User Traffic Profile for Traffic Reduction and Effective Bot C&C Detection

Soniya Balram and M. Wilscy
(Corresponding author: Soniya Balram)

Department of Computer Science, University of Kerala, Kariyavattom, Trivandrum, India – 695581
(Email: soniya.balram@gmail.com)

(Received Mar. 29, 2012; revised and accepted Nov. 6, 2012)

Abstract

Bots are malicious software components used for generating spams, launching denial of service attacks, phishing, identity theft and information exfiltration and such other illegal activities. Bot detection is an area of active research in recent times. Here we propose a bot detection mechanism for a single host. A user traffic profile is used to filter out normal traffic generated by the host. The remaining suspicious traffic is subject to detailed analysis. The nature of detailed analysis is derived from a characterization of bot traffic. The detection system is tested using two real world bots. Performance evaluation results show that the proposed system achieves a high detection rate (100%) and a low false positive rate (0–8%). The traffic filtering yielded an average reduction in traffic by 70%.

Keywords: Bot traffic characterization, host-based bot detection, traffic reduction

1 Introduction

A botnet is a network of compromised machines under the influence of malware (bot) code. The botnet is commandeered by a "botmaster" and utilized as "resource" or "platform" for attacks such as distributed denial-of-service (DDoS) attacks, and fraudulent activities such as spam, phishing, identity theft, and information exfiltration. In order for a botmaster to command a botnet, there needs to be a command and control (C&C) channel through which bots receive commands and coordinate attacks and fraudulent activities. The C&C channel is the means by which individual bots form a botnet [6]. Botnets are one of the most dangerous species of network-based attacks today because they involve the use of very large, coordinated groups of hosts for both brute-force and subtle attacks. Botnets derive their power by scale, both in their cumulative bandwidth and in their reach [16].

Botnets and their detection has been an active area of research in recent times. Many detection techniques have been proposed based on honeynets, DNS activities, network traffic, host based logs and so on. Our scope of bot detection is for a specific host and particularly focusing on "home computers" which includes home-based PCs, laptops, tablets and other such devices which directly connect to the Internet. These devices with an always-on Internet connection are increasingly used for online shopping, banking and other transactions. But most users do not put in the effort to secure these devices with protection software and timely updates. This enables criminals to recruit these vulnerable devices for their nefarious purposes. Bots infect the system unobtrusively and remain unobserved and stealthy on the system. They are many nowadays being used for harvesting personal information as well as information such as credit card details which can be sold in the organized underground crime market. This makes detection of malicious bots on "home computers" so important.

Based on their architecture, bots can be characterized as centralized or distributed [10]. In centralized architecture, there is a single C&C server with which the bots communicate for receiving commands and sending updates. The weakness of centralized architecture is that the bot master or the C&C server can be easily identified and brought down. Some bots also use a peer to peer (P2P) architecture. P2P architecture is more resilient against failures or take-over attempts by defenders. Another approach for classification of botnets is based on the communication protocol used between C&C servers and bots. Botnets can be classified as Internet Relay Chat (IRC) based, Hypertext transport protocol (HTTP) based or P2P based. IRC based bots are the most prevalent but nowadays most organizations use firewalls to block IRC traffic. HTTP bots are used to circumvent this and can pass off as normal traffic. P2P bots are less common but are expected to grow fast in the near future [7].

Bots have evolved over the years and employ several evasive techniques to avoid detection including use of packers [21], polymorphism and other code obfuscation methods [11], rootkit techniques. But it remains a fact that bots need to communicate with their bot master in order to be of use. This communication cannot be distorted or manipulated to evade detection. Hence bot detection through examining the traffic generated by the host or more

specifically, identification of bot C&C traffic on the host machine assumes relevance. There are several challenges for detecting bot C&C traffic [18]. Bots are mostly inactive and stealthy in their communication. The C&C traffic generated is very similar to normal traffic especially with HTTP bots. The volume of traffic generated by the bots is also low. Besides, more recent bots use encryption in their C&C hence preventing any payload inspection for detection.

We propose a detection mechanism for bot C&C traffic by analysing "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&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&C traffic are made and an algorithm is proposed for detailed analysis and bot detection. The evaluation of our proposed system yielded a detection rate of 100% with 8% false positives. The traffic filtering yielded an average reduction in traffic by 70%.

The rest of the paper is organized as follows. Section 2 reviews related work. The characterization of bot behavior and the details of the proposed system are covered in Section 3. Section 4 presents the experimental setup used and the results. Concluding remarks and future directions are presented in Section 5.

2 Related Work

Many of the earlier works in botnet detection were network-based [6, 16, 18]. They detect bots by looking for similar traffic from various hosts on the network to a common destination. Network-based detection systems have produced wonderful results but would be unsuitable for detecting a single bot infested host. There are several host-based approaches to bot detection as well. Here we briefly present some of the techniques and related issues:

Jose Andre Morales et al. [8] proposed a detection strategy based on DNS activities of the host, digital signatures and process/file system tampering, the absence of a GUI and no required reading of user input. It is possible that with advanced rootkit techniques of bots some of these properties might be suppressed by the bot to evade detection. Another approach is to use presence of attack traffic, like generation of scan, email spam and DoS for detection [1, 22]. These methods would detect the bot only after it serves its purpose of launching attacks.

Bot detection based on analysis of network packets generated by the host is another method [17, 20]. It is based on the assumption that bots need to communicate with their masters for them to be of use. No suppression or evasion techniques can be applied on this bot behavior. The research presented in [17] proposed a system which monitors outbound packets from a host and compares with destination-based whitelists. The white-lists are generated by observing an un-infected PC. Although this is a straightforward technique, the detection can be done only during the non-operating time of the PC.

The work in [20], proposed by H Xiong et al, is a host-based bot detection system for HTTP traffic. The detection system is based on the assumption that users have low diversity in the web sites. Out-of-band retrieval and analysis of requested web page is done. Only white-listed web page requests are permitted. The user is informed and asked to take a decision about non white-listed requests. This would be intrusive to the user. Besides out-of band retrieval increases the bandwidth usage by a factor of 2 and slows down the user's browsing experience. The proposed system also has similar assumptions about the destinations contacted by the user, but does not use out-of-band data retrieval nor involves the user in the decision making.

3 Proposed Method

The premise on which the current work is based is that a user tends to use only a common limited set of applications for his tasks whether for education, entertainment or work. So the traffic generated by the user also belongs to a specific set or pattern. This fact is used to generate a traffic profile for a particular user. The profile is generated dynamically after analyzing flows to all destinations in a timeslot. Currently, the profile is limited to a set of destinations contacted by the user. The profile is maintained as an xml file.

The profile is used by the detection system to filter out normal traffic. The abnormal traffic is characterized as "suspicious" and is further analyzed to zero in on bot traffic. The detailed analysis of suspicious traffic is based on the observations made in the characterization of bot traffic described in Section 3.1.

Processing of traffic is done on a timeslot basis. A timeslot is intuitively chosen to be 30 minutes. Smaller timeslots entail frequent calls to the detailed analysis module. Besides, bots tend to keep a low profile in their C&C communication to avoid detection. So, frequency of communication is kept a minimum. It is observed in the study of bot behavior that bots communicate in intervals of 10 minutes, 20 minutes or even more, with the bot master. So smaller time intervals would not pick up bot signals. At the same time, very large timeslots would require processing of large number of packets in a single go. Hence in the current work we limit the timeslot to 30 minutes. A generic detection mechanism independent of the timeslot size would be implemented in the future.

In the following sections, the various modules which make up our proposed system are presented.

3.1 Characterization of Bot C&C Traffic

Since bot traffic traces are not easily available, bot

characterization is done through setting up of a small network of 10 machines on a DETER [3] testbed to study the bot generated traffic. DETER testbed is a facility available to researchers for conducting experiments in the area of computer security. It is supported by USC Information Sciences Institute (ISI), and the University of California, Berkeley.

The work in [2, 3, 13, 15] are also used to understand the behavior of HTTP bots. HTTP bots are considered since more botnets have discarded IRC and are using HTTP and P2P as their C&C mechanism. On home computers, in which HTTP comprises the predominant traffic, HTTP bots can hide themselves easily.

Based on the characterization and the study mentioned above, the following observations are made regarding bot traffic:

1) In a successfully configured botnet system, the bots communicate periodically with the bot master for updating their status information as well as getting updated configuration information.
2) Bots generate repeated DNS queries to resolve the domain names of their bot masters.
3) On unsuccessful name resolution using DNS, NetBIOS name resolution is also repeatedly tried.
4) Some bots also try to generate scan traffic regularly in order to contact the bot masters.
5) On unsuccessful name resolution, some bots try to contact bot masters through hard coded IP addresses.
6) Bot code might be injected into one or more benign processes on an infected host.
7) The destinations being contacted by the bots are not the ones commonly accessed by a normal host.

The bot detection method put forward in the following section is built on these observations.

3.2 Building the Normal Profile

Normal Profile for user traffic is generated on-the-fly by examining the destination domain names/IP addresses of outgoing network packets. We start with an initial seed list of commonly occurring destinations like www.google.com. Additions to the seed list are made after examining flows to each of the destinations. A flow is a set of packets which share the same (Source IP, Source Port, Destination IP, Destination Port, Protocol) tuple. As explained in detail in Section 3.4.1, normal traffic shows a specific pattern. The flows to destinations are analysed and destinations which exhibit normal behavior are added to the normal profile. Algorithm 1 explains how the profile is built.

The profile is maintained as an xml file which is dynamically updated.

3.3 Traffic Filtering

Algorithm 1 –Build-Normal-Profile(SL,NP,t)

1: Begin
2: Initialize SL = {google.com, yahoo.com, microsoft.com, hotmail.com} is the seed list for destinations in the normal profile;
3: NP = {set of all destinations d which exhibit normal behavior as mentioned in Section 3.4.1};
4: t is the traffic captured for the timeslot.
5: if NP == ∅
6: NP = SL
7: From t, eliminate all packets such that destination d ∈ NP
8: for destination d ∉ NP
9: Do detailed analysis of traffic to **d**
10: If traffic to **d** shows behavior described in Section 3.4.1
11: NP = NP U **d**
12: End

Bot detection is preceded by traffic filtering in order to reduce the amount of traffic on which detailed analysis need to be done. Packets which contact destinations in the normal profile are considered "innocent" and are filtered out. The remaining packets are termed "suspicious" and sent for detailed analysis.

3.4 Detailed Analysis

Traffic to "suspicious" destinations is subjected to detailed bot analysis, to classify it as bot or normal. It involves analysis of the traffic along the following lines:

1) Look for similar flows to a destination IP or domain at periodic intervals. This indicates an active bot.
2) Find periodic failed DNS queries to the destination domain.
3) Find many Netbios queries to a destination IP or domain.
4) Look for many SYN scans in the traffic.

Any of the last three behaviors indicates an inactive bot. An inactive bot is a bot that is not able to connect to its command and control (C&C) server either temporarily or persistently [19].

3.4.1 Active-Bot Detection

Suspicious traffic in a single time slot is grouped into flows. The flows are grouped based on destination IP or domain name, so as to separate flows to each destination. Now the time-gap between flows in the same group are found.

In most "home computers", the major component of "suspicious" destinations is from browser traffic. Although users follow a specific pattern in web usage [20], at certain times, new destinations are contacted. So the "suspicious" destinations could be due to browser traffic or possible

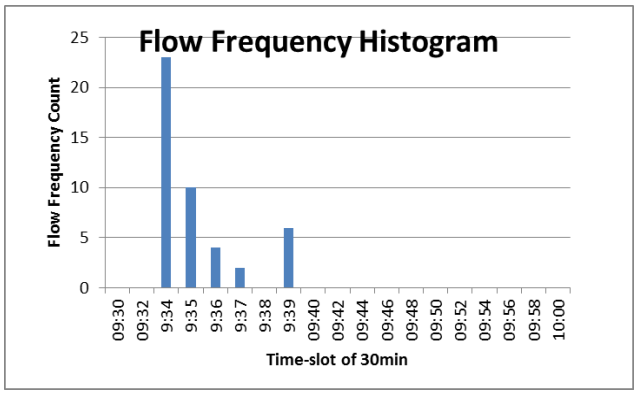

Figure 1: Browser traffic

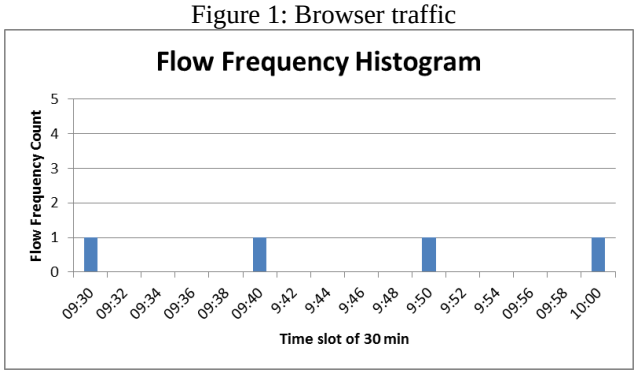

Figure 2: Bot traffic

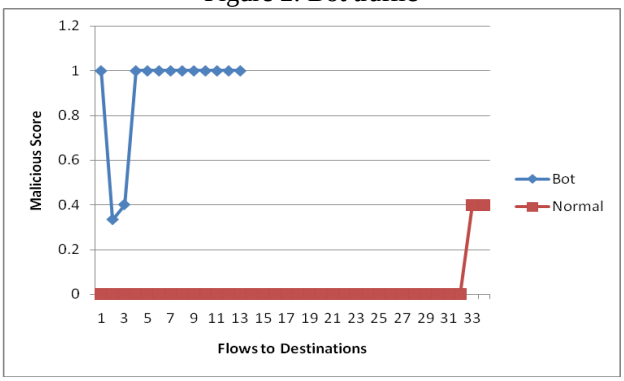

Figure 3: Malicious scores

malicious bots. We need a mechanism to distinguish between the two. It is observed that browser traffic is bursty resulting in multiple flows to the same destination over a short time period. But bots have evenly spaced out flows over a large time. This behavior is shown in the Figures 1 and 2 respectively.

It is seen that the gap between consecutive flows in the traffic is very small in browser traffic while there is a large and repeating gap in the latter. This characteristic of traffic is used to distinguish between browser and bot traffic. A malicious score for the traffic is computed based on the number of "large" time gaps between consecutive flows. The work [12], analyzes the difference between starting times of web flows. It was found that 80% of the intervals were within 1000ms and the remaining 20% intervals were in the range 1s to 100s. From these findings we arrive at a threshold for "large" time gaps as 2 minutes which is slightly higher than the upper bound of 100s mentioned in the paper.

Malicious score of traffic to the destination = number of "large" time-gaps between flows / total no: of flows.

Figure 3 shows how the malicious score differs for browser and bot traffic.

3.4.2 In-Active Bot Detection

The traffic output by the filtering module has only flows to "suspicious" destinations. In the Inactive Bot Detection module, the traffic thus produced is scoured for the 3 behavioral patterns mentioned in Section 4.2.

1) All failed DNS queries to a destination in a time slot are grouped. Failed DNS queries are few in normal traffic. Periodic failed DNS queries are rarer. A malicious score is defined for periodic failed DNS queries as

   Malicious Score = {1-(1/no: of failed DNS queries)} + { 1-(1/no: of periods).}

   It takes into consideration the number of failed DNS queries and the number of periods.

2) The normal user profile mentioned also has a normal destination list for Netbios traffic. Any Netbios query to other "suspicious" destinations, exceeding a threshold of 5 per timeslot, is marked as bot traffic.

3) SYN scan traffic is detected using the algorithm mentioned in [14].

4 Evaluation

This section presents the evaluation of the bot detection algorithm presented. A brief description of the experimental setup for the evaluation and classification accuracy of the detection mechanism is presented.

4.1 Experimental Setup and Data Collection

DETER testbed [3] is used to setup a botnet for observing the bot behavior. A Zeus [2] botnet is set up with one botmaster and 9 bots on Windows XP SP2 machines. The traffic generated by bots are observed. It is noticed that the bots periodically communicated with the botmaster to update their status as well as to get updates and configuration information. The domain names of the botmaster are configured into the bot clients. In cases where the bots are not able to contact the bot master, they periodically tried to generate scan traffic, DNS queries, NetBIOS queries. It is also observed that no new process was created in the bot client, but the bot injected itself into bot processes services.exe and explorer.exe.

A prototype detection system is implemented on a Windows XP host machine. Microsoft Network Monitor 3.4 is used to capture traffic generated by the processes on the host. Testing is done for traffic generated by Zeus and BlackEnergy [9] bots.

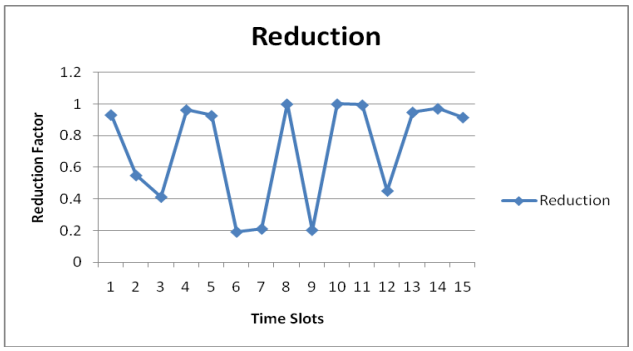

Figure 4: Traffic reduction

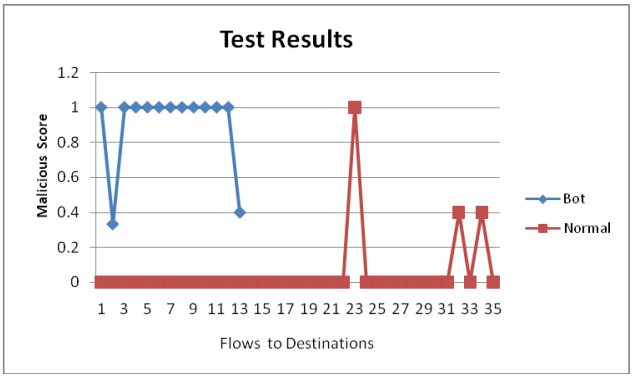

Figure 5: Test results

4.2 Result

4.2.1 Traffic Reduction

Traffic filtering using normal profile brings about a reduction in traffic processed by the detailed analysis module by a factor of 70% or more. The Figure 4 shows the reduction in traffic for various time slots.

4.2.2 Bot Detection

The traffic to "suspicious" destinations is analysed and designated as either normal or bot. Figure 5 shows the malicious score values for bot and normal data instances in test data. A malicious score value greater than 0.1 is designated as bot. This is justified because even a small percentage of flows separated by a periodic gap represents bot traffic.

It is seen from Figure 5 that bot instances are classified correctly. It also shows 3 false positives out of the 35 normal data instances. The following metrics are used to estimate the goodness of the classification provided by our algorithm.

1) Detection Rate or Sensitivity of the detection system is defined as the number of malware instances detected by the system divided by the total number of malware instances present in the test set. For our system, the detection rate is 100%.

2) Specificity is the true negative rate, that is, the proportion of negative instances that are correctly classified. Our system shows a specificity of 92%.

3) Precision denotes the percentage of data samples which are really positive out of the total number which are classified as positive by the system which is 81% for our system.

4) False Positive Rate is the number of normal data instances incorrectly classified as malware. Our system generates 3 false positives out of 35 normal samples resulting in a false positive rate of 8 %.

A detailed look at the traffic revealed that two of the false positive samples stand for browser update traffic. It is felt that use of a sliding window of timeslots, with correlation with other timeslots could resolve the false positives. Further study needs to be done in this direction. Table 1 shows a comparison of performance of several bot detection techniques. It is seen that our technique has a high detection rate with a relatively low false positive rate. We have used two real bots in our evaluation and have achieved a high rate of traffic reduction too. But more number of bot samples need to be considered for testing the universality of the detection system.

Table 1: Comparison of bot detection techniques

Approach	Livadas et al. [16]	Gu et al. [6]	K Wang et al. [19]	Fednyshyn et al. [4]	The proposed Technique
Core Technique	Machine Learning	Spatial Temporal Correlation	Fuzzy Pattern Recognition	Data Mining - Classification	Statistical Thresholding
Bot Samples	1	8	44	7	2
Rate of Traffic Reduction	N/A	N/A	More than 70%	N/A	More than 70%
Inactive bot detection	No	No	Yes	No	Yes
True positive Rate	92%	100%	95%	92.9%	100%
False Positive Rate	11-15%	0-6%	0-3.08%	7.8%	0-8%

5 Conclusion and Future Work

We conclude that our approach is valid for detecting the bot traffic and the destination contacted by the bot. We have achieved a detection rate of 100% with a false positive rate of 8%. Our approach is a host-based approach intended for "home computers" which are vulnerable to phishing, data stealing and data exfiltration which happens stealthily. Our approach also achieves a data reduction by a factor of 0.7 through profile based filtering.

The false positives can be reduced by using a sliding window method over various time slots. Besides our work does not consider IRC or P2P traffic which are also very actively used by bots as a C&C mechanism. The time slot considered is of 30 minutes duration. Future works intend to overcome this limitation with a more generic detection algorithm.

References

[1] J. R. Binkley and S. Singh, "An algorithm for anomaly-based botnet detection," in Proceedings of USENIX Steps to Reducing Unwanted Traffic on the Internet Workshop (SRUTI), pages 43-48, July 2006.

[2] H. Binsalleeh , T. Ormerod , A. Boukhtouta , P. Sinha , A. Youssef , M. Debbabi , and L. Wang, "On the analysis of the zeus botnet crimeware toolkit," in *Eighth Annual International Conference on Privacy, Security and Trust* , pp. 31-38, 2010.

[3] R. Borgaonkar, "An analysis of the asprox botnet," in *4th International Conference on Emerging Security Information, Systems and Technologies*, pp. 148-153, 2010.

[4] DETERlab. http://www.isi.deterlab.net/

[5] G. Fedynyshyn, M. C. Chuah, and G. Tan, "Detection and classification of different botnet C&C channels," in *Proceedings of the 8th International Conference on Autonomic and Ttrusted Computing*, pp. 228-242, 2011.

[6] G. Gu, R. Perdisci, J. Zhang, and W. Lee, "BotMiner: Clustering analysis of network traffic for protocol-and structure-independent botnet detection," in *Proceedings of the 17th USENIX Security Symposium*, pp. 139-154, 2008.

[7] O. R. Jeong, C. Kim, W. Kim, and J. So, "Botnets: threats and responses," *International Journal of Web Information Systems*, vol. 7, no. 1, pp. 6-17, 2011.

[8] J. A. Morales, E. Kartaltepe, S. Xu and R. Sandhu, Symptoms-based detection of bot processes, *Computer Network Security*, pp. 229-241, 2010.

[9] J. Nazario, *Blackenergy DDoS Bot Analysis*, Arbor Networks, Technical Report, 2007.

[10] N. S. Raghava, D. Sahgal, and S. Chandna, "Classification of botnet detection Based on botnet architechture," *IEEE International Conference on Communication Systems and Network Technologies*, pp. 569-572, 2012.

[11] M. Sharif, A. Lanzi, J. Giffin, and W. Lee, *Impeding Malware Analysis using Conditional Code Obfuscation*, School of Computer Science, College of Computing, Georgia Institute of Technology, USA.

[12] L. Shuai, G. Xie, and J. Yang, "Characterization of HTTP behavior on access networks in Web 2.0," in *IEEE International Conference on Telecommunications*, pp. 1-6, 2008.

[13] P. Sinha, A. Boukhtouta, V. H. Belarde, and M. Debbabi, "Insights from the analysis of the mariposa botnet," in *Fifth International Conference on Risks and Security of Internet Systems*, pp. 1-9, 2010.

[14] B. Soniya and M. Wiscy, "Detection of TCP SYN scanning using packet counts and neural network," in *IEEE International Conference on Signal Image Technology and Internet Based Systems*, pp. 646-649, 2008.

[15] B. Stone-Gross et al., *Your Botnet is My Botnet: Analysis of a Botnet Takeover*, UCSB Technical Report, 2009.

[16] W. Strayer, D. Lapsley, B. Walsh, and C. Livadas, "Botnet detection based on network behavior," *Advances in Information Security*, vol. 36, pp. 1-24 Springer, 2008.

[17] K. Takemori, M. Nishigaki, T. Takami, and Y. Miyake, "Detection of bot infected PCs using destination-based IP and domain whitelists during a non-operating term," *IEEE Global Telecommunications Conference*, pp. 1-6, 2008.

[18] T. Wang and S. Z. Yu, "Centralized botnet detection by traffic aggregation," in *IEEE International Symposium on Parallel and Distributed Processing with Applications*, pp. 86-93, 2009.

[19] K. Wang, C. Y. Huang, S. J. Lin, and Y. D. Lin, "A fuzzy pattern-based filtering algorithm for botnet detection," *Computer Networks*, vol. 55, no. 15, pp. 3275-3286, 2011.

[20] H. Xiong, P. Malhotra, D. Stefan, C. Wu and D. Yao, "User assisted host-based detection of outbound malware traffic," in *Proceedings of the 11th International Conference on Information and Communications Security*, pp. 293-307, 2009.

[21] W. Yan, Z. Zhang, and N. Ansari, "Revealing Packed Malware," *IEEE Security and Privacy*, vol. 6, no. 5, pp. 65-69, 2008.

[22] L. Zhuang, J. Dunagan, D. R. Simon, H. J. Wang, and J. D. Tygar, "Characterizing botnets from email spam records," in *1st Usenix Workshop on Large-Scale Exploits and Emergent Threats*, pp. 1-9, 2008.

Soniya Balram is a PhD Scholar at the Department of Computer Science, University of Kerala, Trivandrum, India. Her research interests include Intrusion Detection, Port Scan and Botnet Detection, Neural Networks and Fuzzy Systems.

M. Wilscy is a B.Sc (Engg) graduate in Electrical Engineering from TKM College of Engineering, Kerala, India, Master of Engineering (ME) from School of Automation & Computer Science, Indian Institute of Science, Bangalore, India and Ph.D from the Department of Computer Science and Engineering, Indian Institute of Technology, Madras, India. She is currently Professor and HoD, Dept. of Computer Science, University of Kerala, Trivandrum, India. Her areas of research interests are Digital Image Processing, Pattern Recognition, Neural Networks and Fuzzy Systems, and Intelligent systems.