International Journal of Applied Information Systems (IJAIS) – ISSN : 2249-0868
Foundation of Computer Science FCS, New York, USA
Volume 2– No.6, May 2012 – www.ijais.org

Detecting HTTP Botnet using Artificial Immune System (AIS)

Amit Kumar Tyagi
Department of Computer Science,
School of Engineering and Technology,
Pondicherry University, Puducherry-605014, INDIA

Sadique Nayeem
Department of Computer Science
School of Engineering and Technology,
Pondicherry University, Puducherry-605014, INDIA

ABSTRACT
Today's various malicious programs are "installed" on machines all around the world, without any permission of the users, and transform these machines into *Bots*, i.e., hosts completely under to control of the attackers. Botnet is a collection of compromised Internet hosts (thousands of Bots) that have been installed with remote control software developed by malicious users to maximize the profit performing illegal activities like DDoS, Spamming, and Phishing etc attack on online network. Moreover various types of Command and Control(C&C) infrastructure based Botnets are existing today e.g. IRC, P2P, HTTP Botnet. Among these Botnet, HTTP Botnet has been a group of Bots that perform similar communication and malicious activity patterns within the same Botnet through GET and POST message. Generally the evolution to HTTP began with advances in "exploit kits" e.g., Zeus Botnet, SpyEye Botnet, Black Energy Botnet but in later due to *protocol changing; Communication encryption; intermittent communications; Botnet subgrouping Source concealment, and firewall friendly of Botnet*, HTTP Botnet is the most interest of the research community. In this paper, we proposed a new general HTTP Botnet detection framework for real time network using Artificial Immune System (AIS). Generally AIS is a new bio-inspired model which applies to solving various problems in information security; we used this concept in our proposed framework to make it more efficient in detection of HTTP Botnet. Hence finally in this paper, we used AIS to detect effectively malicious activities such as spam and port scanning in Bot infected hosts to detect these malicious exploits kit from a computer system. Our experimental evaluations show that our approach can detects HTTP Botnet activities successfully with high efficiency and low false positive rate.

Keywords
Botnet; Bot; AIS; Spam; Scan; HTTP Bot

1. INTRODUCTION
As the technology growing hackers are also using the latest technology for malicious purposes. Technology changed from traditional virus to Botnet and Root kits. Very recently, Botmasters have begun to exploit cyber crime [7]. Nowadays, the most serious manifestation of advanced malware on cyber security is Botnet to perform attack. "*Cyber Security embraces the protection of Both private and public sector interest in cyber space and their dependency on digital networks and also the protection of exploitation of opportunities – commercial or public policy – that cyber space offers [15]*."
To make distinction between Botnet and other kinds of malware is C&C infrastructure [6, 7]. To understand about Botnet, we have to know two terms first, Bot and Botmaster. The term 'Bot' is nothing but a derived term from "ro-Bot" [40] which is a generic term used to describe a script or sets of scripts designed to perform predefined function in automated fashion. Botnet is a collection of compromised Internet hosts (thousands of Bots) that have been installed with remote control software developed by malicious users to maximize the profit from performing illegal activities on online network. After the Bot code has been installed into the compromised computers, following services are provided by Bots to its Botmaster:

- Robust network connectivity
- Individual encryption and control traffic dispersion
- Limited Botnet exposure by each Bot
- Easy monitoring and recovery by its Botmaster

Generally the terms BotHerder and Botmaster are used interchangeably in this paper [12]. Bots usually distribute themselves across the Internet to perform attack by looking for vulnerable and unprotected computers. When Bots find an unprotected computer, they infect it without the knowledge of users and then send a report to the Botmaster [7]. The Bot stay hidden until they are informed by their Botmaster to perform an attack or any task to harm online users. Other ways in which attackers use to infect a computer in the Internet with Bot includes sending email and using malicious websites. Based on our understanding, we could say that the activities associated with Botnet can be classified into three parts [38, 39]:
1. *Searching* – searching for vulnerable and unprotected Computers.
2. *Distribution* – the Bot code is distributed to the computers (targets), so the targets become Bots.
3. *Sign-on* – the Bots connect to Botmaster and become ready to receive C &C traffic.
Here in this paper we have to propose a framework to detect HTTP Bot infect network. The difference among proposed framework and proposed by Hossein et al. in [7, 41] with using of AIS to detect IRC/P2P Botnet is that our approach does not require prior knowledge of Botnets such as Botnet signature and other details about Botnets. AIS is related to a genetic term, defined as "Adaptive systems, inspired by theoretical immunology and observed immune functions, principals and models, which are applied to problem solving." akin to other bio-inspired model such as genetic algorithms, neural networks, evolutionary algorithms and swarm intelligence [40]. AIS is inspired from human immune system (HIS) which is a system of structures in human body that recognizes the foreign pathogens and cells from human body cells and protect the body against those diseases [8]. Like HIS which protects the human body against the foreign pathogens, the AIS suggest special feature, a multilayered protection structure [13] for protecting the computer networks against the attacks performed on internet. Consequently it has been focused by network security researchers to use and optimize it in IDS. In this work we have

used AIS techniques to detect effectively the malicious activities, such as spamming and port scanning in the detection process of HTTP Bot infected network.

Hence at last, rest of the paper is organized as follows: in section 2, we describe the related work. Section 3 defines about AIS as introduction. Communication mechanism of HTTP Botnet is explained in section 4. Existing solution to detect HTTP Botnet infect network is defined in section 5. In section 6, we describe our proposed detection framework to detect HTTP Botnet and finally conclude this paper in section 7.

2. RELATED WORK

As discussed, Bots and Botnets are hot topics for last few years due to measuring the large number of attacks through cyber crime to rob valuable data of users [6]. So between them HTTP Botnet is a dangerous activity to perform an attack and infected systems behalf of users. Now a day's, HTTP Botnet is one step forward from defenders and use strongest types of methods, technology (strong cryptography) to perform attacks. For e.g. Asprox affect 3.5 billion computers in US only. To make distinction between Botnet and other kinds of malware is C&C infrastructure. So Botnet used to avoid service failure of C&C server when it goes offline to use advanced hydra fast-flux service network to providing high availability of the malicious content e.g. Mariposa Botnet use its own protocol (Iserdo transport protocol and UDP protocol) to perform any activity and use connectionless service. On other side of HTTP Bot's special feature they are firewall friendly, and use a C&C server approach to perform an attack and changing the environment or IP address during attack [15, 21]. Botmaster try to keep their Botnet invisible and portable by using DDNS (dynamic domain name system) which is a resolution service that eases frequent updates and change in server location [11]. Hence to detect this feature based HTTP Botnet, we use this bio spiral approach "AIS" in this paper. On implementation of our proposed approach, our experimental result shows a big difference with other existed techniques to detect HTTP Bot. Our approach provides a significant and reliable result to detect HTTP based Bots with successfully with high efficiency and low false positive rate.

3. ARTIFICIAL IMMUNE SYSTEM

There is a growing interest in developing biologically-inspired solutions to computational problems e.g., the use of AIS that mimic the adaptive response mechanisms of biological immune systems to detect anomalous events [2, 12, and 29]. Here biological immune system's primary components are lymphocyte, which recognizes specific "non-self" antigens that are found on the surface of a pathogen such as a virus [8, 30, and 37]. Once exposed to a pathogen, the immune system creates lymphocytes that recognize cells with the abnormal antigens on their surface. The lymphocytes have the one or more receptors that bind to cells carrying the abnormal antigens. The binding process prevents the abnormal antigens from binding to healthy cells and spreading the virus.

An important feature of a biological immune system is its ability to maintain diversity and generality [1, 2, and 19]. A biological immune system uses several mechanisms to detect a vast number of antigens (foreign nonself cells) using a small number of antibodies [24]. One mechanism is the development of antibodies through random gene selection [24, 34]. However, in this mechanism the new antibody can bind not only to harmful antigens but also to essential self cells [37]. To help prevent serious damage to self cells, the biological immune system employs negative selection, which eliminates immature antibodies that bind to self cells [7]. Only antibodies that do not bind to any self cell are propagated [13]. Negative selection algorithms have proven to be very good at differentiating among self (normal) and non-self (abnormal), and have, therefore, been used to address several anomaly detection problems [12, 18]. A typical negative selection algorithm [8] begins by randomly generating a set of pattern detectors. Here if the pattern matches self samples, it is rejected otherwise it is included in the set of new detectors [17]. This process continues until enough detectors are created. The created detectors are then used to distinguish among self and non-self samples in new data. So by distinguish self and non self samples, we distinguish the behavior of user active on internet as online. Through which we detect HTTP Bot because AIS provides a multilayered protection mechanism to discriminate between the malicious and safe activities.

4. COMMUNICATION MECHANISM

Generally the evolution to HTTP began with advances in "exploit kits" e.g. Zeus Botnet, SpyEye Botnet. These kits, developed mainly by Russian cybercriminals, include Mpack, ICEPack, and Fiesta. . HTTP protocol is widely spread protocol over the Internet and most of the networks allow traffic on port 80[21]. HTTP protocol based Botnet use C&C infrastructure to communicate between Bot and its Botmaster e.g. Asprox Botnet is based on echo based Botnet. In echo based type, Bot announce their existence to the C&C by sending out a full URL (Uniform Resource Locator) to the web server. Generally HTTP protocol ensures existence of the Bot to the C &C server to perform any activity on cyber. The Bot replays the forum.php post data which is partitioned and tracked by GUID (Globally Unique Identifier). In addition, Bot replays the post data for updating new C&C control servers list, new spamming or phishing campaigns related data, new binary version, and new fake AV(Antivirus XP2008) malware. Responses of forum.php contain stolen credentials of the user, Bot's IP addresses, IP addresses of the C&C server, phishing page resources, and injected scripts [10, 39]. The vulnerable computer infected with Asprox binary frequently poll C&C servers via HTTP protocol i.e. Asprox Bot uses port 80 as an outbound ports, HTTP post static boundary ID, version number, and Windows guide [3, 16]. SpyEye Botnet affected a lot of users in 2011.

Actually advantage of using the HTTP protocol is hiding Botnets traffics in normal web traffics, so it can easily bypasses firewalls with port-based filtering mechanisms and avoid IDS detection. Usually firewalls block incoming/outgoing traffic to unwanted ports, which often include the IRC port. There are some known Bots using the HTTP protocol, such as Bobax [35], ClickBot [36] and Rustock [38]. Gu*et. al.* [24] pointed out that the HTTP protocol is in a "pull" style and perform any activity in GET and POST response operation, as shown in Fig. 1.

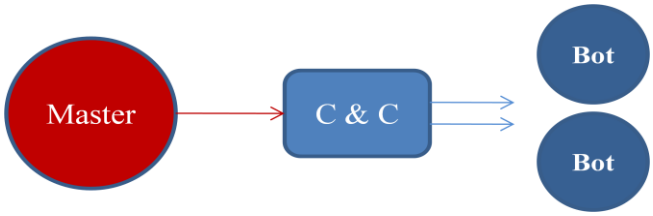

Figure 1: C&C architecture of a Centralized model (e.g. HTTP Bot)

5. LITERATURE SURVEY

Botnet are discussed from last a decade day by day its power and strategies used by the Botmaster is increasing. HTTP Botnet is more dangerous than other existing Botnet (P2P, IRC etc) because of using advanced features to make itself more powerful. This section provides a through survey work of existing Botnets HTTP Botnet. HTTP Botnet can detect:

1. Based on Flow Information[37]
2. Based on Page Access Behavior[27]
3. Based on Degree on Periodic Repeatability[26]
4. Based on Analysis of DNS Traffic[28]
5. Based on Group Correlation Method[22]
6. Through Markovian Protocol Parsing[33]
7. Based on Modeling Connection Behavior[32]
8. Based on Browsing Behavior etc [3]

1. Based on Flow Information

(a) When there is more than one Bot infection machine on the network then DDOS attack is easily possible there [5]. The response to giving an answer or connectivity of client and server may be also infected.

(b) It is a time consuming process to detect spam Bots.

(c) It based on only less no. of infected host that is only for one infected host.

2. Based on Page Access Behavior

(a) Generally in this approach, more dependency on access logs of the past (slop index value) to determine the correlation with browsing time to information size on each page when normal clients browse the web page.

(b) It is not suitable for normal clients.

3. Based on Degree of Periodic Repeatability

(a) We implement this approach only on black energy Bot (an HTTP Botnet) so we cannot say for other HTTP Botnets it will be helpful.

So as a future work we can implement this approach on HTTP Bots and measure the analysis of other types of HTTP Botnet e.g. SpyEye, Zeus, Rusktok etc.

4. Based on Analysis of DNS Traffic

(a) Not efficient to detect HTTP Bots because HTTP Bots are more intelligent and organized for drop off the radar of antiBots.

(b) As we know thousands of the websites are registered everyday as a new service so if an attacker is registered into a white list then after registering, attacker perform the attack on other system, so this type of attack is not possible to detect till now from using this approach.

(c) Not suitable for multi domain Botnets due to changing of name regularly for connecting their C&C server.

(d) It cannot detect bypass attack

Hence as discussed, above approaches have some strength and weakness in it. Now further in long presence of Bot on Internet, few Botnet detection researches also have been proposed by several researchers.

Dagon presents a Botnet detection and response approach [6]. It measures canonical DNS request rate and DNS density comparison of Botnet rallying DNS traffic. However, the approach is inefficient since it generates many false alarms.

Jones [18] describes the Botnet background. He made recommendations so that network and system security administrators can recognize and defend against Botnet activity.

Cooke et al. [4] outline the origins and structure of Botnets. They study the effectiveness of detecting Botnets by directly monitoring IRC communication or other command and control activity.

Barford et al. [1] present an in-depth analysis of Bot software source code. They reveal the complexity of Botnet software, and discuss implications for defense strategies based on the analysis.

BotTracer [21] detects three phases of Botnets with the assistance of virtual machine techniques. Three phases include the automatic startup of a Bot without requiring any user actions, a command and control channel establish with its Botmaster, and local or remote attacks.

Binkley et al. [2] propose an anomaly-based algorithm for detecting IRC-based Botnet meshes. Using an algorithm, which combines an IRC mesh detection component with TCP scan detection, they can detect IRC Botnet channel with high work weight hosts.

Ramachandran et al. [25] develop techniques and heuristics for detecting DNSBL reconnaissance activity, whereby Botmaster perform lookups against the DNSBL to determine whether their spamming Bots have been blacklisted. This approach is derived from an idea that detects DNSBL reconnaissance activity of the Botmaster but it is easy to design evasion strategies.

Zhuang et al. [31] develop techniques to map Botnet membership using traces of spam email. To group Bots into Botnets they look for multiple Bots participating in the same spam email campaign. They apply the technique against a trace of spam email from Hotmail web mail services.

Karasaridis et al. [19] propose an approach using IDS-driven dialog B correlation according to a defined Bot infection dialog model. They combine heuristics that assume the network flow of IRC communication, scanning behavior, and known models of Botnet communication for backbone networks.

BotHunter [12] models the Botnet infection life cycle as sharing common steps: target scanning, infection exploit, binary download and execution, command-and-control channel establishment, and outbound scanning. It then detects Botnets employing IDS-driven dialog correlation according to the Bot infection life-cycle model. Malwares not conforming to this model would seemingly go undetected.

BotSniffer [13] is designed to detect Botnets using either IRC or HTTP protocols. BotSniffer uses a detection method referred to as spatial-temporal correlation. It relies on the assumption that all Botnets, unlike humans, tend to communicate in a highly synchronized fashion. BotSniffer has a similar concept with BotGAD in respect of capturing the synchronized Botnet communication. Different from BotGAD, BotSniffer performs string matching to detect similar responses from Botnets. Botnet can encrypt their communication traffic or inject random noise packets to evade.

BotMiner [11] presents a Botnet detection method which clusters Botnet's communication traffic and activity traffic. Communication traffic flow contains all of the flows over a

given epoch including flows per hour (fph), packets per flow (ppf), bytes per packet (bpp), and bytes per second (bps). The activity traffic identifies hosts which are scanning, spamming, and downloading any Portable Executable binary. Clustering algorithms are applied and performed cross-plane correlation to detect Botnets.
Hence this section some existing approach to detect Botnet specially HTTP Botnet. Now in next section we have to present our proposed framework to detect HTTP Botnet using AIS.

6. PROPOSED BOTNET DETECTION FRAMEWORK

This section proposed a novel approach to detect HTTP Bot. Generally our proposed framework is based on passively monitoring network traffics which can be categorized as IDS approach to detect HTTP Botnet. This framework calculates delay time (td) (which is duration among sending HTTP POST command and HTTP GET RESPONE command) in detection of HTTP Botnet. We share similar idea for definition of Botnet as proposed by guet al. in Botminer.
Hence our proposed approach is a reliable, scalable approach to detect HTTP Botnet detection system, which consist mainly five main components which are: Filtering, Application Classifier, Traffic Monitoring, Malicious Activity Detector Using AIS, Delay Time Analyzer and Analyzer showed in figure -2. Now each term can be defined as:

1. Filtering is responsible to filter out unrelated traffic flows. The main benefit of this stage is reducing the traffic workload and makes application classifier process more efficient [37].
2. Application classifier is responsible for separating IRC and HTTP traffics from the rest of traffics.
3. Malicious activity detector using AIS is responsible to analyze the traffics carefully and try to detect malicious activities that internal host may perform and separate those hosts and send to next stage [8, 18, 19, 20].
4. Traffic monitoring is responsible to detect the group of hosts that have similar behavior and communication pattern by inspecting network traffics [8, 30, and 37].
5. Delay time analyzer is responsible to detect IRC Bots inside the network. Actually analyzer is responsible for comparing the results of previous parts (traffic monitoring and malicious activity detector) and finding hosts that are common on the results of both lists.

Hence in this paper, we have to use AIS model in malicious activity detector (phase 3) to detect malicious activity (HTTP Botnet) after distinguish self and non self samples of a malicious user.

A. Traffic Monitoring

Traffic Monitoring is responsible to detect a group of host that has similar behavior and communication pattern by inspecting network traffics [37]. Consequently, we are capturing network flows and record some special information in each flow. Here we are using Audit Record Generation and Utilization System (ARGUS) which is an open source tool [17, 36] for monitoring flows and record information.
In this tool each flow record has some following information: Source IP(SIP) address, Destination IP(DIP) address, Source Port(SPORT), Destination Port(DPORT), Duration, Protocol, Number of packets (np) and Number of bytes (nb) transferred in Both directions[8, 19]. After determine this information, we insert this information in a data base as shown in Table 1, in which {fi} =1…n is network flows.

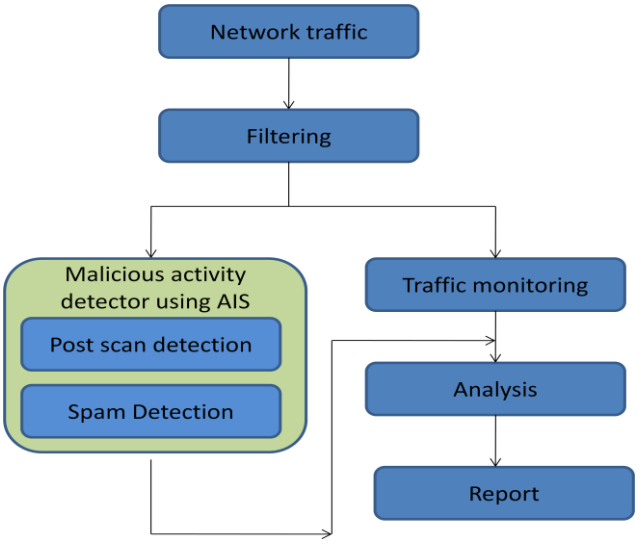

Figure 2: Architecture overview of our proposed detection framework

After this stage, we specify the period of time which is 6 hours and during each 6 hours, all n flows that have same Source IP, Destination IP, Destination port and same protocol (TCP or UDP) are marked and then for each network flow (row) we calculate Average number of bytes per second and Average number of bytes per packet:
• Average number of bytes per second (*nbps*) = Number of bytes/ Duration
• Average number of bytes per packet (*nbpp*) = Number of Bytes/ Number of Packets
Then, we insert these two new values (nbps and nbpp) including SIP and DIP of the flows that have been marked into another database, similar to Table 2. Therefore, during the specified period of time (6 hours), we might have a set of database, {di}i=1…m in which each of these databases has the same SIP, DIP, DPORT and protocol (TCP/UDP). We are focusing just at TCP/UDP protocols in this part [19, 30, and 37].
As we mentioned earlier, the Bots belonging to the same category of Botnet have the same characteristics, similar behavior and communication pattern, also [7]. Therefore, here next step is looking for group of databases that are similar to each other Botnet.
Now further we proposed a simple solution for finding similarities among group of databases. For each database, we can draw a graph in x-y axis, in which x-axis denotes the Average Number of Bytes per Packet (nbpp) and y-axis to Average Number of Byte per Second (nbps) i.e. we can say (X, Y) = (bpp, bps). For example, in database (di), for each row we have nbpp that specify x coordinate and have nbps that determine y-coordinate. Both x-coordinate and y-coordinate determine a point (x,y) on the x-y axis graph.

TABLE 1. RECORDED INFORMATION OF NETWORK FLOWS USING ARGUS

Fi	SIP	DIP	SPORT	DPORT	PROTOCOL	nb	np	Duration
F1								
F2								
F3								
.								
.								
Fn								

We do this procedure for all rows (network flows) of each database. At the end for each database, we have a number of points in the graph; that by connecting those points to each other; we will come up with a curvy graph. Now here our Next step is comparison of different x-y axis graphs and during that period of time (at each 6 hours), those graphs that are similar to each other are clustered (collected) in the same category [19, 30].

The results will show some x-y axis graphs that are similar to each other. Each of these graphs is referring to their corresponding databases in the previous step [2, 37]. We have to take a record of SIP addresses of those hosts and send the list to the next step for analyzing the data.

TABLE 2. : DATABASE FOR ANALOGOUS FLOWS

Source port	Destination port	nbps	nbpp

B. Malicious Activity Detector

In this part we have to analyze the outbound traffic from the network and try to detect the possible malicious activities that the internal machines are performing. Each host may perform different kind of malicious activity e.g. Botmaster perform the following malicious activity like Scanning, Spamming, Binary downloading and exploit attempts. But in this paper, we just focus on scanning and spam-related activities in detection process of HTTP Botnet. In our framework we have utilized the AIS technique to detect these activities. The outputs of this part are the list of hosts which perform malicious activities [12]. Where AIS provides a multilayered protection mechanism to discriminate between the malicious and safe activities. To preserve the generality AIS divides all activities in to self and non-self activities. According to this assumption the self activities are defined as the system's known and normal activities and the non-self are defined as the system's abnormal and harmful activities. However there might be some activities which are unknown to system but harmless. So they considered as self activities in AIS. Hence AIS is used to discriminate between self and non-self activities. The detection procedure in AIS is done in two main steps:

1. Generating and training the detectors: The AIS employs learning techniques using a training data set which contain malicious patterns and system's normal activities to train the detectors against the malicious activities [18, 19, and 23]. The training procedure is result of negative selection algorithm [1, 2, and 19].
2. Monitoring the network activities using the detector sets: After training the detectors, monitor real system's traffic. If any detector is stimulated with any data, the malicious activity will be reported and the detailed information about the activity is stored. In this framework to obtain accurate result we use data capturing technique for a period of time TC to capture normal and abnormal system activities and store them to be used for training the detectors[37]. And the data which is used for spam detection is a set of legitimate and spam message which is delivered in data capturing time.

1) Spam-related Activities: E-mail spam, known as Unsolicited Bulk Email (UBE), junk mail, is the practice of sending unwanted email messages, in large quantities to an indiscriminate set of recipients. A number of famous Botnets that used HTTP as its C&C to perform malicious activity. There are some similarities between SPAM messages and pathogenic microorganisms such as evolution of spam messages by changing the keyword to alternative spelling (e.g. low instead of law) to evade from usual SPAM filters [24]. As well, the similarity between pattern matching techniques used to detect the SPAMs and matching algorithms which are utilized in natural immune system and AIS.

Up to now some immune inspired model had been introduced for SPAM detection [24, 31]. In 2003 by Oda and white [25] uses a regular pattern matching with assigning a weight to each detectors and incrementing and decrementing the weight when it binds with SPAM and legitimate messages respectively. In 2006 Bezerra et al. [29] put forward a new model based on a competitive antibody network by presenting the messages as binary vectors and using a technique for dimensionality reduction. Finally, Guzella et al. [24] in 2008 presented a new effective model with ability to detect more than 98% of SPAM messages. This work which utilized in our framework is named as innate and adaptive immune system (IA-AIS). In this model, negative selection and clonal selection algorithms are combined to train the detectors and detect the SPAM messages [37, 41]. All training data are represented as microorganisms in that the message body and subject are corresponding as antigens and the sender's email address is corresponding as molecular patterns. In generation of microorganisms each character are encoded to a custom 6-bit encoding in order to produce a set of visually similar characters (e.g. 0 and O, 3 and E) to produce different sequences of words which differ in only 1 or 2 bits. By using this technique the detectors with receptor "test" can bind with an antigen "t3st". All detectors are trained using legitimate and SPAM message patterns which are collected during data capturing period.

The process of training is shown in Figure3. Nonself [18] molecular patterns are used for generating the macrophages. Regularity T-cells are generated by randomly selecting from self antigens and B-cells and T-cells selected using same procedure then negative algorithm is applied to eliminate the self reactive detectors. After training the detectors system is ready for detection. Any email is represented as microorganism to the system. At, first sender's email address is presented to macrophage set. Now on activating of any macrophage, email

will be marked as SPAM. Otherwise the message subject and body will be analyzed by B-Cells. If no B-Cell is stimulated, then the message will be marked as legitimate message. Otherwise if any B-Cell stimulated, T-Cells will be used for final investigation and the result will determine whether the message is SPAM or legitimate.

2) Scanning: generally scanning activities may be used for malware propagation and DOS attacks [5]. Most scan detection has been based on detecting N events within a time interval of T seconds. This approach has the problem that once the window size is known, the attackers can easily evade detection by increasing their scanning interval [19, 26]. Now for detecting the port scanning activities performed by Bot infected hosts, Dendritic Cell Algorithm (DCA) is based on artificial immune system which is currently applied for port scanning detection in this paper. DCA presented by Greensmith et al. [31] is another human inspired algorithm.

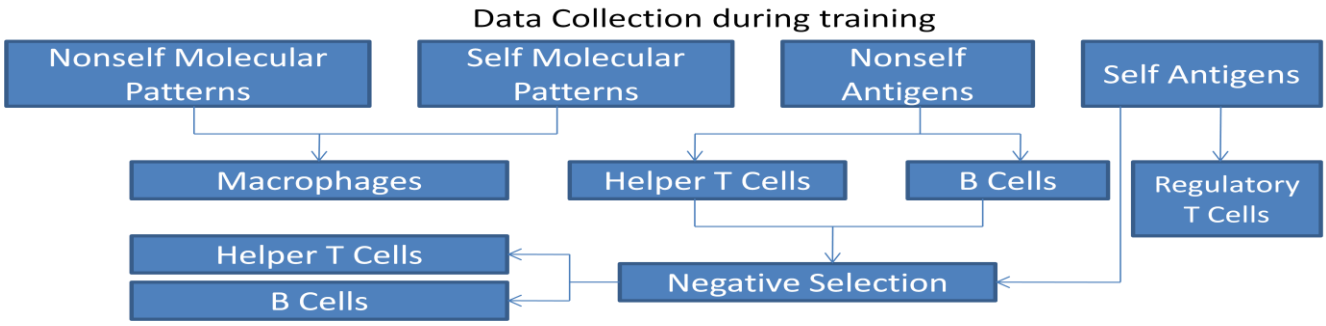

Figure 3: Training Procedure; How to detect malicious host (HTTP Botnet) after port and scan detection in malicious activity detector using combination of T Cells and B Cells.

DCA is based on the new immunological theory called Danger Theory (DT) proposed by Aikelin et al. [34] which is an alternative technique within the AIS. Dendritic cells are a kind of antigen presenting cells within the immune system that is sensitive to changes in signals derived from their surrounding tissues [30, 31]. According to study of Greensmith, DCA is an effective approach for detecting a large scale port scans during an extended time.

In DCA, dendritic cells like a natural data fusion cells process the input signals collected from surrounding environment and produce two kinds of output signals namely IL-12 and IL-10 as response[37]. The algorithm uses two compartments, one for collecting and processing of data called tissue and another for analyzing of data called lymph node. Figure 4 illustrates the DCA framework [2, 30, and 37].

Now used input signals in DCA can categorize in to four categories:

1. PAMPs: It is a signature of abnormal activities e.g., number of errors per second. Presence of this signal usually signifies an anomaly.
2. Danger Signals: It is a measure of abnormality which increases by value e.g. number of network packages per second. Higher value of this signal shows more probability of abnormality.
3. Safe Signals: It is a measure of normal behavior which increases by value e.g. low tolerance of packet sending amount.
4. Inflammation: It is a common system's signal which is inadequate to determine the systems status without presence of other signals. This signal used to magnify the affects of supplementary signals. The input signals are proceed in the system according to a simple weighted sum [37].

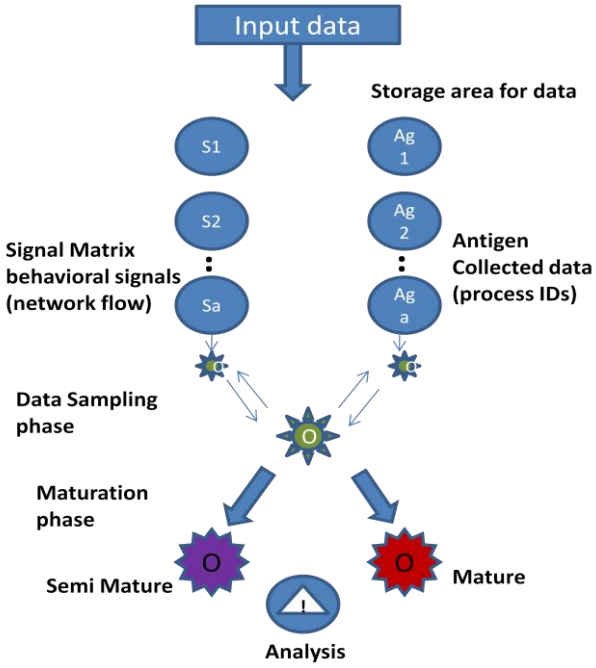

Figure 4: DCA framework

The combination of the input signals lead to either IL-12 or IL-10 signals. The higher value of danger and PAMP signals, the higher mature IL-12 output signals. Reciprocally the higher safe signals, the higher semi mature IL-10 output signals. A schematic diagram of processing the signals in DCA is shown in Figure 5.

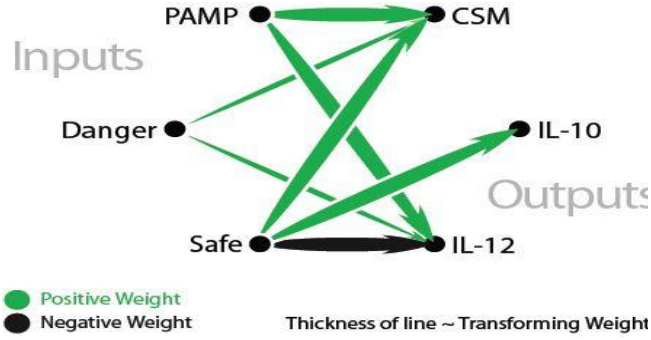

Figure 5: schematic diagram of processing the signals in DCA [30, 37].

7. CONCLUSION

Due to the protocol changing; Communication encryption; intermittent communications; Botnet subgrouping; Source concealment, and firewall friendly of Botnet, today's HTTP Botnet are harder to monitor; shutdown and detection become a challenging problem. So in this paper, we have used AIS for malicious activity detection in HTTP Bot infected hosts. In this paper, we monitor the group of hosts that show similar communication pattern in one step and also performing malicious activities in another step and try to find common hosts in them. Here the point that distinguishes our proposed detection framework from many other similar works are that there is no need for prior knowledge of Botnets such as Botnet signature and other details about Botnets and only requires positive examples, which are readily available before an exploit. At last, experimental evaluations using AIS show that our approach can detects HTTP Botnet activities successfully with high efficiency and low false positive rate.

So in addition, in future we plan to further improve the efficiency of our proposed detection framework is creating and adding unique detection method for centralized part and make it as one general system for detection of all type of Bots (IRC, P2P, UDP, ICMP, FTP FFSN etc protocol based). Hence future research will focus on improving detection accuracy by carrying out a positive selection artificial immune system as well as a fuzzy logic detector.

8. REFERENCES

[1] Zeidanloo, H.R.; BT Manaf, A.; Vahdani, P.; "Botnet Detection Based on Traffic Monitoring", International Conference on Networking and Information Technology, 2010.

[2] Hossein Rouhani Zeidanloo, Azizah BT Abdul Manaf et.al "A proposed framework to detect P2P Bots", IACSIT International Journal of Engineering and Technology, Vol.2, No.2, April 2010 ISSN: 1793-8236.

[3] Wei-Zhou Lu; Shun-Zheng Yu, "An HTTP Flooding Detection Method Based on Browser Behavior", International Conference on Computational Intelligence and Security, 2006.

[4] Steven Gianvecchio, MengjunXie, Zhenyu Wu, and Haining Wang, "Humans and Bots in Internet Chat: Measurement, Analysis, and Automated Classification", IEEE/ACM Transactions on Networking, 2011.

[5] Paul Bächer et.al "Know your Enemy: Tracking Botnets", IEEE, 2005.

[6] Brett Stone-Gross et.al "Your Botnet is My Botnet: Analysis of a Botnet Takeover", 16th ACM conference on Computer and communications security, 2009.

[7] Hossein Rouhani Zeidanloo, Azizah Abdul Manaf, "Botnet Detection by Monitoring Similar Communication Patterns". International Journal of Computer Science and Information Security, Vol. 7, No. 3, Pages 36-45, March 2010, ISSN 1947-5500.

[8] K.W. Yeom, J.H. Park: "An Immune System Inspired Approach of Collaborative Intrusion Detection System Using Mobile Agents in Wireless Ad Hoc Networks", CIS (2) 2005: 204-211 [2005]

[9] Michalis Polychronakis_ Panayiotis Mavrommatis Niels Provos "Ghost turns Zombie:Exploring the Life Cycle of Web-based Malware ",1st Usenix Workshop on Large-Scale Exploits and Emergent Threats, 2008.

[10] Niels Provos, Dean McNamee, Panayiotis Mavrommatis et. al, "The Ghost in the Browser: Analysis of Web-based Malware ",first conference on First Workshop on Hot Topics in Understanding Botnets, 2007.

[11] Keisuke Takemori1, Masakatsu Nishigaki2 et.al, "Detection of Bot Infected PCs Using Destination-based IP and Domain Whitelists during a Non-operating Term", IEEE, 2008

[12] L.N. de Castro, J. Timmis. "Artificial Immune Systems: A New Computational Intelligence Approach" Springer, 2002.

[13] Fu, H., Yuan, X. and Hu, L. "Design of a four-layer model based on danger theory and AIS for IDS", International Conference on Wireless Communications, Networking and Mobile Computing. IEEE, 2007.

[14] Zeidanloo, H.R; Manaf, A.A. "Botnet Command and Control Mechanisms". Second International Conference on Computer and Electrical Engineering, 2009. ICCEE. Page(s):564-568.

[15] Daryl Ashley, "An Algorithm for HTTP Bot Detection", January 12, 2011.

[16] Govil, J.; Jivika, G.; "Criminology of Botnets and theirDetection and Defense Methods", IEEE International Conference on Electro/Information Technology, 2007.

[17] Xiaonan Zang, Athichart Tangpong, George Kesidis and David J. Miller, "Botnet Detection through Fine Flow Classification", CSE Dept Technical Report No. CSE11-001, Jan. 31, 2011.

[18] S. Forrest, A.S. Perelson, L. Allen, R. Cherukuri, "Self–nonself discrimination in a computer", in: Proc. IEEE Symposium on Research Security and Privacy, 1994, pp. 202–212.

[19] S. Hofmeyr, S. Forrest, "Architecture for an artificial immune system", Evolutionary Computation. 2000. 7 (1) 1289–1296.

[20] Gu, G., Perdisci, R., Zhang, J., and Lee, W. (2008). "BotMiner: Clustering analysis of network traffic for protocol- and structure-independent Botnet detection". Proceedings of the 17th USENIX Security Symposium (Security'08).

[21] Tao Wang , Shun-Zheng Yu "Centralized Botnet detection through traffic aggregation", IEEE International Symposium on Parallel and Distributed Processing with Applications, 2009.

[22] Dae-il Jang, Minsoo Kim, Hyun-chul Jung, Bong-Nam "Analysis of HTTP2P Botnet: Case Study Waledac ", IEEE 9th Malaysia International Conference on Communications, 2009.

[23] J.E. Hunt and D.E. Cooke, "Learning Using an Artificial Immune System", Journal of Network and Computer Application, 19, pp. 189-212, 1996.

[24] T.S. Guzellaa, T.A. Mota-Santosb, J.Q. Uchôac, and W.M. Caminhasa, "Identification of SPAM messages using an approach inspired on the immune system" , Science Direct, 2008. 92(3). 215-225.

[25] Oda, T.White, T. "Increasing the accuracy of a SPAM-detecting artificial immune system". In: Proceedings of the IEEE CEC, 2003, vol. 1, pp. 390 396.

[26] Yatagai, T.; Isohara, T.; Sasase, I.; ,"Detection of HTTP-GET flood Attack: Based on Analysis of Page Access Behavior", IEEE Pacific Rim Conference on Communications, Computers and Signal Processing, 2007.

[27] Jehyun Lee, Jonghun Kwon, Hyo-Jeong Shin, Heejo Lee, "Tracking Multiple C&C Botnets by Analyzing DNS Traffic", IEEE, 2010.

[28] Binbin Wang, Zhitang Li , Dong Li ,Feng Liu, Hao Chen, "Modeling Connections Behavior for Web-based Bots Detection", 2nd International Conference on e-Business and Information System Security (EBISS), 2010

[29] Bezerra, G.B., Barra, T.V., Ferreira, H.M., Knidel, H., de Castro, L.N., Zuben, "An immunological filter for SPAM". Lect. Notes Comput. Sci. F.J.V., 2006.

[30] Greensmith, J., Aikelin, U. "Dendritic cells for SYN scan detection". Genetic and Evolutionary Computation Conference. 2007.

[31] J. Greensmith, U. Aickelin, and S. Cayzer." Introducing dendritic cells as a novel immune-inspired algorithm for anomaly detection", In ICARIS-05, LNCS 3627, pages 153–167, 2005.

[32] Hugo F González Robledo, "Types of hosts on a Remote File Inclusion (RFI) Botnet", Electronics RoBotics and Automotive Mechanics Conference, 2008.

[33] Chia-Mei Chen, Ya-Hui Ou, and Yu-Chou Tsai "Web Botnet Detection Based on Flow Information", IEEE, 2010

[34] U Aickelin, P Bentley, S Cayzer, J Kim, and J McLeod. Danger theory:" The link between ais and ids" , In Proc. of the Second Internation Conference on Artificial Immune Systems (ICARIS-03), pages 147–155, 2003.

[35] J. Greensmith, U. Aickelin, and J. Twycross. "Articulation and clarification of the Dendritic Cell Algorithm", In ICARIS-06, LNCS 4163, pages 404–417, 2006.

[36] Argus (Audit Record Generation and Utilization System, HTTP://www.qosient.com/argus)

[37] Hossein Rouhani Zeidanloo "New Approach for Detection of IRC and P2P Botnets" International Journal of Computer and Electrical Engineering, Vol.2, No.6, December, 2010.

[38] Collins, M., Shimeall, T., Faber, S., Janies, J., Weaver, R., Shon, M. D., and Kadane, J., "Using uncleanliness to predict future Botnet addresses," in Proceedings of ACM/USENIX Internet Measurement Conference (IMC'07), 2007.

[39] ZHUGE, J., HOLZ, T., HAN, X., GUO, J., and ZOU, W., "Characterizing the IRC based Botnet phenomenon." Peking University& University ofMannheim Technical Report, 2007.

[40] J. Timmisa, A. Honec, T. Stibord and E. Clarka, "Theoretical advances in artificial immune systems". In: Theoretical Computer Science, Science Direct, 2008. 403(1): 11-32.

[41] Hossein Rouhani Zeidanloo and Azizah Abdul Manaf, "Botnet Detection Based on Passive Network Traffic Monitoring". International Conference on Computer Communication and Network, Orlando, FL, USA, July 2010.

[42] Hossein Rouhani Zeidanloo, et.al "A Taxonomy of Botnet Detection Techniques". International Conference on the 3nd IEEE International Conference on Computer Science and Information Technology. Chengdu, China, July 2010.

9. AUTHOR'S PROFILE

Amit Kumar Tyagi received his Bachelor of Technology degree in Computer Science and Engineering from Uttar Pradesh Technical University, Lucknow, India, in 2009. He is currently pursuing his Master's degree in Computer Science and Engineering in Department of Computer Science from the School of Engineering and Technology, Pondicherry University, India. His research interests include Network Security, Cyber Security, Denial-of Service resilient protocol design, VoIP, Mobile Computing, Cloud Computing, Smart and Secure Computing and Data Mining.

Sadique Nayeem has received his Bachelor of Technology Degree in Computer Sc. & Engineering from Biju Patnaik University of Technology *(*BPUT*)*, Rourkela, India in 2009. He is currently pursuing his Master of Technology degree in Computer Science and Engineering in Department of Computer Science from the School of Engineering and Technology, Pondicherry University, Puducherry, India. . His research interests include Cloud Computing, Holographic Memory and Data Mining.