A Virtual Honeynet Based Botnet Detection (Vhbd) Architecture for Cloud

S. Nagendra Prabhu, S. Shanthi, R. Nidhya

Abstract*: Securing the cloud services from botnets has gained more attention in the recent years. As the cloud environment is flexible, reliable and scalable, the botnets can easily introduce thousands to millions of bots very easily. Thus, securing the cloud from the botnet is mandatory for preventing the services from various attacks such as Distributed Denial of Service (DDoS), spreading malware and hacking of private information. To prevent botnet from the cloud environment a Virtual Honeynet based Botnet Detection (VHBD) architecture is proposed in this paper. The suggested architecture defines the generation of the botnet using botmaster. Further, on receiving the access request from the cloud user, the VHBD checks the authenticity of the cloud user. If the user is authentic,the access permission is provided through an optimal honeypot installed on the guest OS. Whereas, if the user is non-authentic, the honeywall obtains the malicious IP of the botnet and saves them in the block list. The comparison of performance with the existing techniques prove that the proposed architecture provides optimal results than the other techniques.

***Index Terms*: Cloud botnet, honeypot, Virtual Honeynet (VH), Distributed Denial of Service (DDoS),botmaster, honeywall.**

I. INTRODUCTION

Botnet contains a collection of compromised bots controlled by the attacker named botmasters. The cloud bots are client-side applications that obtain the commands, processes them and generates the required reports. The bots are connected to each other through the master bots. The bots and master bots communicate with each other through the HTTP protocol. When compared to the traditional botnets the cloud bot is built in minutes and they are always ready and online. This makes the deployment of the botnet in cloud environment flexible. The key issues involved in the botnet design are minimal exposure and the demand for the prevention of Command and Control (C&C) compromise. Fig.1 illustrates the structure of the traditional cloud bot. Each bot communicates with the other bot through the cloud channel. In the traditional cloud bot generation, the botnet is created by infecting a computer without the knowledge of the owner. After infecting thecomputer with bot software, the botmaster is contacted. The botmaster then sends the orders to the bot for carrying out the tasks. Thus, thousands or millions of bots are created. Some of the common attacks created by thebotnets are as follows[1],

- Distributed Denial of Service (DDoS)
- Transmission of spam email
- Stealing private information
- Click fraud

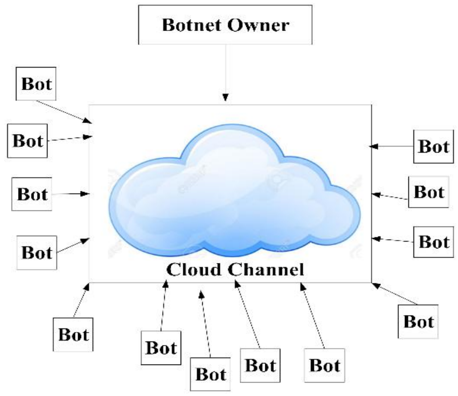

Fig.1. Structure of Botcloud

Generally, the botnet detection is performed using the techniques such as honeypot, honeynet, honeywall, Anomaly based detection and signature-based detection. Among the traditional methods, the honeynet is popularly used for the cloud environment. A simple structure of the honeynet is depicted in Fig.2. Generally, the different sizes of the honeypots are combined together for generating the honeynet. A honeypot can monitor only a smaller network, hence to monitor a larger network the honeynet is deployed.

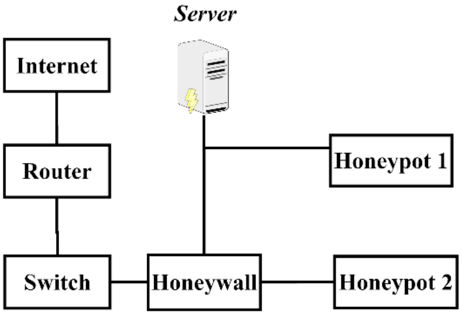

Fig. 2. Organization of honeynet

The demerits of the existing honeynets are minimal scalability and inability to detect the internet attacks. Thus, to address these issues, a honeynet based cloudbot network is designed. The suggested network has two participants such as an individual machine and a cloud server. The

Revised Manuscript Received on June 12, 2019.

S. Nagendra prabhu, Professor, Department of CSE, Malla Reddy College of Engineering & Technlogy, Hyderabad, Telangana, India.

Shanthi.S, Professor, Department of CSE, Malla Reddy College of Engineering & Technlogy,, Hyderabad, Telangana, India.

Nidhya R, Assistant Professor in the Department of Computer Science & Engineering, Madanapalle Institute of Technology & Science, affiliated to Jawaharlal Nehru Technical University, Anantapuram, India

*Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication*

individual machine has two components such as a botmaster and a social bot.

The bot master is a botherder that exploits the Command and Control (C&C) channel for generating the social bots in the cloud server. A social bot is an infected machine that transmits the infection to all the other nodes in the network. To block the attacker from accessing the cloud server, the virtual honeynet is proposed. The suggested virtual honeynet obtains the access request from the cloud user. On receiving the request the virtual honeynet validates the IP of the cloud user. If the IP is authentic, then the user is allowed to access the information, else the honeynet drags the user to a virtual server and tries to obtain the IP address of the attacker through a fake message. Thus, the server is protected from the unauthorized access.

Objectives

- To classify the users as valid and in-valid using the IP address.
- To design the VHBD architecture for preventing the original server from unauthorized access.
- To block the IP address of the botnet attacker from future entry.

The remainder of the paper is organized as follows, Section II illustrates the existing botnet detection techniques and algorithms. Section III describes the proposed honeynet based botnet detection system for detecting the botnet in the cloud. Section IV describes the results and discussions of the suggested method and the paper is concluded in section V.

II. RELATED WORK

This section illustrates the existing techniques and algorithms used for detecting the botnet in the cloud environment. *Eslahi, et al*[2] analyzed the botnet characteristics and their malicious activities. The existing botnet detection techniques and creation of botnets on the cloud and mobile environments were analyzed. *Nagendra, et al*[3] suggested a fuzzy pattern based filtering algorithm for detecting the botnets. The membership function was generated based on the bot behaviors such as generation of failed DNS queries, regular DNS query intervals, generation of failed network connections and same payload sizes for the network connections. The traffic reduction algorithm minimized the network traffic. Experimental analysis proved that the suggested system provided optimal detection rate, low false positive rate, and input raw packet traces. Further, it efficiently detected the inactive botnets. *Kebande, et al*[4] proposed an Artificial Immune System (AIS) based cognitive approach for detecting the botnet in the cloud environment. The merit of the suggested technique was an easy detection of the botnet. *Nagendra, et al*[5] structured the botnet behavioral features, defenses, and detection. Based on multiple criteria the botnet detection techniques were classified. The impact of behavioral features on the accuracy was analyzed. By integrating the complementary approaches, the information was exploited for devising the integrated detection strategies. *Fran, et al*[6] suggested a distributed computing framework, namely, map reducefor detecting the botnets in the cloud environment. The suggested framework exploited the host dependency model and adapted PageRank algorithm. Experimental analysis proved that proposed framework produced optimal results than the open-source based Hadoop cluster. *Garg, et al*[7] proposed a bot detection system for detecting the stealthy bots in networks. The suggested system considered the network traffic as the data stream and divided the traffic into parallel streams. The detection of the botnet was based on failure traffic and communication traffic information. The deployment of bot detection system on Hadoop Map-reduce proved that proposed system was scalable and accurate than the existing methods.*Ji, et al*[8] suggested a novel host-side detection approach for detecting the social bots. At first, the evasion mechanisms of the existing social bots were validated using the state-of-the-art detection approaches then a novel detection approach with two novel correlation mechanism was suggested for classifying the features into lifecycle and failure based. Based on the experimental results the false positive rate, false negative rate, F-measure value and detection rate of the proposed approach were 0.3%, 4.7%, 0.963 and 99.2%. *Rodríguez-Gómez, et al*[9] suggested a novel approach for detecting the abnormal behaviors of parasite P2P botnet resources. The suggested approach had three stages such as pre-processing, training and detection. The key advantage of the suggested approach was resource monitoring. Further, the false positive rate was 0.5% and the accuracy was greater than 99%. *Zhao, et al*[10] suggested a novel approach for detecting the activities of the botnet. The detection of the botnet was based on the traffic behavior analysis. The network traffic behavior was divided using the machine learning algorithms and the feasibility of the botnet detection activity was analyzed by classifying the behavior based on time intervals. Experimental results proved that the suggested approach detected the botnets with increased accuracy. *Kamaldeep Singh, et al*[11]proposed a big data analytic framework for detecting the Peer-to-Peer (P2P) botnet. The random forest based decision model was used for addressing the issues in the P2P botnet detection. The distributed dynamic feature extraction framework characterized the packet flow statistics. Further, the P2P security threat detection module efficiently classified the malicious traffic on the cluster. *Graham, et al*[12] suggesteda conceptual framework for estimating the presence of command and control botnet. The suggested framework was able to gather the communication traffic profiles and was also capable of preventing the detection and tampering of the malware in the virtualized environment.*Datar and Namrata*[13] analyzed the various botnet detection techniques in the cloud and proposed a data mining based botnet detection system for Collaborative Network Security Management System (CNSMS). The suggested CNSMS approach exploited the cloud storage for estimating the malicious attacks and also secured the cloud from the botnet.*Zhao, et al*[14] suggested a novel form of cloud-based push-styled mobile botnets for performing the command dissemination. The Cloud to Device Messaging (C2DM) service was used for creating the heart beat and command traffic. Efficient defense strategies were proposed for the push-styled mobile botnet. The evaluation results proved that the proposed C2DM was stealthy, resource-efficient and controllable.*Venkatesh, et* *al*[15]proposed a graph-based method, namely, BotSpot for detecting the nodes in the structured P2P botnet. The dense sub graphs were detected using the local search optimization method. Experimental results proved that the time complexity of the proposed technique had

a linear relationship with the volume of the traffic. Further, the proposed technique produced increased F-measure and robustness values. *Shoubao, et al*[16] proposed an Improved Dendritic Cell Algorithm (IDCA) for detecting the P2P bots. The raw data exploited for P2P botnet detection was retrieved from the API Trace tool. The suggested algorithm estimated the correlation between the behaviors of the normal process and the P2P bots. The correlation behavior was based on the antigen based signal specification. Experimental results proved that the proposed algorithm efficiently classified the abnormal processes.*Jadhav, et al*[17] designed a cloud-based system for detecting the android botnet malware. Based on the botnet detection learning dataset and multi-layered algorithm the botnet family was estimated for a particular application. *Jiang and Shao*[18] detected the P2P botnets usingcommand and control (C&C) communications. By exploiting the clustering technique the suggested approach detected the P2P bots and normal hosts. Experimental results proved that the proposed approach integrated with flow dependency efficiently detected the P2P botnet. Further, the suggested approach provided higher detection rate and lower false positive rate. *Yeh, et al*[19] suggested an automatic botnet detection and notification system for notifying the users about the malware infections on their computers. The suggested approach detected the infected bots with 15 minutes. Hence, the suggested approach was able to restrain the scale of the botnets. *Zhang, et al*[20] suggested a novel botnet detection system for detecting the stealthy P2P botnets. At first, the hosts involved in the P2P communication were estimated then the statistical fingerprints used for the P2P botnet traffic and legitimate P2P traffic classification was obtained. The experimental results proved that the suggested framework was able to detect the stealthy P2P botnets with increased detection accuracy and lower false positive rate. *Bhatia, et al*[21] suggested a virtual honeynet based data collection mechanism for detecting the botnet. Experimental results proved that the suggested approach efficiently detected the IRC and HTTP botnets.From the analysis of the existing botnet detection techniques and algorithms, it is clear that the existing techniques do not provide satisfactory botnet detection accuracy for the cloud environment. Thus, the VHBD architecture is proposed for detecting the attackers in the cloud environment.

III. PROPOSED METHODOLOGY

This section illustrates the proposed virtualhoneynet based botnet detection system for detecting the botnet cloud. The overall flow of the proposed botnet detection system is illustrated in Fig.3.

A. *Botnet generation*

The generation of the attacker is performed in two steps. Initially, the botmaster who controls the bot is established then the social bot is created by infecting a system without the knowledge of the system owner. The infection is spread across multiple systems for creating a botnet. The botmaster takes control of the botnet and sends orders to the bot for carrying out the tasks. The tasks of the botmaster are performed using three components such as,

- Bot worker
- Bot updater
- Master controller

i. Bot worker

The bot worker component is responsible for the creation and maintenance of the social bot. The creation of the social bot includes a creation of the socially attractive user profile. The user profile is generated by three main tasks such as active e-mail address, the creation of user profile and solving the captcha. It is considered that every user has a single profile and multiple user accounts are managed by an individual organization. After the creation of the user profile, the credentials of the profile such as user name and password are provided to the social bot malware for controlling the entire profile.

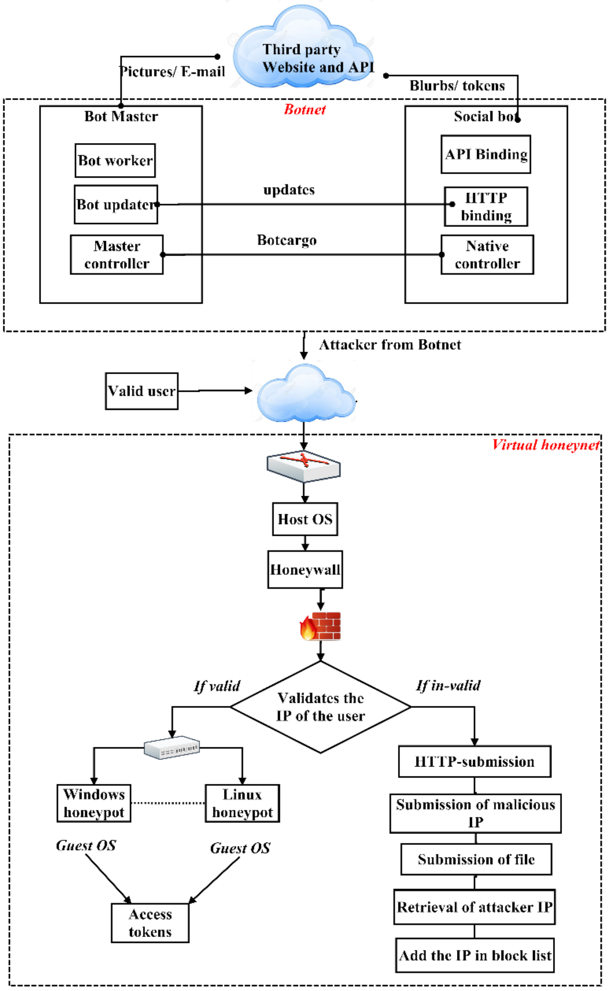

Fig.3. Architecture of the proposed botnet detection system

ii. Bot updater

This component enables the software updates such as native commands and updated HTTP request templates for the social bot. The updates are

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

provided to the social bot through the Command and Control (C&C) channel. The C&C engine collects and maintains the master commands and executes them on the master controller.

iii. Master controller

The master controller component communicates with the C&C engine and defines a set of master commands for enabling the master controller to send the commands to the social bots. The master controller maintains three phases such as setup, bootstrapping and propagation. During the setup phase, the botmaster builds the social bots and updates the malware. In the bootstrapping phase, the social bot connects with multiple user profiles and breaks the connectivity between social bots. During the propagation phase, the botmaster makes the social bot collect all the neighborhood information and return them as botcargo.

The communication between the social bots is performed using two channels such as C&C channel and socialbot-Online Social Network (OSN) channel. The social bot-OSN channel is meant for carrying the OSN specific Application Programming Interface (API) calls and normal HTTP traffic. The HTTP traffic can be generated by increased activity or can also be created by the normal user.

B. *Virtual honeynet*

Virtualization is the process of allowing more than one honeynet in a single system. The honeynet network is deployed on a cloud server that contains a firewall gateway for controlling and capturing the data. The access request from the user is forwarded to the virtual honeynet for processing. The user can be either a valid user or can be an attacker from the botnet. On receiving the request, the virtual honeynet validates the user. If the user is valid the access request is forwarded to the original server for retrieving the information. Whereas, if the user is in-valid, the user is blocked from accessing the original server. On receiving the access request from the users, the router routes the request to the host OS.

i. Honeywall

The honeywall is a gateway device that acts as anentry and exit points for all the network traffic for the honeynet. Thus, the entire network traffic of the honeynet is monitored. The key objectives of the honeywall are controlling, capturing and alerting the data. The honeywall together with the firewall validates the authenticity of the user.

ii. Distribution of access tokens to the user

If the IP address of the user matches with the registered IP list, then the user is switched to the suitable guest honeypots for accessing the original server. The access rights are provided in terms of access tokens.

iii. Prevention of attacker

On the other hand, if the user IP does not match the registered IP list, the honeynet obtains the access request from the attacker and provides a duplicate web page as the search result. When the attacker accesses the duplicate web page, the IP address of the botnet attacker is obtained and saved in the block list. Thus, the attackers are prevented from accessing the original server in future.

Advantages of virtual honeynet

- Minimal cost
- Minimal memory consumption
- Portable
- Easy management

IV. PERFORMANCE ANALYSIS

This section illustrates the comparison of the performance for the existing ANFIS, Machine Learning Technique (MLT), Page rankand the proposed VHBD architecture. The metrics that are used for validating the performance of the proposed system are as follows,

- Detection rate Vs Packet time interval
- False positive rate
- False negative rate
- Precision
- Recall

A. *Detection rate Vs Packet time interval*

The detection rate is defined as the amount of time consumed between successive packet transmissions. The comparison of detection rate with respect to the packet time interval for the existing ANFIS, MLT, page rank techniques and the proposed VHBD technique is depicted in Fig. 4. From the figure, it is analyzed that the suggested VHDB architecture provides increased detection rate than the existing algorithms.

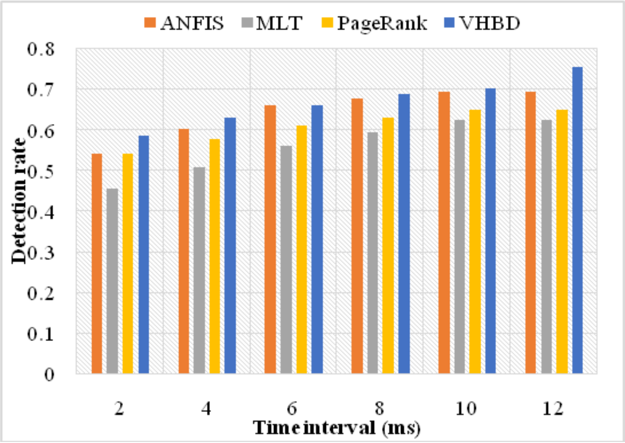

Fig. 4. Comparison of detection rate for the existing and the proposed algorithms

B. *False positive rate*

The false positive rate is the proportion of valid IP address detected as bots. Increased false positive rate minimizes the overall performance of the system. The comparison of false positive rate for the existing ANFIS, MLT, page rank algorithms and the proposed VHBD architecture is depicted in Fig. 5. The analysis results show that the proposed VHBD architecture provides minimal false positive rate than the existing algorithms.

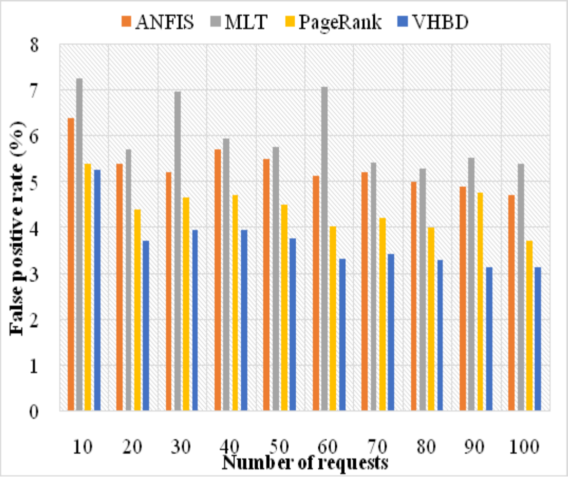

Fig. 5. Comparison of false positive rate for the existing and the proposed algorithms

C. *False negative rate*

The false negative rate is defined as the proportion of malicious IP address not detected as the botnet. It is mandatory to minimize the false negative rate because the increase in this rate minimizes the overall performance of the botnet detection system. The comparison of thefalse negative rate for the existing ANFIS, MLT, page rank and the proposed VHBD architecture is depicted in Fig. 6. The analysis results show that the proposed VHBD architecture provides minimal false negative rate than the existing algorithms.

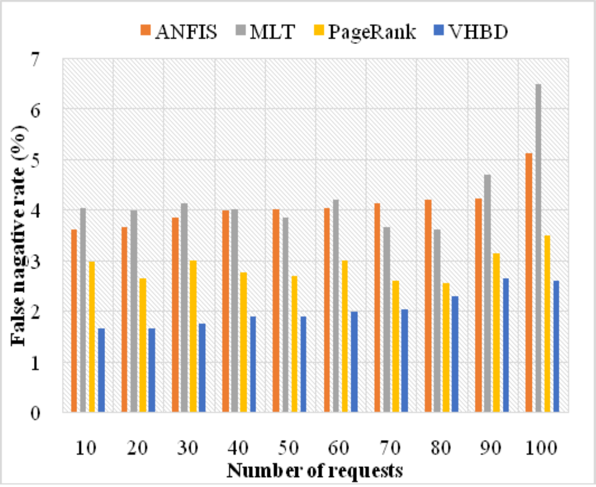

Fig. 6. Comparison of false negative rate for the existing and the proposed methods

D. *Precision*

The precision measure estimates the purity of the cloud environment by considering the fraction of botnet nodes to the total number of nodes in the cloud.

$$Precision = \frac{|b \cap c|}{|c|} \qquad (1)$$

Where,

b represents the number of malicious IP address

c denotes the total number of IP addresses that enters the VHBD

The comparison of precision for the existing ANFIS, MLT, page rank algorithms and the proposed VHBD architecture is depicted in Fig. 7. The analysis results prove that the proposed VHBD provides increased precision values than the existing algorithms.

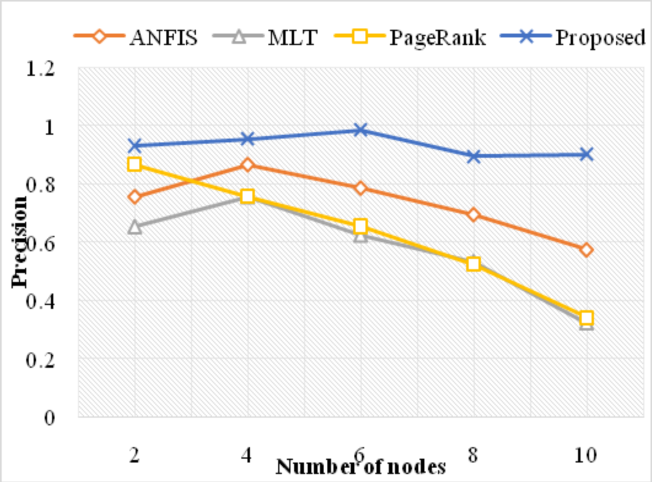

Fig. 7. Comparison of precision for the existing and the proposed methods

E. *Recall*

The recall measure estimates the total fraction of bots being identified.

$$Recall = \frac{|b \cap c|}{|b|} \qquad (2)$$

The comparison ofrecall for the existing ANFIS, MLT, page rank algorithms and the proposed VHBD architecture is depicted in Fig. 8. The analysis results prove that the proposed VHBD architecture provides increased recall than the existing methods.

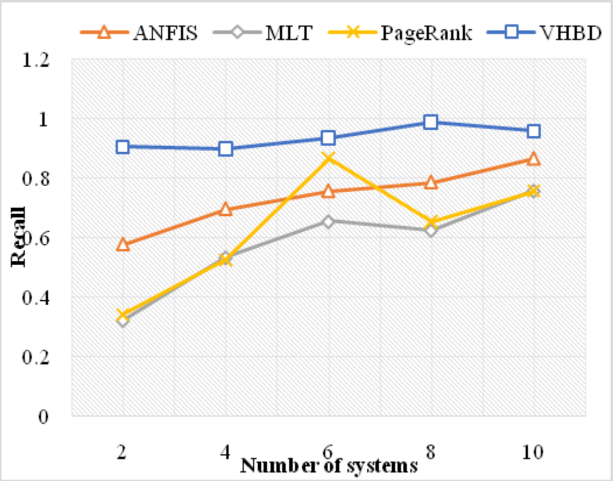

Fig. 8. Comparison of recall for the existing and the proposed methods

V. CONCLUSION

In this paper, an efficient virtual honeynet based botnet detection architecture is proposed for detecting the botnet in the cloud environment. The suggested architecture includes two functions such as the description of botnet generation and prevention of botnet from accessing the original server. On receiving the server access request from the cloud user, the honeywall of the suggested architecture validates the IP address of the cloud user. If the IP address is matched with the list of the valid IP addresses the cloud user is allowed to access the original server through the honeypot.Whereas, if the cloud user is found to be malicious, the IP address of the cloud user is retrieved and saved in the block list. Thus, the request from the botnet is prevented in future. By comparing the performance of the proposed VHBT architecture with the existing ANFIS, MLT and page rank techniques, the superiority of the proposed architecture is validated. The comparison results prove that the suggested architecture provides optimal precision, recall, detection rate, false positive rate and false negative rate.

ACKNOWLEDGMENT

We would like to thank Dr. VSK Reddy, Principal, Malla Reddy College of Engineering and Technology and Dr. D. Sujatha, HOD, CSE dept., MRCET for their kind help and motivation of doing research.

REFERENCES

1. K. C. M. W. F. M. T. Brazier, "Bot-Clouds The Future of Cloud-based Botnets?," 2010.
2. M. Eslahi, R. Salleh, and N. B. Anuar, "Bots and botnets: An overview of characteristics, detection and challenges," in *IEEE International Conference on Control System, Computing and Engineering (ICCSCE)*, 2012, pp. 349-354.
3. S. Nagendra Prabhu, and D. Shanthi, "An Efficient Botnet Detection System in Large Scenario Networks Using Adaptive Neuro Fuzzy Inference System Classifier," Journal of Computational and Theoretical Nanoscience, vol. 14, issue no. 4 pp. 1-5, 2017.
4. V. R. Kebande and H. S. Venter, "A cognitive approach for botnet detection using Artificial Immune System in the cloud," in *Third International Conference on Cyber Security, Cyber Warfare and Digital Forensic (CyberSec)*, 2014, pp. 52-57.
5. S. Nagendra Prabhu, and D. Shanthi, "A Survey on Anomaly Detection of Botnet in Network," *International Journal of Advance Research in Computer Science and Management Studies*, vol. 02, issue 1, pp. 552-558, 2014.
6. J. Fran, x00E, ois, S. Wang, W. Bronzi, R. State, *et al.*, "BotCloud: Detecting botnets using MapReduce," in *IEEE International Workshop on Information Forensics and Security (WIFS)*, 2011, pp. 1-6.
7. S. Garg, S. K. Peddoju, and A. K. Sarje, "Scalable P2P bot detection system based on network data stream," *Peer-to-Peer Networking and Applications*, pp. 1-17, 2016.
8. Y. Ji, Y. He, X. Jiang, J. Cao, and Q. Li, "Combating the evasion mechanisms of social bots," *Computers & Security*, vol. 58, pp. 230-249, 5// 2016.
9. R. A. Rodríguez-Gómez, G. Maciá-Fernández, P. García-Teodoro, M. Steiner, and D. Balzarotti, "Resource monitoring for the detection of parasite P2P botnets," *Computer Networks*, vol. 70, pp. 302-311, 2014.
10. D. Zhao, I. Traore, B. Sayed, W. Lu, S. Saad, A. Ghorbani, *et al.*, "Botnet detection based on traffic behavior analysis and flow intervals," *Computers & Security*, vol. 39, Part A, pp. 2-16, 11// 2013.
11. S. C. G. Kamaldeep Singh, Abhishek Thakur, Chittaranjan Hota "Big Data Analytics framework for Peer-to-Peer Botnet detection using Random Forests," *Information Sciences*, vol. 278, pp. 488-497, 2014.
12. M. Graham, A. Winckles, and E. Sanchez-Velazquez, "Botnet detection within cloud service provider networks using flow protocols," in *IEEE 13th International Conference on Industrial Informatics (INDIN)*, 2015, pp. 1614-1619.
13. N. A. s. a. Datar, "A Review-Botnet Detection and Suppression in Clouds " *Journal of Information Engineering and Applications* vol. 3, pp. 1-6, 2013.
14. S. Zhao, P. P. Lee, J. Lui, X. Guan, X. Ma, and J. Tao, "Cloud-based push-styled mobile botnets: a case study of exploiting the cloud to device messaging service," in *Proceedings of the 28th Annual Computer Security Applications Conference*, 2012, pp. 119-128.
15. B. Venkatesh, S. H. Choudhury, S. Nagaraja, and N. Balakrishnan, "BotSpot: fast graph based identification of structured P2P bots," *Journal of Computer Virology and Hacking Techniques*, vol. 11, pp. 247-261, 2015.
16. Y. S. Shoubao Su, Mingjuan Xu and Xianjin Fang, "An Improved Dendritic Cells Algorithm for Detecting P2P Bots," *International Journal of Grid and Distributed Computing*, vol. 9, pp. 117-126, 2016.
17. S. Jadhav, S. Dutia, K. Calangutkar, T. Oh, Y. H. Kim, and J. N. Kim, "Cloud-based Android botnet malware detection system," in *17th International Conference on Advanced Communication Technology (ICACT)*, 2015, pp. 347-352.
18. H. Jiang and X. Shao, "Detecting P2P botnets by discovering flow dependency in C&C traffic," *Peer-to-Peer Networking and Applications*, vol. 7, pp. 320-331, 2012.
19. L.-Y. Yeh, Y.-L. Tsai, B.-Y. Lee, and J.-G. Chang, "An Automatic Botnet Detection and Notification System in Taiwan," in *Proceedings of the International Conference on Security and Management (SAM)*, 2013, p. 1.
20. J. Zhang, R. Perdisci, W. Lee, U. Sarfraz, and X. Luo, "Detecting stealthy P2P botnets using statistical traffic fingerprints," in *IEEE/IFIP 41st International Conference on Dependable Systems & Networks (DSN)*, 2011, pp. 121-132.
21. R. K. S. J.S.Bhatia, Sanjeev Kumar, "Botnet Command Detection using Virtual Honeynet," *International Journal of Network Security & Its Applications (IJNSA)*, vol. 3, pp. 177-189, 2011.

AUTHORS PROFILE

Dr. S. Nagendra Prabhu currently working as Professor, Department of Computer Science and Engineering, Malla Reddy College of Engineering & Technlogy, Dhulapally, Secunderabad, India, an autonomous Institution under the affiliation of Jawaharlal Nehru Technological University, Hyderabad. He completed his PhD in Network security in cloud computing from Anna University, Chennai, India. He completed his Master of Engineering in Network Engineering from Anna University. His research interest includes Cloud computing, Botnet attack, Web based network Security. Currently the author is doing research related security issues in Cloud Computing.

Dr. S.Shanthi received her Ph.D. degree from University Of Mysore, Mysore, India, in 2016, and M.E Degree from Sathyabama University, India in 2008. She is currently working as a Research Professor with Malla Reddy College of Engineering and Technology, Hyderabad, Telangana, India, an autonomous Institution under the affiliation of Jawaharlal Nehru Technological University, Hyderabad. She is an author and co-author of more than 20 papers in Technical Journals and Conference Proceedings, and she has contributed to Book Chapters in her areas of interest and to her credit she has 2 patents. Her research interests include image processing, wireless Adhoc and sensor networks, machine learning, Network Security.

Dr. R.Nidhya is presently working as Assistant Professor in the Department of Computer Science & Engineering, Madanapalle Institute of Technology & Science, affiliated to Jawaharlal Nehru Technical University, Anantapuram, India. She received the M.Tech and Ph.D degree from Anna University, Chennai. Her research interests include wireless body area network, network security and data mining. She published 12 papers in refereed international journals and 13 papers in conferences. She is an active member of ISTE, IAENG, SAISE and ISRD.