Middle-East Journal of Scientific Research 23 (3): 426-431, 2015
ISSN 1990-9233
© IDOSI Publications, 2015
DOI: 10.5829/idosi.mejsr.2015.23.03.22084

Anfis Based Multilayered Botnet Detection with Traffic Reduction Technique

[1]*M. Kempanna and* [2]*R. Jagadeesh Kannan*

[1]Department of CSE, Karpagam University, Coimbatore, India
[2]Department of CSE, V.I.T University-Chennai Campus, Chennai, India

Abstract: In recent years, Botnets have become the most serious threat for cyber-security. Botnet is a malicious software application which can be controlled by remote system from outside network through command and control channel. Most of the existing behavior based techniques could not able to detect and predict the botnet as they change their structure and pattern. For increasing of the efficiency of botnet detection, the multi-agent systems have been deployed. Intelligent systems with fuzzy and neural-fuzzy techniques improve the botnet presence degree in computer networks. In this paper, Adaptive Neuro Fuzzy Inference System (ANFIS) is used to train the system for future prediction. The multi-layered architecture is combined with ANFIS for the detection of a wide range of existing and new botnets. In addition, to improve the overall system performance, we develop a traffic reduction algorithm to reduce the amount of network traffic required to be inspected by the proposed system. Simulation results show that the proposed system achieves a high detection rate (98.75%) and a low false positive rate. The traffic reduction algorithm reduces an average traffic by 80%.

Key words: Botnet detection · ANFIS · Network security · Cyber-threat · Traffic reduction

INTRODUCTION

Botnets are one of the most dangerous species of network-based attacks today because they comprises of coordinated groups of hosts for both brute-force and subtle attacks. Botnet is a compromised network of computer called as bots which is remotely controlled by commands send by attacker with the intention of spreading spam’s, generating distributed denial of service, information hacking such as making malicious activities. In order to command a botnet, a command and control (C and C) channel is required through which bots receive commands and coordinate attacks and fraudulent activities. The C and C channel is formed by which individual bots communicate with botnet [1]. Botnets derive their power by scale, both in their cumulative bandwidth and in their reach [2]. Now a day’s botnets have become a popular technique for spreading internet crimes.

Bot detection systems are classified into two categories, i.e., signature based and behavior based techniques. Although a signature-based technique is accurate, it has the following drawbacks. First, signature based techniques are not helpful in identifying unknown bots. Second, a string signature is applicable only for a specific bot. When a bot has a variant behavior, string signatures cannot be applied. Hence, the false negative rates may increase in detection when new bots are developed. Third, an extremely large database containing all identified bots’ signatures may accidentally match benign software. Finally, it is possible for a bot to bypass signature based checks by using code obfuscation techniques. In contrast, behavior based techniques attempt to identify bot activities with the observed behavior of a specific bot. If behavior based techniques are tuned properly, they can be able to match the performance of signature based techniques in terms of detection rate. In addition, a behavior-based system does not need to maintain a signature database to detect bots. Such a system can be much more lightweight than a signature-based system.

The bot master commands the existing bots to compromise more user computers. There are many techniques to compromise a computer such as exploiting software vulnerabilities and social engineering. Once a targeted host computer is compromised, remote controllable software will be installed and launched so that the information about the compromised computer can be reported to the bot herder. In the attack phase, a bot herder sends commands to compromised hosts.

Corresponding Author: M. Kempanna, Department of CSE, Karpagam University, Coimbatore, India.

On receipt of the commands, each bot launches various tasks based on the instructions embedded in the commands. A bot herder is therefore able to ask bots to collect valuable information, report botnet status with regular network traffic. To improve botnet detection efficiency, traffic reduction technique is incorporated to avoid bot-irrelevant traffic. As a result, the ANFIS bot detection algorithm will be able to concentrate only on boot traffic.

Once the botnet detection has been completed, boot defenses mechanism has to be organized and developed within a flexible, structured architecture that is aimed at involving antibody software writers into a coordinated environment. The designed architecture has the following attributes such as Hierarchical and secure, Multilayered and open to being constantly enhanced with new antibody modules. The multilayer architecture is purposely designed such that the kernel is closed and secured and at the same time allows the modification, patching and enhancing of the system

Literature Review: Lividest *et al.* [3] developed a system to detect C and C traffic of IRC botnets. The system leverages machine learning techniques. The system contains two stages. In the first stage, it extracts several per-flow traffic attributes including flow duration, maximum initial congestion window and average byte counts per packet. In the second stage, it uses a Bayesian network classifier to make the classification balanced between false negative rates and false positive rates. However, the false positive rate is still high (15.04%). Sedan *et al*. [4] also tried to identify botnet C and C traffic and they observed that this type of traffic appears periodically. However, the observation was made for a simulated boot, not a real world bot. The validness of the results needs to be further examined by using real world bots. Choy *et al*. [5] proposed a botnet detection mechanism solely based on monitoring of DNS traffic. In an aggregated network trace, they found that DNS queries sent from bots can be easily grouped together by similarities of DNS requests and hence is able to be used to detect boot activities. Variants of bots can be detected as well. Go *et al*. [6] proposed the ''Boot-Sniffer.'' They identified boot hosts based on spatial–temporal correlation of collected network traces. They used bots collected from real world, reimplemented bots and self-produced bots to evaluate their solution. Although the results showed a high detection rate and low false positive rates, the number of evaluated real bots is very limited.

Fuzzy pattern based filtering algorithm is based on behavior based botnet detection for all type of botnet. With the help of fuzzy membership function, intended to identify malicious domain names and IP address. In this techniques define membership function for generated failed DNS queries, have similar DNS query interval, generate failed network connections an similar payload size for network connections. The main advantage of fuzzy membership function is that, it can easily altered and modified in order to improve the performance. Fuzzy pattern based filtering algorithm is help to detect human like behavior of botnet in [7].

Adaptive Neuro Fuzzy Inference System: The usage of artificial intelligence has been applied widely in most of the fields of computation studies. Main feature of this concept is the ability of self learning and self-predicting some desired outputs. The learning may be done with a supervised or an unsupervised way. Neural Network study and Fuzzy Logic are the basic areas of artificial intelligence concept. Adaptive Neuro-Fuzzy study combines these two methods and uses the advantages of both methods. ANFIS is an adaptive neuro fuzzy network which allows the usage of neural network topology along with fuzzy logic. It not only includes the characteristics of both methods, but also avoids disadvantages of both fuzzy logic and artificial neural network. ANFIS combines both neural network and fuzzy logic; it is capable of handling complex problems. Even if the targets are not given, ANFIS may reach the optimum result rapidly.

Neural networks are composed of various functional units operating in parallel. The architecture of the neural network has been derived from research on the biological functions of the cerebral cortex. Commonly neural networks are adjusted, or trained, based on a comparison of the output and the target, until the network output matches the target. Typically many such input target pairs are used in the supervised learning to train a network. Back propagation is an important learning algorithm for training multi layer feed forward networks. It was created by generalizing the Windrow-Hoff learning rule to multiple layer networks and nonlinear differentiable transfer functions. Input vectors and the corresponding output vectors are used to train a network until it can approximate a function, associates input vectors with specific output vectors in an appropriate way as defined by the user.

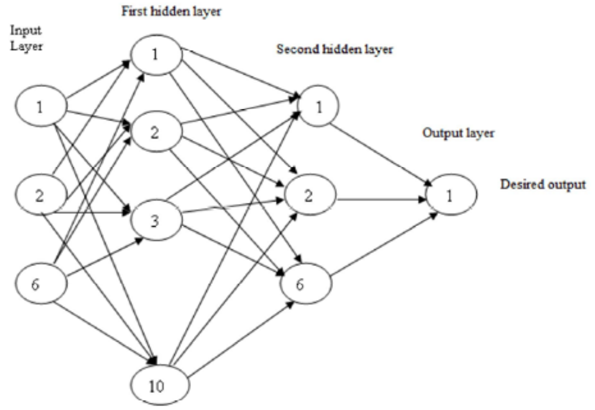

Fig. 1: Neuro fuzzy training network

Fuzzy inference is the process of formulating the mapping from a given input to an output using fuzzy logic. The mapping then provides a basis to its appropriate membership value. Two well known types are Madman-type and Surgeon-type. Both can be implemented in fuzzy logic toolbox. These two types differ in the way output's are determined. Madman-type inference expects the output membership functions to be fuzzy sets and requires defuzzification. A Sugeno fuzzy model has a crisp output, the overall output is obtained via weighted average, thus neglecting the time consuming process of defuzzification required in a Mamdani model [8].

The process of identifying a fuzzy model is generally divided into the identification of the premises and that of the consequences. Sugeno's method finds the best fuzzy model by repeating the followings: (i) the selection of the structures in the premises, (ii) the identification of the parameters in the premises, (iii) the selection of the structures is the consequences and (iv) the identification of the parameters in the consequences. The fuzzy training network is shown in Fig.1. This identifying process is time consuming and the characteristics of a fuzzy model depend heavily on the structures rather than on the parameters of the membership functions. The selection of the structures is first done once in the process. The selection of the structures of types I and II is done only in the premises since the structures of these types in the consequences are automatically determined with those in the premises [8]. After the structures are selected, the fuzzy neural network (FNN) identify the parameters of fuzzy models automatically. While implementing the ANN program, the network is trained using the back

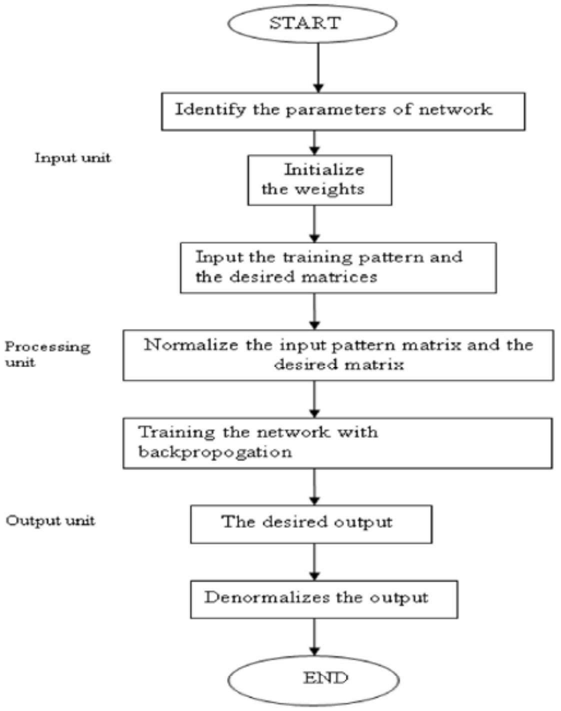
START
Identify the parameters of network
Input unit
Initialize the weights
Input the training pattern and the desired matrices
Processing unit
Normalize the input pattern matrix and the desired matrix
Training the network with backpropogation
Output unit
The desired output
Denormalizes the output
END

Fig. 2: Flow chart for ANFIS training algorithm

propagation algorithm (BPA). The training process of the network is shown in Fig. 2. The input unit includes: identifying the parameters of the network, identifying the weights entering the training patterns and the desired matrices and normalizing the input and the desired matrices. After the training of the network, the unknown patterns will be inserted in the program. The network will recognize the pattern and give the required classification [9].

A fuzzy set is an extension of a classical set. If X is the universe of discourse and its elements are denoted by x, then a fuzzy set A in X is defined as a set of ordered pairs. $A = \{x, \mu_A(x) \mid x ? X\}$, $\mu_A(x)$ is called the membership function (or MF) of x in A. The membership function maps each element of X to a membership value between 0 and 1. The basic membership functions are formed using straight lines. Of these, the simplest is the triangular membership function. Two membership functions are built on the Gaussian distribution curve: a simple Gaussian curve and a two-sided composite of two different Gaussian curves. Although the Gaussian membership functions and bell membership functions reach smoothness, they are unable to specify asymmetric membership functions, which are important in certain applications.

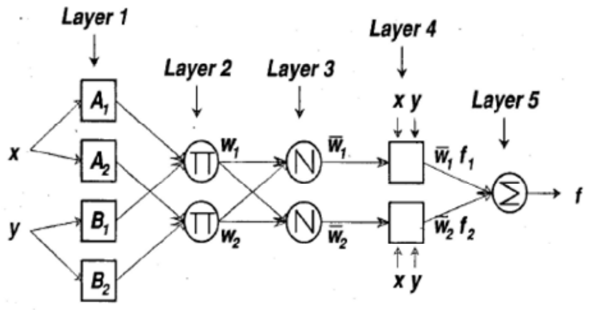

Fig. 3: ANFIS network

ANFIS was first proposed by Jang (1993). ANFIS serves as a basis for constructing a set of fuzzy 'if-then' rules with appropriate membership function to generate the stipulated input-output pairs. The membership functions are tuned to the input-output data. ANFIS is about taking an initial fuzzy inference (FIS) system and tuning it with a back propagation algorithm based on the collection of input-output data[10]. The basic structure of a fuzzy inference system consists of three conceptual components: a rule base, which contains a selection of fuzzy rules; a database, which defines the membership functions used in the fuzzy rules; and a reasoning mechanism, which performs the inference procedure upon the rules and the given facts to derive a reasonable output or conclusion. These intelligent systems combine knowledge, technique and methodologies from various sources. They possess human-like expertise within a specific domain – adapt themselves and learn to do better in changing environments. In ANFIS, neural networks recognize patterns and help adaptation to environments. Fuzzy inference systems incorporate human knowledge and perform interfacing and decision-making. Matlab's Fuzzy logic toolbox is used for the entire process of training and evaluation of FIS. Fig. 3 shows an ANFIS structure for two inputs.

Given a data set X = {x1,x2,…….x_n}, where the data point x_j ◎ R_p(j = 1,..., n), n is the number of data and p is the input dimension of a data point, traditional FCM [11] groups X into c clusters by minimizing the weighted sum of distances between the data and the cluster centers or prototypes defined as

$$Q = \sum_{i=1}^{c} \sum_{j=1}^{n} u_{ij}^{m} \left\| x_j - o_i \right\|^2 \qquad (1)$$

Here, ‖ ‖ is the Euclidean distance. u_{ij} is the membership of data x_j belonging to cluster i, which is represented by the prototype o_i. The constraint on u_{ij} is $\sum_{i=1}^{c} u_{ij} = 1$ and m is the fuzzification coefficient.

Proposed Technique: The goal is to use multilayered architecture to detect wellknown bots and botnets and to detect new classes and variants using ANFIS technique with traffic reduction. A general detection framework was developed in order to incorporate the detection methodology itself, as well as the data collection and storage modules and all the necessary management functions. Some performance tests were already carried out on the proposed system and the results obtained show that the system is stable and fast and the detection approach is efficient, since it provides high detection rates with low computational overhead. This paper presents a new approach, based on ANFIS networks, that could be able to detect zombie PCs based on the historical traffic profiles. The evaluation of the proposed methodology relies on traffic traces obtained in a controlled environment and composed by licit traffic measured from normal activity of network applications and malicious traffic. The results obtained show that the proposed methodology is able to achieve good identification results, being at the same time computationally efficient and easy to deploy in real network scenarios.

We propose a detection mechanism for bot C and C traffic by analyzing "suspicious" flows created after filtering out normal traffic from the traffic generated on a host. The filtering is based on a normal profile of the traffic generated by a user on a host. The profile is built dynamically by examining the behavioral pattern of flows to all destinations. A characterization of bot C and C behavior is also proposed, to derive a set of distinguishing attributes based on which detailed analysis is to be done. From the characterization, a few observations about the C and C traffic are made and an algorithm is proposed for detailed analysis and bot detection. Bot detection can be performed with or without regard to bot families. Users and network administrators are usually indifferent to information about bot families. Their primary concern is to protect their systems and networks from infections, regardless of details about bot family. On the other hand, security researchers are particularly interested in identifying bot families. The degree of prevalence of different botnets, their geographical distribution and the common characteristics of botnets are some of the plausible reasons for their heightened interest. Detection of bots indicates vulnerability of a host or network to botnet infection. This can be followed byremedial strategies aimed at recovering from the infection and preventive measures to avoid getting infected in future.

Active Detection: Active bot detection involves participating in the botnet operation. This typically involves impersonating as a component of the botnet. Active detection approaches involve Infiltration and C and C Server Hijack.

Infiltration: In infiltration, a defender-controlled machine masquerades as an actual bot and probes the C and C server or other peers in case of a P2P based botnet to gain details about other bots. The defender progressively gains information about other bots by repeatedly issuing crafted, bogus messages to declare itself as a new bot and subsequently obtain new peer-list.

C and C Server Hijack: Bots can be actively detected by C and C server hijack. Bots report to and receive commands from C and C server. Taking control of the C and C server will reveal all the bots that contact it. This can be achieved by exploiting botnet rallying mechanism. A defender can use this information to his/her advantage to hijack the server. This approach also leverages knowledge of botnet topology. Centralized botnet structures are more amenable to C and C server hijack. In decentralized botnets, the C and C server can be any peer and will, at most, reveal information about bots in its peer list. To gain further information, some other techniques need to be employed, such as active crawling of the P2P botnet. The seizure of C and C servers can be Physical or Virtual.

RESULTS AND DISCUSSION

To improve the overall system performance, we developed a traffic reduction technique to reduce the amount of network traffic. Table 1 compares the proposed BDS with the existing techniques in various aspects. Our method has a superior TPR and FPR with additional traffic reduction feature. The results based on normal traces also show a high traffic reduction rate of over 80% and low false positive rates (0–2%). Both results show that the proposed algorithm is not only efficient but also highly accurate. The results are compared with machine learning technique (MLT) and Neural network based approaches. The comparison graph for detection rate and FPR is shown in Fig. 4 and Fig. 5. In addition, the proposed algorithm can detect inactive botnets, which can be used to identify potential vulnerable hosts. Experimental results show that the proposed BDS technique has high detection rate of 98.75% for malicious IP addresses.

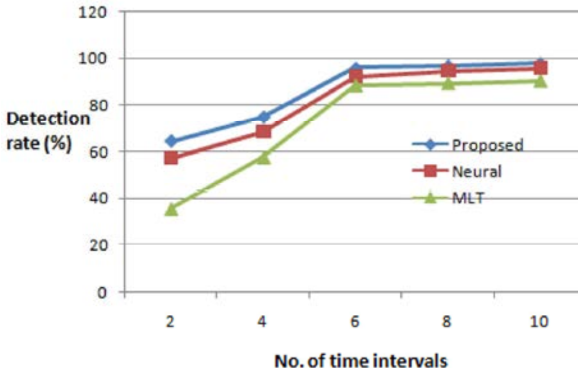

Fig. 4: Detection rate comparison

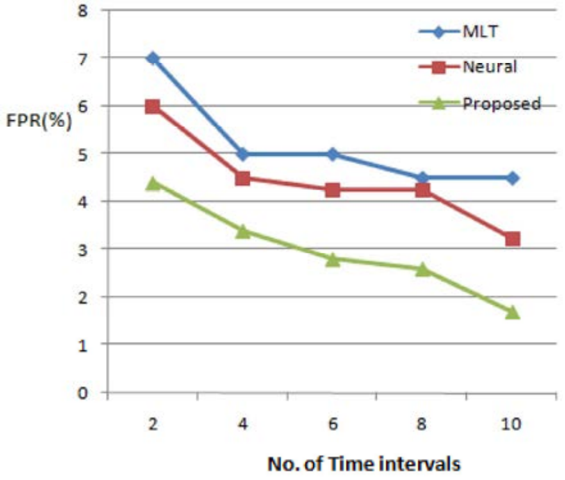

Fig. 5: Comparison of False positive rate (%)

Table 1: Comparison of Botnet detection methods

S. No	Attributes	MLT	Neural Network	Proposed ANFIS
1	Traffic reduction	N/A	N/A	Over 80%
2	Inactive Bots detection	No	No	Yea
3	True positive rate	94%	97.5%	98.5%
4	False positive rate	0-6%	0-2.5%	0-1.5%

CONCLUSION

This work proposed a new approach to detect botnet based on multilayered architecture using ANFIS learning technique. To improve the overall system performance, we developed a traffic reduction technique to reduce the amount of network traffic. Experimental results show that the proposed BDS technique has high detection rate of 98.75% for malicious IP addresses and yields a low false positive rates (0–1.5%). Both results show that the proposed algorithm is not only efficient but also highly accurate. In addition, the proposed algorithm can detect inactive botnets, which can be used to identify potential vulnerable hosts. The high accuracy of this technique could play a vital role for botnet defense mechanism.

REFERENCES

1. Gu, G., R. Perdisci, J. Zhang and W. Lee, 2008. BotMiner: Clustering analysis of network traffic for protocol-and structure-independent botnet detection, in Proceedings of the 17th USENIX Security Symposium, pp: 139-154.
2. Strayer, W., D. Lapsley, B. Walsh and C. Livadas, 2008. Botnet detection based on network behavior," Advances in Information Security, Springer, 36: 1-24.
3. Livadas, C., R. Walsh, D. Lapsley and W.T. Strayer, 2006. Usilng machine learning technliques to identify botnet traffic, in: Proceedings of the 31st IEEE Conference on Local Computer Networks, IEEE, pp: 967-974.
4. Sadhan, B., J.M.F. Moura and D. Lapsley, 2009. Periodic behavior in botnet command and control channels traffic, in: Proceedings of the 28th IEEE Conference on Global Telecommunications, IEEE Press, pp: 2157-2162.
5. Choi, H., H. Lee and H. Kim, 2007. Botnet detection by monitoring group activities in DNS traffic, in: Proceedings of the 7th IEEE International Conference on Computer and Information Technology, pp: 715-720.
6. Gu, G., J. Zhang and W. Lee, 2008. Botsniffer: Detecting botnet command and control channels in network traffic, in: Proceedings of Network and Distributed System Security Symposium.
7. Dedinski, I., H. Meer, L. Han, L. Mathy, D.P. Pezaros, J.S. Sventek and X.Y. Zhan, 2009. Cross-layer peer-to-peer traffic identification and optimization based on active networking, in: Active and Programmable Networks: Proceedings of IFIP TC6 7th International Working Conference, pp: 13-27.
8. Kezi Selva Vijila, C., P. Kanagasabapathy and S. Johnson, 2005. Adaptive Neuro Fuzzy Interference System for Extraction of ECG, IEEE Indicon 2005 conference, Chennai, India.
9. Horikawa, S., T. Furuhashi and Y. Uchikawa, 1992. On Fuzzy Modeling Using Fuzzy Neural Networks with the Back-Propagation Algorithm" IEEE Trans. on neural networks, 3(5): 801-806.
10. Jang, J.S.R., 1993. ANFIS: Adaptive-network-based fuzzy inference system, IEEE Transactions on Systems, Man and Cybernetics, 23: 665-685.
11. Pal, N.R., K Pal, J.M. Keller and J.C. Bezdek, 2005. A possibilistic fuzzy c-means clustering algorithm. IEEE Transactions on Fuzzy Systems, 13(4): 517-30.