detecting method against SYN flooding attack. In *Parallel and distributed systems, 2005. proceedings. 11th international conference on* (Vol. 1, pp. 709-715). IEEE.

[4] Yu, J., Lee, H., Kim, M. S., & Park, D. (2008). Traffic flooding attack detection with SNMP MIB using SVM. *Computer Communications*, *31*(17), 4212-4219.

[5] Park, J. S., & Kim, M. S. (2008). Design and implementation of an SNMP-based traffic flooding attack detection system. *Challenges for next generation network operations and service management*, 380-389.

[6] Li, J., & Manikopoulos, C. (2003, June). Early statistical anomaly intrusion detection of DOS attacks using MIB traffic parameters. In *Information Assurance Workshop, 2003. IEEE Systems, Man and Cybernetics Society* (pp. 53-59). IEEE.

[7] Ramah, K. H., Ayari, H., & Kamoun, F. (2006). Traffic anomaly detection and characterization in the tunisian national university network. *Lecture notes in computer science*, *3976*, 136.

[8] Jun, J. H., Oh, H., & Kim, S. H. (2011, December). DDoS flooding attack detection through a step-by-step investigation. In *Networked Embedded Systems for Enterprise Applications (NESEA), 2011 IEEE 2nd International Conference on* (pp. 1-5). IEEE.

[9] Ahmed, E., Mohay, G., Tickle, A., & Bhatia, S. (2010). Use of ip addresses for high rate flooding attack detection. *Security and Privacy–Silver Linings in the Cloud*, 124-135.

[10] Streilein, W. W., Fried, D. J., & Cunningham, R. K. (2003, September). Detecting flood-based denial-of-service attacks with snmp/rmon. In *Workshop on Statistical and Machine*

Learning Techniques in Computer Intrusion Detection, Fairfax, Virginia, USA.

[11] Stein, G., Chen, B., Wu, A. S., & Hua, K. A. (2005, March). Decision tree classifier for network intrusion detection with GA-based feature selection. In *Proceedings of the 43rd annual Southeast regional conference-Volume 2* (pp. 136-141). ACM.

[12] Kruegel, C., & Toth, T. (2003). Using decision trees to improve signature-based intrusion detection. In *Recent Advances in Intrusion Detection* (pp. 173-191). Springer Berlin/Heidelberg.

[13] Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2009). The WEKA data mining software: an update. ACM SIGKDD explorations newsletter, 11(1), 10-18.