This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/COMST.2015.2459015, IEEE Communications Surveys & Tutorials

A Comprehensive Study of Email Spam Botnet Detection

Wazir Zada Khan, Muhammad Khurram Khan, Fahad Bin Muhaya, Muhammad Y Aalsalem and Han-Chieh Chao

Abstract—**The problem of Email spam has grown significantly over the past few years. It is not just a nuisance for users but also it is damaging for those who fall for scams and other attacks. This is due to the complexity intensification of Email spamming techniques which are advancing from traditional spamming (direct spamming) techniques to a more scalable, elusive and indirect approach of botnets for distributing Email spam messages. In this paper we first discuss the sources and architectures used by the spamming botnets for sending massive amount of email spam. Then we present detailed chronicles of spamming botnets which systematically describes the timeline of events and notable occurrences in the advancement of these spamming botnets. This paper also aims to represent a comprehensive analysis of particular Email spamming botnet detection techniques proposed in the literature. We attempt to categorize them according to both their nature of defense and method of detection, also revealing and comparing their advantages and disadvantages extensively. We also present a qualitative analysis of these techniques. Finally we summarize the future trends and challenges in detecting email spamming botnets.**

***Index Terms*—Email, Botnet, Spam Email, Email spam botnet detection.**

I. INTRODCTION

SPAM is information which is conveyed or distributed to a large number of recipients without informing them. Spam can be categorized into a wide variety of classes like w*eb spam, Voice over Internet Protocol (VOIP) spam, mobile phone messaging spam, social networking spam, instant messaging (IM) spam, Usenet newsgroup spam* but Email spam is the most recognized form of spam and also the focus of this paper.

Today Email has irreversibly and deeply entrenched in our society as most of the research efforts have been made for making Email technology more convenient, intuitive to use and costing virtually nothing. Thus, an Email system has become an important and essential communication approach for millions of people since one can conveniently transfer messages electronically to anyone within seconds at visibly zero cost [1].

In order to deal with Email, users have to use a mail client to access a mail server. The mail client and mail server use a variety of protocols for exchanging information with each other [2]. The users can access Email in several ways, but most popular ones are for instance Post Office Protocol (POP), Interactive Mail Access Protocol (IMAP) and Webmail. POP version 3 (POP3) is the current standard and it is documented in RFC 1939 [3]. POP is designed to support offline mail processing and works best for those who use single computers all the time. With POP protocol, messages are delivered to the mailboxes and users can access their mailboxes and download messages from the mail server to their computers by using mail client programs (Eudora, Outlook etc.). Once the messages are delivered to the computer the messages are deleted from the mail server. It offers several advantages; first, the internet connection can be disconnected once Email messages are downloaded and one can access (read) Emails in the spare time as it is not stored on the server and thus available when one is offline, incurring no further communication costs. Second, POP frees server disk space since once the Emails and attachments are downloaded they are deleted from the server. Thus, the disk usage on the server is less [2]. Third, POP is supported by any Email client (software). On the other hand its main disadvantage is that while downloading Emails, a lot of messages (including spam or viruses) may be transferred in which one may not be interested, making it much harder to do server-side filtering. Also, it can be much slower to check mail. Moreover, the Email messages will not be accessible from the machine different from that from where the messages were downloaded to.

IMAP is the more complex and recent development which is designed for the users to stay connected to one or more Email servers while reading creating and organizing messages. IMAP4 is the current implementation of IMAP and is documented in RFC 2060 [6]. With the IMAP protocol, Email is delivered to the server and Email messages can be read by connecting to the server. The Email is not stored on the

Manuscript received 27 May 2014; revised 02 December 2014 and 22 April 2015; accepted 23 Jun 2015. This Project was funded by the National Plan for Science, Technology and Innovation (MAARIFAH), King Abdul-Aziz City for Science and Technology, Kingdom of Saudi Arabia, Award Number 09-INF883-02.

W Z Khan and Muhammad Y Aalsalem are with Faculty of Computer Science and Information System, Jazan University, Kingdom of Saudi Arabia (email: wazirzadakhan@jazanu.edu.sa; aalsalem.m@jazanu.edu.sa).

Muhammad Khurram Khan is with Center of Excellence in Information Assurance, King Saud University, Riyadh, Kingdom of Saudi Arabia (email: mkhurram@ksu.edu.sa).

Fahad Bin Muhaya is with King Saud University, Riyadh, Kingdom of Saudi Arabia (email: fmuhaya@ksu.edu.sa).

Han-Chieh Cha is with Department of Computer Science & Information Engineering, National Ilan University, I-Lan, Taiwan and the School of information Science and Engineering, Fujian University of Technology, Fuzhou, China (email: hcc@niu.edu.tw).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/COMST.2015.2459015, IEEE Communications Surveys & Tutorials

computer but is held for users on their server. Its only disadvantage is that when one is offline, an Email is not usually available. On the other hand it offers many advantages; first, Email can be accessible from any machine which the one is using, checking Email either from someone else's computer or a public terminal, without requiring a mail client installed. Second, there is no need to set up manual filters but "server side" filtering is done by the server. Webmail offers same advantages as IMAP but it is often preferred due to its ease of use and the fact that it offers complete access to one's Email without any Email being downloaded to one's computer. Email can be accessed with one's web browser however, webmail depends on a web browser (e.g. Firefox, IE, Opera, Chrome, Safari, etc.) which can take some time to load, access the webmail page, login and load the GUI. The Email delivery mechanism is simple as summarized in Figure 1[2].

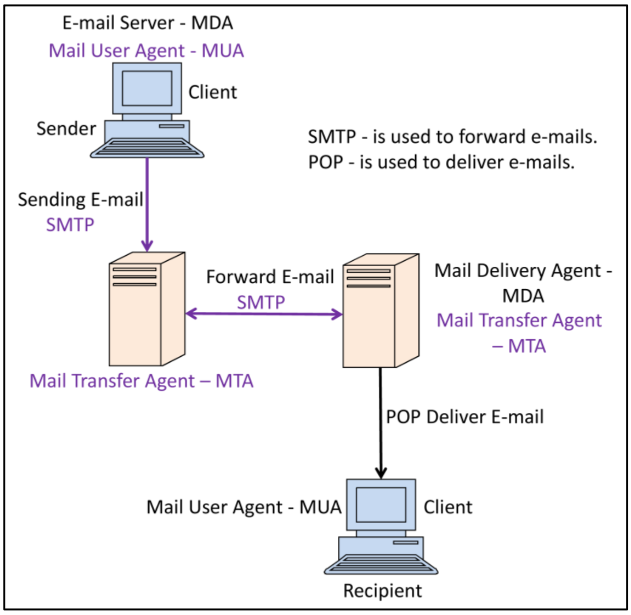

E-mail Server - MDA
Mail User Agent - MUA
Client
Sender
SMTP - is used to forward e-mails.
POP - is used to deliver e-mails.
Sending E-mail
SMTP
Forward E-mail
SMTP
Mail Delivery Agent - MDA
Mail Transfer Agent – MTA
Mail Transfer Agent – MTA
POP Deliver E-mail
Mail User Agent - MUA
Client
Recipient

Figure 1: Email Delivery Process [2]

Figure-1 shows that the actual Email delivery is performed by MDA after accepting a piece of Email from an MTA. The Mail Transfer Agent (MTA) receives Email messages from the Mail User Agent (MUA) or from another MTA on another Email server. The header of a message describes how the message can be forwarded in order to reach its destination. If the mail is sent to such a user whose mailbox is on the local server then the Email is passed to the Mail Delivery Agent (MDA). But when the mail is for a user who is not present on the local server then the MTA forwards the e-mail to the MTA on the appropriate server. The MDA places it into the appropriate users' mailboxes after receiving all the inbound mails from the MTA. The final delivery issues, like virus scanning, spam filtering and return-receipt handlings are resolved by MDA. The Simple Mail Transfer Protocol (SMTP) is used to send and forward Emails from the client (MUA) to the MTA whereas Post Office Protocol (POP) is used to deliver Emails from the Email server (MDA) to the Email client.

Besides its indispensability, Email is prone to misuse in the form of Email spam. An Email spam, also called unsolicited bulk Email (UBE), junk Email or unsolicited commercial Email (UCE), is a pessimistic use of Email as it is an unwanted Email which is indiscriminately sent by a sender who has no relationship with the recipient [4]. The first piece of spam was sent on 3rd May, 1978 to 643 recipients on ARPANET by Gary Thuerk [154]. In recent years, Email system has been misused and abused greatly in the form of unwanted Emails (spam), forged Emails, identity theft and fraud through Phishing Emails, virus and worm attachments, and Email DoS attacks etc. The growing problem of Email spam is mainly due to the lack of authentication of sender and recipient as anyone can send message without any prior approval.

According to the statistics in [7], around 90% of Email messages are spam. Spam is not only irritating and nuisance; it is also a persistent problem which can cause significant harm negatively affecting the internet users and administrators. It has also increasingly become extremely dangerous as 83% of spam contains a URL so phishing sites and Trojan infections are just one click away [8]. Email spam is not only wastage of time but it also consumes storage on the server and blocks communication channels until the recipient takes some action on it. Also there is a chance of deletion of an important Email while deleting spam Emails. Spam Email is also a great malware carrier in order to infect computers with viruses. Chronicle

According to recent reports [40], 83.1% of Email spam was sent through botnets in March 2011 since bots are inexpensive, relatively easy to propagate and very hard to detect, remunerating their botmasters (controllers) financially. Although the subject of botnet and botnet detection has gained a lot of attention and thus, there are several surveys on botnets [12-14][20-31] in the literature but all of them cover the botnet problem in general, discussing different aspects like botnet phenomenon, botnet detection techniques and countermeasures. Also what was missing in these surveys is that all of these studies review only a few of all the research efforts intended only for detecting Email spamming botnets. For example, the survey in [14] comprehensively discussed the botnet problem in general and presents a deeper and wider tutorial like study which summarizes the solution proposals spanning the entire botnet research field. The survey also offers a brief discussion of prominent open research issues. However, the survey only reviews a few of the Email spamming botnet detection solutions categorizing in passive monitoring detection techniques. Moreover, Zhu et al. [19] clarifies certain issues regarding the botnet phenomenon and presents the analysis of Bot types, metrics for determining botnet size and detecting and tracking botnets. In the end some enterprise solution countermeasures against spam are discussed but it presents only a single instance for spam botnet detection and lacks all the other existing detection techniques particularly developed for spam botnet detection.

Similarly the survey in [23] briefly presents certain aspects of botnet research in general including the evolution and future of botnets, propagation speed and mechanisms, population size, data sources available for botnet detection. This survey also categorizes the command and control channels as centralized and distributed and then the research on botnet into two broad categories botnet detection techniques and botnet measurement studies. The section of botnet detection techniques is somewhat limited in scope and briefly reviews only a few of the particularly spam botnet detection techniques.

Lei Zhang et al. [29] present a survey which focuses on advanced botnet mechanisms i.e. Fast flux and Domain flux and comprehensively classify and discuss the latest botnet fluxing features and detection techniques. The survey also compares and evaluates existing botnet fast and domain flux detection techniques against multiple criteria.

The survey in [152] first comprehensively discussed the taxonomy of web spam and then presents a systematic review of web spam detection techniques focusing on the algorithms and principles used for web spam detection. Web spam is relevant to Email spam but web spam phenomenon mainly takes place when web content is generated deliberately for the purpose of triggering unjustifiably favorable relevance or importance of some web page or pages. Web spam is designed to pollute the search engines by driving traffic to particular spammed web pages which corrupts the user experience and degrade the quality of information on World Wide Web [155]. On the other hand, Email spam is an unsolicited or unwanted Email message that is delivered to the recipient indiscriminately, directly or indirectly that contains information such as advertising for a (likely worthless, illegal, or non-existent) product, bait for a fraud scheme, promotion of a cause, or computer malware designed to hijack the recipient's computer. Since, the detection criterion for web spam is substantially different, so, the Email spam coming from botnets cannot be handled by the web spam detection techniques.

The survey in [151] focuses on emerging approaches to spam filtering built on recent developments in computing technologies which are limited in their application to a small scale. But traditional spam filtering approaches can only segregate ham from spam and are unable to detect the spamming botnets as well as the source of Email spam. Comparatively, our survey relies on analyzing Email spamming botnets and their detection mechanisms. The above mentioned concerns have prompted and motivated us to present this comprehensive survey and assemble all the recent research efforts done specifically for the detection of botnet generated Email spam. This paper offers the following contributions:

1) We have explored and discussed the current status of the entire development period in botnet generated Email spam, commencing from the beginning of this gruesome affliction.
2) We have investigated, reviewed and broadly categorized almost all the recent detection mechanisms for botnet generated Email spam on the basis of their level of defense, pointing out their stunning outcomes and noteworthy weaknesses.
3) We have further performed a quantitative analysis of all the existing mechanisms of each category by grasping different aspects of these defensive strategies.
4) Finally we have called the attention to some important challenges in detecting botnet generated Email spam.

The rest of the paper is organized as follows: Section II provides detailed overview of the whole era of spamming botnets. Section III reviews all the recent techniques for the detection of Email spam botnets. Section IV provides a detailed discussion identifying important issues and challenges in detecting botnet generated Email spam and finally Section V concludes the paper.

II. EMAIL SPAMMING BOTNETS

2.1 Sources of Email Spam

To this day Email spamming is still a most popular method that the cybercriminals and the spammers used to exploit for accomplishing various malicious activities. The major sources of Email spam are:

2.1.1 Direct Spamming

In direct spamming, spammers may purchase an upstream connection from spam friendly ISPs or occasionally buy connectivity from those ISPs which do not ignore this activity, forcing them to change ISPs. Thus traditionally, spam was sent by single source mass mailers or by static sources like marketing companies, various e-commerce firms etc [9].

2.1.2 Open Relays and Proxies

An open relay is an SMTP server that accepts relay requests from any source to any destination (also shown in Figure 2) [24].

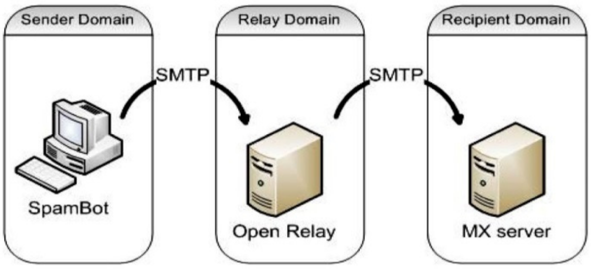
In the past years, open relay was the most common method used by spammers because of the default behavior of any SMTP server i.e. they accept relay requests from unauthenticated hosts to any other host. Nowadays, the mail server's default settings disable open relay requests. Built-in checks for relay requests are developed depending on the network settings. Some ISPs use user authentication for their customers while others accept relays from IP addresses and domains that they trust i.e. white lists. Botnets are now using their own SMTP servers which act as an open relay.

Instead of using misconfigured mail servers, which are rarely found, spambots use other bots who act as an open relay. These bots run SMTP 4 server on high port numbers and they may reside in the same domain or in different domains. Messaging Anti-Abuse Working Group (MAAWG) [19] recommends that this form of relaying spam can be blocked if the spambot's network manages port 25 traffic.

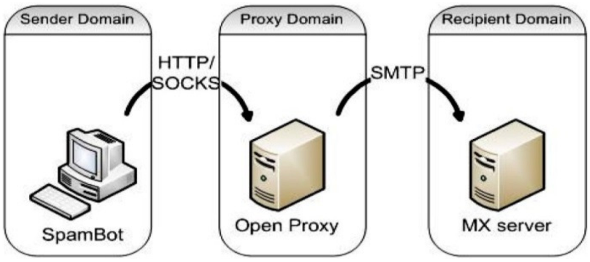
With this practice, all outgoing Email traffic is dropped except from legitimate mail servers. Today, an increasing number of networks are adopting these policies. As a result it has been found that open relays are rarely used in today's spamming campaigns [20].

Open proxy is a proxy server that allows connections from any client to any server on any port. These can be legitimate proxies that have been misconfigured or they can be compromised machines (bots) running a proxy service on a particular port. A proxy server helps the spammer to stay anonymous as it alters the IP address of the source. Open proxies are often used by spambots in order to launder the spam traffic [21]. They receive requests from spambots and forwards spam traffic to the requested mail servers as shown in Figure 2.The origin of the spam Email is thus hidden from the recipient. A study found that the top protocols used for proxying spam are HTTP and SOCKS4/5 [21]. In order for spambots to utilize this service, they need to have IP addresses of open proxies. This can be achieved by either a network scan or by a download from the controlling servers.

A spammer is able to send messages from behind a proxy server via the SMTP protocol directly to a Mail Transfer Agent (MTA) or Mail exchange (MX) server. If the messages are sent to the MX server in the same domain as the proxy server then it is called proxylock. In this case, the proxy server is responsible to look up the MX record of its domain. Proxylock is beneficial for spammers as the message looks more legitimate, because the proxy is in the same domain. Also it requires much less effort to find a relay mail server and the original source is hidden by sending the message in a more distributed fashion. Proxy lock may be disadvantageous since inbound and outbound mail requires deferent processing and thus the mail servers are often separated. The MX record of a particular domain provides the IP address of the inbound mail server of the domain. As a result, spammers are unable to send spam to recipients outside the domain. Rather the spammer can use the SPF record which gives the IP of the authorized outbound mail server, and the bot can obtain the SMTP server settings from the Email client of the infected machine but major ISP policies like user authentication and rate limiting obstruct malicious activities of the spammers.



(a)



(b)

(c)

Figure 2: (a) Open Relays, (b) Open proxies and (c) Proxylock used for Spamming [24]

2.1.3 BGP Spectrum Agility

A recent study shows that spammers may use the technique of BGP spectrum agility [11] by hijacking IP address space via the Border Gateway Protocol from which they send spam. In BGP spectrum agility, spammers briefly (for a short time) hijack large portions of IP address space (that do not belong to them), send spam, and withdraw routes immediately after spamming. This technique is most sophisticated one and it is hard to trace out the spam source. According to [78], less than 10% of today's spam is sent by using this technique.

2.1.4 Botnets

It is often stated that recently spammers have started to use a bonnet infrastructure in order to deliver spam more efficiently. Botnets are the networks that are formed by a collection of compromised hosts called "bots" (derived from the word robot) which are commanded and controlled by a single entity called "Botmaster" [14]. Their goal is to setup a private communication infrastructure for performing a variety of malicious activities [15-16]. Although botnets have different structures they share some similar components (also shown in Figure 3) [17-18]. A bot is a type of malware that uses a software program installed into a compromised computer or victim machines and bots are initialized when the victim boots its computer. A Botmaster sends commands to victim machines for performing specific tasks by using a special command and control (C&C) infrastructure. Figure 3 shows the elements of a typical spamming botnet.

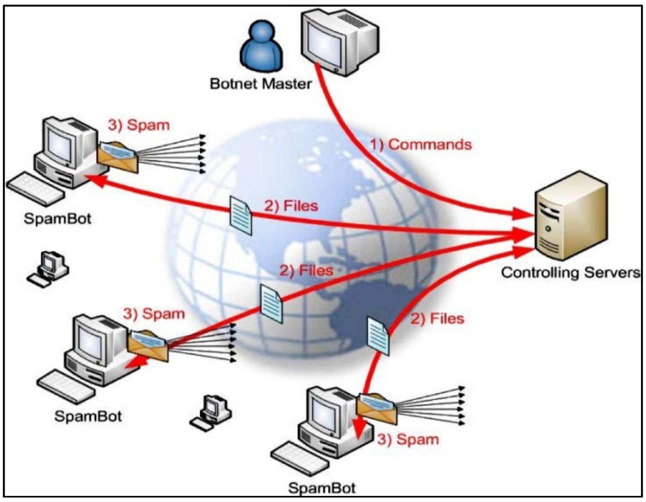


Figure 3: Elements of Spamming Botnet [24]

A typical botnet can be created and maintained by following a typical botnet life-cycle and thus an infected Email spamming host becomes an active host after going through several phases [14]. In the first phase of Initial Infection, a host is infected and becomes a potential bot by using a proper computer infection procedure that can be carried out by using unwanted downloads of malware, infected removable discs, infected Email attachments etc. After the successful completion of the first phase, the second phase called Secondary Injection starts in which an infected host runs a software that searches for malware binaries in a given network data base. After downloading and executing these binaries, hosts behave like real bots or zombies. After this phase bots establish a C&C channel by contacting servers and wait for commands to perform malicious activities. As a last phase of a bot life cycle, botmasters maintain and update the malware in order to keep an army of zombies alive.

2.2 *Botnet Architecture*

An important characteristic of botnets that differentiates them from other malware is the command and control infrastructure used for their communication. There are mainly two types of C&C mechanisms used by botnets namely, centralized and distributed [12].

2.2.1 *Centralized Architectures*

In centralized architectures, bot masters use central servers to issue commands to the selected set of bots. IRC (Internet Relay Chat) and HTTP (Hyper Text Transfer Protocol) are the most common protocols used in centralized architectures. Some of the examples of centralized spamming botnets are Bagle [61], Bobax [87], etc.

In the beginning most botnets use IRC protocol as earliest bots were derived from benign IRC bots. It was attractive because of its several advantages like interactive nature for two way (mlticast) communication between server-client; readily available source code for easy modifications; ability to control multiple botnets using nicknames for bots and password protected channels; and redundancy achieved by linking several servers together, scalability, and versatility that allows code reuse for bots and the creation of new botnets [38].

IRC is a text based instant messaging protocol over the Internet which works on client-server architecture but it is also suitable for distributed environments. Mostly, interconnected IRC servers communicate with each other and each has own subscribers. Thus, when IRC servers are interconnected and are on the same channel, a subscriber on an IRC server may communicate with others. This interconnection between the IRC servers is called multiple IRC (mIRC). IRC-based bots can leverage this infrastructure for malicious activities by managing access lists, moving malicious files, sharing clients, sharing channel information etc. A typical IRC based botnet is shown in Figure 4 [79].

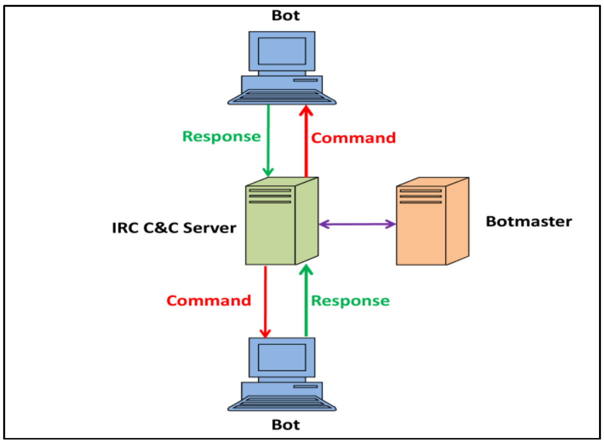


Figure 4: IRC based Botnet Architecture [79]

In IRC botnets, a botmaster creates IRC channels on the command and control (C&C) servers, to which the zombie machines (bots) will connect and wait for commands to carry out malicious tasks. Victim computers are the compromised internet hosts which run the executable bot that is triggered by a specific command from IRC server. Once a bot is installed on a victim host, it will make a copy into a configurable directory and let the malicious program to start with the operating system. The attacker controls the botnet and manages all the bots by setting up a secured channel called control channel. IRC server can be a compromised machine or even a legitimate service provider. Figure 4 shows the attacker which opens a private IRC channel on an ordinary IRC server. After spreading malwares on victim computers an attacker waits bots to subscribe his own private IRC channel. Then he/she gives commands and controls the botnet infrastructures for his/her malicious purposes. Even though IRC protocol is very flexible but it has a serious limitation of single point of failure as it is normally very easy to detect and interrupt the operation of IRC traffic and administrators simply block it with firewalls.

Spammers began to use HTTP protocol for implementing C&C communication due to the restrictions on IRC traffic in corporate networks. The HTTP protocol is used to publish the commands on certain web servers. Botmasters can use HTTP

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/COMST.2015.2459015, IEEE Communications Surveys & Tutorials

protocol to hide their activities among the normal web flows and easily avoid current detection methods like firewalls. It has problem of being central point of failure due to the employment of centralized architecture but it is beneficial for spammers as HTTP traffic is permitted in most networks due to a wide range of HTTP services used. It is not easy to block HTTP service and the detection of HTTP Botnets is even more difficult when the Botmasters use the legitimate websites (e.g. hacked servers) or normal services (e.g. social bots) to establish their command and controls. The HTTP architecture is shown in Figure 5.

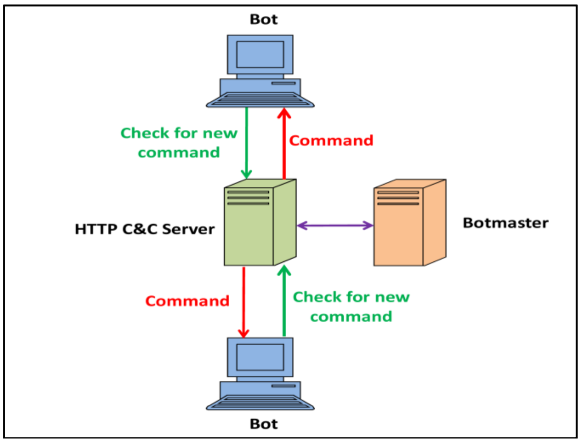


Figure 5: HTTP based Botnet Architecture [79]

2.2.2 *Distributed Architectures*

Distributed architectures, also called decentralized architectures, have no central server and each bot is a peer acting as a server and as a client at the same time. Bots connect with each other when they join the botnet. Distributed architectures are most commonly based on P2P protocols [14]. One of the widespread P2P spamming botnet is the Storm Botnet [10][60][64]. Storm employs a tiered coordination mechanism. In order to exploit remote code execution vulnerabilities in network services it uses autonomous spreading malware to propagate further. If the exploit is successful, the malware transfers a copy of itself to the victim's machine and executes this copy in order to propagate from one machine to another. Figure 6 shows the P2P based botnet.

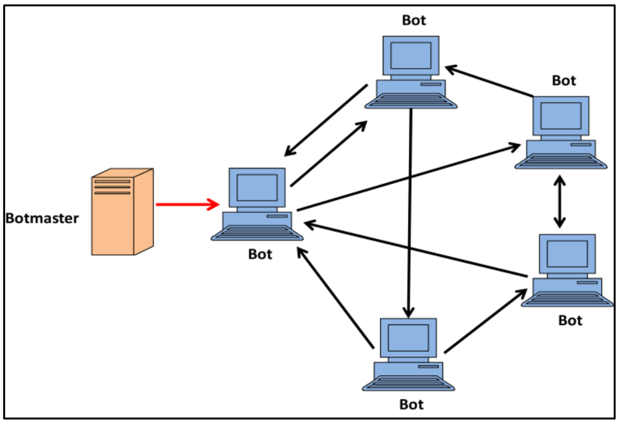


Figure 6: P2P based Botnet Architecture [79]

Due to the issue of central point of failure in IRC and HTTP protocols, spammers began to move towards P2P botnet architecture. In P2P botnet architecture, the botmaster sends a command to one or more bots, and they deliver it to their neighbors. These bots form the backbone structure of the botnet by connecting to each other. Botmaster is able to inject his/her commands into any hosts of the botnet. Each bot transmits the command only to its known directed neighbors. If one bot is detected by intrusion detection systems only its neighbors are affected. It is difficult to detect P2P botnets since any bot in a P2P botnet can publish a command and even if a botmaster is identified and taken down, the P2P botnet will still be functional because any bot can issue botnet commands (i.e. be a botmaster). But P2P architecture suffers from several weaknesses like P2P botnets are slow in convergence, have greater response time, are difficult to manage, and are not scalable. Also the botmaster commands are distributed by other bots, the botmaster is not able to monitor the delivery status of the commands. Moreover, the implementation of a P2P botnet is difficult and complex.

2.2.3 *Hybrid Architectures*

Due to the complexities in handling P2P botnet architectures, the botnet spammers are moving towards hybrid architecture which inherits the properties of both centralized and decentralized/P2P architecture. In hybrid architecture, there exist one or more distributed networks, each with one or more centralized servers. It is advantageous for spammers as if one of these servers is disconnected in the worst case; it will not affect the rest of the architecture, allowing the rest of the botnet to continue its normal operation [96]. A specific hybrid P2P botnet architecture is presented in [120] (also shown in Figure 7) which consists of two types of bots called client bots and servant bots. The client bots behave as clients and as servers in a traditional P2P file sharing network whereas Servant bots are connected to each other and form the backbone structure of the botnet. Botmaster can inject his commands into any hosts of the botnet and can easily manage the entire botnet by issuing a single command. Each bot knows only its directed neighbors and transmits the command to its neighbors. If an intrusion detection system detects one bot only its neighbors are affected. One example of hybrid botnet is Waledac botnet [131-133].

2.3 *Characteristics of Spamming Botnets*

Skilled spammers have started using botnets for sending a huge percentage of spam Emails because of several advantageous characteristics of botnets [67][74].

First, a botnet provides an expedient infrastructure for sending out large volumes of spam Emails since it is an enormous computing distributed network with substantial bandwidth [19]. A botmaster is able to send tens of millions of Emails within a few hours by leveraging thousands of infected machines as it is not hard for spammers to infect machines and then enlist them as new members into a botnet.

Second, as botnets operate in a way that overall tasks are distributed among all the enlisted infected machines thus the amount of required resources for the botmaster is greatly reduced which in turn increases the effective throughput.

Third, the sources or IP addresses of infected machines are constantly changing thus most botnets have a degree of geographic diversity which allows the spammer to evade spam filtering and IP blacklisting techniques no matter how often they are updated.

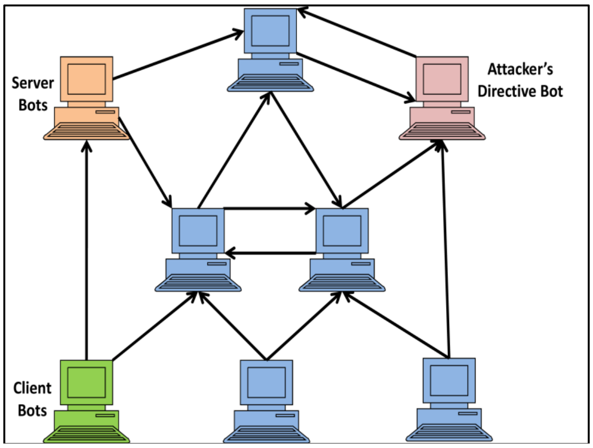


Figure 7: Hybrid P2P based Botnet Architecture [120]

2.4 History of Spamming Botnets

In the past, the notion of botnets didn't incorporate harmful activities and the first known IRC bot called Eggdrop was published in 1993 to assist IRC channel operators [12]. But then the malicious IRC bots appeared which were developed with the intention of attacking other IRC users or entire servers. After that, these bots were used to launch Denial of Service (DoS) and Distributed Denial of Service (DDoS) attacks. In April 1998, the GT-BOT [13][124-125] was the first malicious Bot to be released.

Botnets can be used to generate profits by performing a wide variety of attacks like Distributed Denial of Service (DDoS) attacks, Financial Fraud, Search Engine Optimization (SEO) poisoning, Pay-per-Click (PPC) Fraud, Corporate and Industrial Espionage, Bit coin Mining and Spamming.

Spammers respond to use a strategy called Whack-a-Mole" in which they use a new computer IP address every time and shut down the old one which was first observed in 1996 and then another innovation in spam originated called botnet. The first spamming botnets appeared in 2003 which originate spam Emails from tens of thousands of IP addresses that change frequently. Nowadays, spammers are using botnets for sending large volumes of Email spam, and they also have adopted low and slow spam sending patterns, making all the filtering and Blacklisting methods ineffectual. According to [64], the total daily volume of spam exceeded 120 billion messages per day in year 2008. According to Symantec report of year 2010, more than 89% of all Email messages on the internet were spam Emails, and 88% of these spam messages were sent through botnets [67].

The most popular Email spam sending botnets are Rustock, Cutwail, Storm, Waledac, Bobax, Kraken, Kelihos, Grum, Lethic, Festi, MeagD, Srizbi, Ozdok, Pushdo etc. Table 1 shows the trait analysis of all the Email spam sending botnets. It demonstrates the attributes of all the spamming botnets including their C&C architectures, C&C protocols used, their appearances in the spamming world and the period of their take down operations that are taken against the botnets, the propagation pattern they use for propagating their spamming activities, the infection vector used by them and the spamming campaigns in which the botnets are involved.

Botnet spamming began with the entry of Bagle and Bobax in 2004 which were the first spamming botnets [83]. By the collective efforts of Bagle and Bobax, the cybercriminals started to amplify their spamming activities for attaining more victims' PCs. The initial damage from Bagle.A was first viewed on January 18, 2004 which was not widespread and thus it stopped spreading after January 28, 2004 [84]. A new version of Bagle called Bagle.B appeared and sighted on February 17, 2004 which was much more widespread and appeared in large numbers but it was considered as a "medium" threat by Network Associates as its dominance diminished soon and it stopped spreading spam Emails after February 25, 2004. Bobax was the first template based spamming botnet and was a strong contestant in the Email spamming dome even after its demise was reported in 2007, sending out 9 billion pieces of junk Emails per day, with 185,000 spam-capable bots [85].

Rustock Botnet started operating in early 2006. It was estimated to have 150,000 to 2,400,000 machines capable of sending up to 25,000 spam messages per hour [86]. Also Kraken appeared in year 2006. It was misrepresented as new botnet because it uses the same code by a common author using different C&C domains and fundamentally different C&C protocol [87].

With the scrambling efforts of Bagle, Bobax, Rustock and Kraken, the birth of Storm, Srizbi, Cutwail and Grum spamming botnets was seen in the year 2007. Storm was the most prominent and largest botnet as it was at its height in September 2007, responsible for sending out approximately 20% of spam volume worldwide by using 1 million to 50 million infected PCs. [88] Storm botnet began to decline in late 2007, and by mid-2008, its number of infected computers reduced to about 85,000 computers which was far less than it had infected a year earlier. [89]. The Srizbi botnet estimated to have around 450,000 compromised machines capable of sending around 60 billion spam messages a day [90]. Mega-D botnet appeared in early 2008 and when it went offline in February 2008 for 10 days, the Srizbi botnet heaved out to fill the gap [91]. Thus in mid-2008, Srizbi botnet shows aggressive growth in the amount of Email spam, sending about 60% of all spam in May, 2008 which decreases sharply to 40% of all spam in July, 2008. Cutwail botnet was mostly involved in sending spam Emails around late 2007 and early 2008. In 2008 Kraken was also active sending out 600,000 spam Emails in a 24 hour period with 400,000 to 650,000 bots [92]. It was reported that it was the most difficult spamming botnet to detect in 2008 but it was reported to death in 2009. On November 11, 2008, largest Web hosting provider

McColo was taken down which was hosting C&C servers for Srizbi, Mega-D, Cutwail, Rustock, Bobax and Gheg spamming botnets thus global spam rate was reduced 75% of the but Srizbi botnet was affected badly. Soon after Srizbi disappeared, Xarvester started operating which was very similar in design and operation [93] and controlled about 100,000 hosts. According to MessageLabs Intelligence reports, by the end of 2009, Cutwail was responsible for 17% of all the spam, Bagle about 16%, Bobax returning to its pre-McColo spam levels sent about 13% and Grum botnet about 20%, Mega-D about 58.3% of all spam. Mega-D was destroyed on November 4, 2009 and returned again on November 13, 2009 sending about 4.5% of all spam.

By the end of 2009 the number of bots controlled by Xarvester was also reduced greatly and it faded away by April 2010. Also two newcomers appeared in 2009 which named Festi and Maazben, sending about 3-6% and 2% of all the spam respectively [94] Lethic botnet also appeared in 2009 and it was shut down in 2010 and now it is in decline. Waledac botnet which first appeared in 2008 was shut down in February, 2010 and resurrected again, reaching its highest level in 2012. Meanwhile the Rustcok botnet was up and spamming again in early 2010 with 77% of all spam by using Transport Layer Security (TLS) but in 2011 its spamming rate was below 0.5% of all spam worldwide due to a legal action led by Microsoft [95]. According to [69], Srizbi botnet was up again in 2010 because it implemented a domain generation algorithm (DGA) as a recovery mechanism. Kelihos Botnet which appeared in 2010 was believed to be a version of either Storm or Waledac botnets because of the similarities in the operating system and source code of the bot. It was first taken down in September 2011 by Microsoft. However it reappeared in January 2012 and according to Kasper-sky Lab experts [97], this second Kelihos botnet was built using the same code of the original Kelihos with new features of Bitcoin for mining and wallet theft. After shutting it down on March 19, 2012 by the Kaspersky Lab, the CrowdStrike Intelligence team, Dell Secureworks and the Honeynet Project, another Kelihos botnet version appeared on April 2, 2012. Grum Botnet was taken down on July 19, 2012 by FireEye [99]. According to [100], Cutwail and Festi were the global leaders in sending massive amount of Email spam in year 2012. Some of the most serious uses of the Cutwail spambot (in year 2013-2014) involve the distribution of spam e-mail which helps in spreading the Zeus banking malware [101]. More recently by January 2014 Waledac botnet resurrected by starting a new spamming campaign. Also at this time the Rustock botnet is also observed to be spamming Pharmaceutical Emails again.

III. Email spamming botnet detection

In the past years, spammers sent junk Emails just like regular Emails from their own accounts which were very easy to locate and block the addresses of the sender. It was also easy to trace them back to origin server requesting them to revoke the sender’s account. In order to get rid of this problem the spammers began to relay spam Emails through open relays. The foremost solution provided for countering these spam attacks was to use IP blacklisting which were effective when fixed IP addresses are used by the spam senders.

To combat IP blacklisting techniques, spammers started using botnets for sending massive amount of Email spam messages as these botnets or zombie networks provide them more bandwidth and CPU cycles so they can easily evade IP based blacklisting by using image spam, agile IP address space, BGP route hijacking to steal IP address blocks [47] etc. By exploiting the botnets, the spam Emails are sent at low rate by spammers to any single domain through each IP address. By using this strategy, any single spamming IP address is not identified as suspicious. Thus IP blacklists need to be updated continuously but in actual, blacklisting all IP addresses is not a good solution as it creates annoyance for the mail sever administrators and even fails to detect spamming bots and botnets.

Most of the spam detection is accomplished by using Email spam filters or Email spam filtering techniques [32-37][81] which are considered to be a priority for the network administrators and security researchers as these techniques are easy to deploy and are relatively accurate. But these systems or techniques are capable of only separating or classifying spam (unsolicited Email) and ham (legitimate Email) messages and can neither detect the infected machines (bots) nor trace down the whole botnet or bot family.

It is usually assumed that spam Emails which are identical or similar in context are sent from the same group of spammers or spamming botnet also called spam campaigns. Spam campaigns focus on some meticulous goal which may include selling a product, spreading malware, or committing financial fraud [64]. For each spam campaign, the spammers create tempting message content by mechanically combining a set of subject and body text templates, gather and target a particular set of recipients or addresses and maintain sufficient IP address and message content diversity in ensuring to evade blacklists and textual spam filters. These spamming campaigns may vary in length from only a few hours to as long as months for instance Storm’s e-card campaign of mid-2007 [64]. In this campaign spam messages containing links to view e-cards, or postcards are sent. It carries a simple message, using spoofed source addresses that claims to originate from a friend, colleague or family member.

Table 1: Trait analysis of Email Spamming botnets

Botnets	**First Appeared In**	**Type of C&C Protocol**	**Type of C&C Architecture**	**Take Down Operations**	**Propagation Method**	**Infection Vector**	**Current Status**	**Spam Campaigns**
Bagle [61]	January 2004	SMTP	Centralized	-	Auto-self propagating	to trick recipients into opening the infected attachment, excel file icons were used as attachment icons, other social engineering techniques and P2P files	Inactive	Pharmaceutical, medical ,and replica watches
Bobax [87]	May 2004	HTTP	Centralized	-	-	sending a copy of itself as an attachment to e-mail addresses that it gathers from various locations	Inactive	-
Rustock [134] [135]	Early 2006	IRC + HTTP	Centralized	March. 16, 2011	Auto-self-propagating	drive by exploits, spam Emails, pay-per-install, drive by download	Active	Inflating penny stocks, Selling counterfeit Pfizer pharmaceuticals, Fake Microsoft lottery scams, offers for fake potentially dangerous prescription drugs
Kraken [87]	2006	Encrypted UDP/TCP port 447	-	2009	Auto-self propagating	Spam Emails	Inactive	-
Storm [10][60] [64]	January 2007	P2P	Distributed	-	Auto-self-propagating	e-mails with infected attachments	Inactive	pump-and-dump stock spam, pharmaceutical spam and job-offer (phishing mule) Emails
Srizbi [45]	June 2007	HTTP	Centralized	-	Non-auto-self-propagating	Compromising pornographic websites lead the unsuspecting visitor to websites containing the MPack program. Spam Emails also contain a link pointing to the MPack malware kit	Inactive	spam containing links to fake videos about celebrities.
Cutwail [67] [130]	September 2007	Encrypted HTTP	-	-	Auto-self propagating	regularly distributes malware, sending Emails attachments, usually .zip files	Active	spam promoting pharmaceuticals, designer rip-offs or software, social networking brands
Pushdo [67] [130]	2007	HTTP	Centralized	-	Non-auto-self-propagating	drive by download	Active	-
Grum [149]	October 2007	HTTP	Centralized	July. 19, 2012	Auto-self propagating	infects files referenced by the auto-run registries	Inactive	Fake prescription drugs, pharmaceutical products
MegaD [65]	January 2008	P2P	Distributed	November 2008 November 2009	Non-auto-self-propagating	Drive by download or PPI	Inactive	pharmaceutical products including Herbal King, Express Herbals, and VPXL, male sexual enhancement pill promotions
Maazben [61]	May 2008	SMTP	Centralized	-	Auto-self propagating	Infect files by using spam Emails	-	Casino spam to e-mail addresses belonging to Eastern-European citizens
Waledac [131-133]	October 2008	HTTP + P2P	Centralized & Distributed	February 2010,	Auto-self propagating	Spam Emails pointing to malicious websites, social engineering schemes	Inactive	fraudulent greeting cards and breaking news events
Festi [147]	January 2009	HTTP	Centralized	-	Auto-self propagating	pay-per-Install	Active	spam e-mails lead users back to web sites selling pharmaceutical products primarily male enhancement and watches/jewelry
Lethic [61]	September 2009	IRC	Centralized	Early Jan, 2010 Arose again after take down	Auto-self propagating	-	Inactive	Counterfeit goods pharmaceutical and replica spam
Kelihos [148]	December 2010	P2P	Distributed	Sep, 2011, March 9, 2012	Auto-self propagating	sending malware links to users in order to infect them with a Trojan horse	Active	Online dating

Propagation Method: It defines the techniques for propagation of bots. In Auto self -propagating method, bots automatically propagate them-selves. In non-auto self-propagating method, bots spread themselves with the help of people or other methods. **Infection Vector:** It defines the ways or patterns to infect the victims.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/COMST.2015.2459015, IEEE Communications Surveys & Tutorials

The plain text mails carry a link to a 'greeting' left at a site whose name is selected from a long list of likely titles for such greetings systems, usually also including copyright information matching the selected site title. Following the link leads to one of the same botnet of compromised systems which have been spamming out the mails, hosting a site using exploits to attempt a drive-by download or, if visited by systems lacking the required vulnerabilities, simply presenting a link to download the executable [128]. According to an estimate by Sophos, a global leader in IT security and control, a large volume of 9 million e-greeting card spam mails have been sent across the Internet during the 48 hours over August 14-15, 2007[129]. It is also observed that multiple campaigns use a single botnet or sometimes multiple botnets are involved in a single campaign.

Email spamming botnet detection can be proceeded by targeting any one of the three types of infrastructures namely Botnet detection, bot detection and bot family detection. Botnet detection uncovers all the components of a botnet including the Botmaster, C&C servers and all the bots of the botnet. Bot detection only indicates susceptibility of a host computer or network to botnet infection so that the users and the administrators can take some preventive measures or follow remedial strategies to shun away the infection in future. Bot family detection involves the identification of bot families by analyzing similar behavior, communication or traffic patterns or characteristics.

The researchers and the botnet defenders are constantly trying to keep pace with spamming botnets by mitigating and detecting the harmful intentions of these spamming botnets and consequently coming up with novel solutions. Thus, in this section we provide an extensive review of most of the recent Email spamming botnet detection mechanisms.

We have categorized most of the existing solutions proposed for defending against Email spamming botnets according to their level of activity into two main classes namely Active and Passive techniques. The level of activity measures the degree of intervention of any detection method in the botnet operation. Active techniques are those that are actively involved in the spamming botnet Command and Control (C&C) in order to manipulate network flows for extracting information about possible C&C communication. The main strategy of these techniques is to take part in the spamming botnet operation by masquerading as a component of the botnet and to take down the botnet operation whereas Passive techniques silently observe and analyze the ongoing botnet spamming activities and then make decisions on the basis of this information. Passive detection involves the analysis of network traffic, transaction logs, behavioral characteristics of spamming bots, and other data regarding the Email spam message without influencing or altering spamming botnet operation. A detailed classification chart is shown in Figure 8.

3.1 Passive Techniques for Email Spamming Botnet Detection

Passive detection mechanisms for Email spamming botnets involve silent observation of the spamming activities of the botnets and make important decisions after the in depth analysis of a large collection of Email messages collected at end-user mailboxes. Passive techniques are further categorized into three types; Signature based, DNS based and Anomaly based detection techniques.

Some passive techniques reviewed below can only detect spamming botnet hosts while other can correctly group them into spam campaigns. The major common limitation of these passive approaches is that they can only monitor a small portion of the internet. Also their detection mechanisms are entirely based on the malicious behavior or the type of C&C which is the target of such analysis.

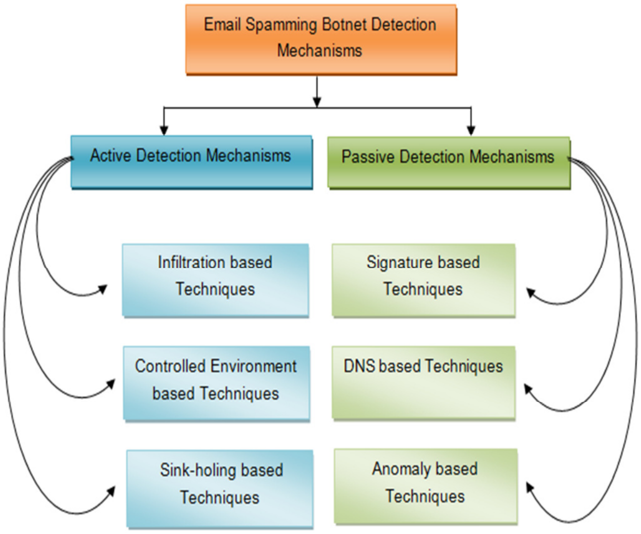

Email Spamming Botnet Detection Mechanisms
Active Detection Mechanisms
Passive Detection Mechanisms
Infiltration based Techniques
Controlled Environment based Techniques
Sink-holing based Techniques
Signature based Techniques
DNS based Techniques
Anomaly based Techniques

Figure 8: Taxonomy of Email Spamming Botnet Detection Mechanisms.

3.1.1 Signature based Techniques

Signature based techniques involve the detection of spamming botnets by leveraging malware executable signatures/rules of known botnets, or first create spam signatures or fingerprints and then search for known malicious patterns in the payload. Signatures have a discriminating power and can make immediate detection by using fewer amounts of resources. Some of the signature based spamming botnet detection techniques are [42-43][45-46].

Xie et al.[42] have developed a framework called AutoRE, which generates spam signatures and characterizes spamming botnets by examining both the content based features in the Email message body i.e. URLs and spam server traffic properties. AutoRE takes a set of unlabeled Email messages (Emails which are not classified as spam or ham) and produces a set of spam URL signatures and a list of related botnet host IP addresses. The spam URL signatures can be either in the form of a complete URL string or a URL regular expression which are used to classify the botnet generated spam Emails. On the other hand, the botnet host identities filter out other spam Emails which can originate from these infected hosts. The authors also look for Email traffic patterns which are exhibited by spamming botnets; first that Emails which originate from botnets hosts are bursty as they are sent in a highly synchronized fashion and second, the botnets hosts are distributed as they usually span a large and dispersed IP address space. By using the three months input data from randomly sampled Hotmail Email messages, a total of 7,721 botnet based spam campaigns have been identified by AutoRE which all together include 580,466 spam messages sent form 340,050 distinct botnet host IP addresses that span 5,916 autonomous systems (ASes). The limitation of AutoRE is that it is unable to generate signatures for the spam messages with image or "tinyurl" for which URLs are not discriminatory.

Zhuang et al. [43] have presented techniques to map and identify botnet membership. The authors have applied shingling algorithm to spam Email messages for efficiently grouping them into spam campaigns. A number of fingerprints are computed for the Email messages which have more than a few common fingerprints either very close in content or identical. A reliable sender IP address is extracted heuristically for each message after the raw-format messages are initially processed. Although a malicious relay can easily alter that IP address so the authors continue to follow the whole chain of relaying IP addresses till they reach the first unrecognized IP address in the Email header. The authors have analyzed the large traces of Hotmail Web mail service collecting about 5 million spam messages over a 9-day period. They have identified 294 botnets and also estimated the botnet sizes by analyzing spam campaign durations and geographic distribution of botnets. The major drawback of the proposed technique is the use of text shingling algorithm which is unable to handle complete image spam. Moreover, the author's assumption that spam hosts participating in the same campaign are part of the same botnets, may not be always true because more than one botnets can be involved in a single spam campaign.

Mori at al. [45] have described a method for detecting the infected hosts of Srizbi botnet by leveraging TCP fingerprinting and TCP/IP stack of the system to identify the operating system of a host. The previously unknown variants of TCP signatures which are also allied to the botnet are extracted to further assist in the detection of bots. After using the Greylisting technique for the classification the spam messages, SMTP logs and tcpdump are correlated together on the basis of their timestamps for associating the spam messages with their respective TCP fingerprints. Authors have also conducted temporal and spatial analysis of spam sending patterns of bots for quantifying the volume of spam sent from the botnet. The authors have conducted an extensive measurement study that spans a large volume of Email delivery logs (SMTP logs) and packet traces (tcpdump) collected from five vantage points for a period of two years (July 2007 to November 2009), estimating the size and spread of Srizbi botnet.

Masaru Takesue [46] has presented a method for detecting spamming botnets by exploiting the information in the hash table of spam Emails. They have applied lightweight clustering in a hierarchical way based on finger prints of message bodies and spam specific words in the Subject headers of the received mail. The clustering of spam is performed in three steps. In the first step, the hash collision in each bucket of the hash table is partially removed. In the second step, the obtained clusters are merged on the basis of the similarity of message bodies of the leaders. In the last step, the second step clusters are further merged using the spam-specific words in the Subject headers of their leaders. After clustering, the IP addresses in the first internal lines of spam Email's Received headers in each cluster are analyzed to identify the botnets and their members. The authors have analyzed the message sources of real world Emails received by the Mozilla mailer on the Linux system.

Jianxing Chen et al. [156] have proposed propose a new method which is based on fuzzy hashing for clustering spam with common goals into the same spam campaign. Fuzzy hashing helps in identifying emails with similar contents even though usual identifiers are obfuscated. The authors have implemented fuzzy hashing to compare the content of spam and compute a similarity score. Spam emails from one campaign have a high similarity score among each other and a low score with other emails. Spam emails are accurately clustered into campaigns by combining fuzzy hash results and the common features, such as URL or email address. The proposed method works in three steps. In the first step the spam emails are clustered. In the second step the characteristics of campaigns are analyzed. And in the last step the source IPs and activity time of bots from large spamming botnets are retrieved by using complete route path of each spam email. The size of botnets is estimated corresponding to the spam campaigns identified in the first two steps. By using fuzzy hashing the computation cost is low because the method relies on a simple hash. In result the authors have extracted two types of spam campaigns from their analysis: template based complicated campaigns and URL based simple campaigns. It was found by their analysis that different campaigns may use the same infrastructure to deliver spam messages.

Son Dinh et al. [157] have proposed a software framework for spam campaign detection, analysis and investigation which identifies spam campaigns on-the-fly and labels and scores the campaigns as well as gathers various information about them. The proposed framework provides law enforcement officials with a powerful platform to conduct investigations on cyber-based criminal activities. The authors have observed that spammers may alter the email subjects in the same campaign, but these subjects still have the same meaning when interpreted manually. The proposed spam campaign detection method and works in the following steps: First the database is carefully chosen so that it is scalable, flexible and able to store relationships between different entities. Then the parser takes spam emails, each of which consists of a header and a body, as its inputs. It then extracts and stores their features in the central database for further analysis. Next in the spam campaign detection phase the frequent-pattern tree FP-Tree is employed which is based on the premise that the more frequent an attribute is, the more it is shared among spam emails. Less frequent attributes usually correspond to the parts that are obfuscated by spammers. Building FP-Tree from email features allows spam emails to be grouped into campaigns. The authors have calculated the text similarity scores of each campaign after identifying spam campaigns using the FP-Tree. Three measurements are applied to generate three similarity scores of all the email pairs in each campaign: w-shingling and the Jaccard coefficient, Context Triggered Piecewise Hashing (CTPH) and Locality-Sensitive Hashing (LSH).

Peter Haider and Tobias Scheffer [160] have studied the problem of identifying botnets and the IP addresses which they comprise by finding a minimal clustering of the graph of messages and is based on the observation of a fraction of the global email spam traffic. The authors have developed an evaluation protocol that is guided by the basic scientific principle that a model has to be able to predict future observable events. Their evaluation metric quantifies the model's ability to predict which email spam campaign a given IP address is going to participate in. Minimal Graph Clustering is proposed for clustering the graph of email messages in which messages are linked if they originate from the same IP address or match the same campaign template. A probabilistic model is devised that directly describes the conditional probability of a clustering given the input graph without making distributional assumptions about the generation of the observable data. The authors have conducted a case study on botnet detection and recorded incoming spam emails over a period of 11 days in January 2012 at a large email service provider. The emails that have been blacklisted on the grounds of three content-based filtering techniques were selected. The authors have compared the proposed Minimal Graph Clustering model to three baselines, threshold-based agglomerative clustering algorithm, spectral clustering algorithm, and generative clustering model for email graphs. From the experiments the authors have concluded that the proposed minimal graph clustering model outperforms a number of reference methods, i.e. spectral clustering, a generative model, and a threshold-based agglomerative clustering model in terms of its area under the ROC curve.

Ching Hao Mao et al. [161] have presented EIGENBOT, a spamming botnet clustering and tracking mechanism that identifies a botnet-based spamming email campaigns. EIGENBOT extracts the key concepts among the spam emails, despite the high dimensionality and the noise in the input and cluster the spamming botnets behavior by leveraging both the semantic intentions of spam and its sources. The authors have proposed a semantic graph analysis approach to differentiate dynamically the different intentions of the sources of the spamming bots. EIGENBOT consists of four components: Spam Syntactic Preprocessing, Semantic Host Graph Construction, Spam Intention Identification, and Latent Spamming Intention Discovery. The spam syntactic preprocessing component is used to eliminate redundant words/symbols, and for transformation and term segmentation purposes. Semantic host graph construction uses a matrix presentation to represent the semantics of a spamming botnet, while spam intention identification is in charge of a number of independent component determinations. Finally, the latent spamming intention discovery, the kernel component extract s the latent behavior. The authors have deployed EIGENBOT to Taiwan network environment for pre-incident protections in information and communication technology center (ICST) and then evaluated EIGENBOT by using real spamming botnet data on the Internet, more than one million spam emails which were collected from Internet service providers (ISPs) in Taiwan. The authors have concluded that their proposed system EIGENBOT is more tolerant to variations, sparseness, and noisy data and has successfully identified spamming botnet groups at a high true positive rate of 82%, thereby improving the detection rate of baseline approaches by 10 absolute percentage points.

3.1.2 DNS based Techniques

A typical bot activity recommences by getting commands and execution parameters of commands from command and control center. Thus bots they need to send DNS queries to know the IP address of the command and control center. DNS based detection techniques try to detect botnet DNS traffic by monitoring the DNS activities and detecting unusual or unexpected DNS querying. A little effort has been devoted to the exploitation of DNS based characteristics of spamming botnets in detecting spam sending bots or their entire botnet family [41][52].

Ramachandran et al. [41] has exposed botnet membership by monitoring lookups to a DNS-based black-hole list (DNSBL). They have performed counter intelligence that helps in discovering bot identities which is based on the perception that reconnaissance lookups are essentially performed by the botmasters in order to determine their bot's blacklist status. They have first applied heurists to make a distinction between legitimate DNSBL traffic and the reconnaissance queries from a botnet. Then after studying the types of DNSBL reconnaissance techniques used by spamming bots, the authors have collected and analyze two datasets from November 2005 to December 31, 2005. First dataset contains the DNSBL query logs to a mirror of a large DNSBL and second dataset contains the logs of bot connections to a sinkhole from a Bobax botnet. They have verified their scheme for discovering additional bots by comparing the IP addresses in the DNSBL query graph against the IP addresses of spammers in a large corpus collected at a spam honey pot in their previous work [78]. The proposed technique, reconnaissance poisoning, misinforms the botmaster about a particular bot which is either clean or listed in the DNSBL. Its main drawback is that it does run the risk of false positives. Moreover, when the botmasters refer to multiple blacklist providers which maintain independent lists, such techniques can be defeated.

K. Ehrlich et al. [52] have presented a method for the detection of Email spam originating hosts, spam bots and their respective controllers by using the network flow data and DNS metadata. The botnet C&C is first identified by analyzing

SMTP flow traffic characteristics for the detection Email spam originating hosts based. If the host traffic profiles of these spam hosts give an indication of their compromise then real-time botnet analysis algorithms are applied for further processing for identifying centralized or distributed botnet controllers. At the initial stage several metrics are applied to flow traffic of hosts to identify potential controllers which work together with these compromised spammers. The flow records of the suspicious spam hosts and the DNS metadata of the suspected controllers are then analyzed by using a DNS passive replication database. Although the distributed controllers of largest spamming botnets like Cutwail, Ozdok, Zeus and Waledac has automatically been detected, but this technique can be evaded by the malware authors e.g. spammers can use legitimate SMTP servers to challenge spam host detection by legalizing reputation and traffic statistics but it requires a considerable struggle for setting and authentication of accounts on legitimate mail services. Moreover, spam messages can be created by the spammers whose size statistics can match a normal Email and they can also alter the protocols and the way they use them in order to prevaricate the controller's detection.

3.1.3 Anomaly based Techniques

In Anomaly based botnet detection, unusual Email traffic patterns that do not conform to the expected normal behavior are observed and analyzed to detect Email spamming botnets. The anomaly based detection techniques detect spamming bot activities based on several network behavior anomalies like unexpected network latencies, network traffic on unusual and unused ports, high volumes of traffic for a mid-class network or unusual system behaviors, anomalies in statistical features of Email spam and spam bots that could indicate the existence of spamming bots in the network. A number of approaches appeared in this context [44][47-54].

P. Sroufe et al. [44] presented a mechanism which detects botnet by analyzing the shape of an Email. They have developed an Email Shape based Botnet Detector called EsBod which inputs spam Emails to a Shape Generator which results in extracting its skeleton. The skeleton is a set of number of character count of each line in the HTML code of the Email. The shape of an Email is derived by employing a form of human intelligence. The Classifier based on Hellinger distance is then used to classify an Email spam to different botnets by feeding the derived Email shape as its input. The performance of EsBod is evaluated by analyzing a corpus of almost 1200 spam Emails with 45 hand labeled botnets. A limitation of EsBod is that it may not be a standalone solution for the detection of Email spamming botnets but it can be combined with network behavior analysis of spam botnets to improve the performance of the detector.

Ramachandran at al. [47] have presented a system called Spam Tracker, which uses behavioral blacklisting for classifying Email senders on the basis of their sending behavior rather than their identity. The authors have designed a behavioral blacklisting algorithm which uses the set of target domains that a particular IP address sends mail to as the primary indicator of its behavior. Spectral clustering algorithm is used to construct the clusters on the basis of sender's Email sending patterns and then each sender is classified on the basis of behavioral similarity to a cluster resembling the known spamming behavior. The authors have analyzed those sending patterns of spammers which differentiate them from those of legitimate senders. They have also observed Email sender behavior across many receiving domains. They have shown that the spam senders who are not persistent in spamming IP addresses and distribute spam evenly across target domains can evade a blacklist or detection entirely. The primary data used for the evaluation of Spam Tracker is a set of Email logs from an organization that hosts and manages mail servers for over 115 domains and also the full blacklist traces from the database of Spamhaus [150] for a period of one month. The experimental results show that Spam Tracker can perform better than the existing filtering mechanisms as it can capture spam which the existing filters do not. But the clustering algorithm used by Spam Tracker has a limitation that if target domains from the same distribution are selected by the spamming bots in a botnet (i.e. the spammers reduce their sending rate or expansiveness of their target list), these spammers will be included in the same cluster. Thus, Spam Tracker cannot detect similar IPs sending spam as the fraction of spam does not change much over short timeframes.

Zhao et al. [48] have designed and implemented a system called BotGraph for detecting a new type of botnet spamming attack named Web-account abuse attack which targets major Web mail providers. The authors have observed that it is difficult to differentiate between a spammer and a legitimate user as bot users send a limited number of Emails and both of them use a same PC. Thus, BotGraph identifies stealthy botnet users by first analyzing correlations among botnets activities and behavioral features (like active time spam contents, Email sending strategies etc) and then constructing large user-user graphs. BotGraph works in two steps, the first step is to detect aggressive signups for restricting the spammer to own a limited number of accounts. A simple Exponentially Weighted Moving Average (EWMA) algorithm is used for the detection of sudden changes in signup activities. In the second step, the login activities of bot users are used to detect the remaining stealthy bot-users. The sharing patterns of single or multiple IP addresses is observed and leveraged to detect bot-user activities. The evaluation is performed using two datasets which include two logs, a Hotmail user-login log, and Hotmail signup log. BotGraph has detected tens of millions of bot-users and millions of botnet IPs when EWMA-based anomaly detection is performed on the signup log and graph based detection is performed on the login log. Although the authors have used their technique to detect only Web-account abuse attack, it can also be used to detect other botnet spamming attacks. The botnet spammers can evade BotGraph detection either by reducing the number of user sign ups of each bot or by impersonating the normal user Email sending behavior of less number of Emails sent per account per day.

Pathak et al. [49] have observed and studied some characteristics of spam campaigns like burstiness and distributedness which are content agnostic features. Burstiness shows the duration of a spam campaign and Distributedness shows extensity or pervasiveness of the sources of a spam campaign. The spam campaigns are manually identified by analyzing content agnostic characteristics of a campaign. Authors have collected a spam trace by using an open relay for a period of five months and they have observed that a few providers e.g. Yahoo, Gmail, Hotmail, Hinet were targeted for a lot of spam. Moreover, the spam sources that are generating bulk of spam Emails used to send spam to almost all the domains. On the other hand, the domains that are receiving high volumes of the spam tend to receive spam Emails from almost all the spamming hosts. Seven major URL-based spam campaigns have been identified containing 2,042 distinct URLs and 2.09 million SMTP connections, then characterizing the duration and distributedness of these URL based spam campaigns which were identified from the relay trace. It is shown that URL based spam campaigns can be prolonged and the burstiness cannot be used as a necessary criteria for clustering URL-based spam campaigns. Finally, the workload distribution and coordination amongst the members of a botnet which instigate a spam campaign are analyzed to study the individual spammer behavior. The bots which are involved in several spam campaigns are deliberated to generate spam Emails for each campaign that is close-by time or have the same workload per campaign, but the recipients for each campaign are distinctive.

Tsigkas et al. [50] have presented a visual analytics tool based on abstract graphs by developing an interactive graph-based visualization method which provides summarized representations of multi-dimensional clusters of security events that are likely to be due to same phenomenon. In order to demonstrate the use of this tool the authors have analyzed the spam Email campaigns in March 2011 when the takedown of "Rustock" botnet took place and provide various visualizations of these spam campaigns. The spam data set was provided by Symantec. Cloud and then all the Email traffic is analyzed by extracting many different features from the Emails including Email headers, message content, sender's IP address, name of the bot, embedded URIs, etc. During the whole year 2011, a total of 3,111,140 spam messages were collected, parsed and stored in the database, whereas, 336,921 messages were collected only for March 2011. They have formed 3-partite graphs by considering the interconnections of spam Emails with the sending botnet and the day when the spam Email was sent showing that while Bagle botnet exhibits a high activity during the whole course of March 2011, there are two botnets (Grum and Xarvester) that exhibit their highest activity just after Rustock was taken down.

Smith et al. [51] have presented a technique in which traffic statistics are used to fingerprint spammers by identifying some unique characteristics of spam sending bots. The distribution skewness is measured for many traffic features by using entropy. These features include Email per recipients, rate of change in recipient list and destination domains, packet inter-departure time and inconsistency in Email header information of the outgoing Email traffic. They have developed a tool that works behind the mail server in a network and captures SMTP data packets and then analyze the traffic while keeping all the personal Email data private. The deviation in these features is measured from the benign Emails for decisively detecting spam-bots.

Ping-Hai Lin et al. [53] have used the Bro intrusion detection system which monitors the SMTP sessions of a university campus and tracks the number and the uniqueness of the recipient's Email addresses in the outgoing mail messages from each individual internal host as the features for the detection of spamming bots. The Bloom filters are used to store and efficiently manage the huge number of Email addresses observed in the SMTP sessions. The underlying principle of this detection method is that the recipients' Email addresses (REAs) from a spamming bot tend to be unique to each other to diversify the recipients of spam messages, while those from a normal user tend to be repetitive because they usually belong to familiar persons. The authors have conducted the statistical analysis of the number and the uniqueness of the REAs from the internal hosts within a period, by deploying the Bro network intrusion detection system which monitored the outgoing SMTP sessions initiated from the internal hosts in the campus of National Chung Cheng University over a period of six months and then they have classified the hosts by the features. After classifying the SMTP activities of a host, the authors judged whether a host is a mail server or not by probing its port 25 or checking in the Bro logs whether its port 25 was passively connected. If it is not a mail server, they proceeded to detect whether it is a spamming bot or not; otherwise, they extracted the IP address of the real source from the Received path in each mail header, and detect whether the source is a spamming bot in the campus. A total of 65 dedicated spamming bots are found by the detection method in the campus, observing 1.5 million outgoing spam messages from them. Also account cracking events are found on 14 legitimate mail servers, on which some user accounts were cracked and abused for spamming. This detection method is dedicated for detecting the spamming bots in the campus not external ones.

Duan et al. [54] have developed an effective spam zombie detection system named SPOT by monitoring outgoing messages of a network. Its design is based on Sequential Probability Ratio Test (SPRT) and can automatically identify a compromised machine quickly. SPOT algorithm works as follows: The message is classified as either spam or ham by the spam filter when SPOT receives an outgoing message and the sending machine's IP address is recorded.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/COMST.2015.2459015, IEEE Communications Surveys & Tutorials

Table: 2 Qualitative Analysis of Passive Detection Mechanisms for Botnet Generated Spam

Passive Techniques	Mode of Operation	Depth of Analysis	Direction of Analysis	Specificit y	Degree of Automation	Location of Deployment	Perspicacity	Detection	Date from Dataset
Signature based Techniques									
AutoRE [42]	Offline	Header	Top down	General	Semi-Automated	Network based	Discriminate	Entire botnet detection	Nov 2006, June 2007 and July 2007
Zhuang et al. [43]	Offline	Deep Packet Inspection	Top down	General	Semi-Automated	Network based	Discriminate	Entire botnet detection	May. 21, 2007 to May. 29, 2007
Mori et al. [45]	Offline	Partial Deep Packet Inspection	Top down	Specific	Semi-Automated	Network based	Discriminate	Entire botnet detection	July 2007 to Nov 2009
Masaru Takesue [46]	Offline	Deep Packet Inspection	Top down	General	Semi-Automated	Network based	Discriminate	Entire botnet detection	Oct. 3, 2008 to Dec. 27, 2008 and May 25, 2009 to Oct. 16, 2009
Jianxing Chen et al. [156]	Offline	Deep Packet Inspection	Top down	General	Semi-Automated	Network based	Discriminate	Entire botnet detection	Jan. 2011 to Dec. 2013
Son Dinh et al. [157]	Offline	Deep Packet Inspection	Top Down	General	Automatic	Network based	Discriminate	Entire botnet detection	Apr. 2012 to March 2013 and Apr. 2013
Peter Haider Et al. [160]	Offline	Deep Packet Inspection	Top down	General	Semi-Automated	Network based	Discriminate	Entire botnet detection	11 days in Jan. 2012
EigenBot [161]	Offline	Deep Packet Inspection	Top down	General	Semi-Automated	Network based	Discriminate	Entire botnet detection	Dec. 2010 to June. 2011 and April 16, 2011 to Dec 15, 2011
DNS based Techniques									
Ramachandran et al. [41]	Offline	Deep Packet Inspection	Top down	General	Automated	Network based	Discriminate	Entire botnet detection	Nov. 17, 2005 to Dec. 31, 2005
K. Ehrlich et al. [52]	Offline	Header	Top down	Specific HTTP, IRC	Automated	Network based	Discriminate	Entire botnet detection	-
Anomaly based Techniques									
EsBod [44]	Offline	Partial Deep Packet Inspection	Top down	General	Manual	Network based	Discriminate	Entire botnet detection	-
Spam Tracker [47]	Offline	Partial Deep Packet Inspection	Bottom up	General	Semi-Automated	Network based	Discriminate	Entire botnet detection	March. 1 2007 to April. 30, 2007
BotGraph [48]	Offline	Header	Bottom up	General	Automated	Host based	Indiscriminate	Single bot Detection	June 2007 and Jan 2008
Pathak et al. [49]	Offline	Header	Bottom up	Specific	Manual	Network based	Discriminate	Entire botnet detection	Sep. 30, 2007 to Feb. 28, 2008
Tsigkas et al. [50]	Offline	Deep Packet Inspection	Bottom up	General	Automated	Network based	Discriminate	Entire botnet detection	March 2011
Smith et al. [51]	Offline	Header	Top down	Specific	Semi-Automated	Host based	Indiscriminate	Single bot detection	June 2007 to June 2008
Ping-Hai Lin et al. [53]	Offline	Header	Diffuse	General	Semi-Automated	Host based	Indiscriminate	Single bot detection	Nov 2011 to April 2012
SPOT [54]	Offline	Partial Deep Packet inspection	Top down	General	Semi-Automated	Host based	Indiscriminate	Single bot detection	Aug. 25, 2005 to Oct. 25, 2005
Carlo Schafer [158]	Offline	Header	Top Down	General	Manual	Network based	Indiscriminate	Entire botnet detection	Dec. 2013 to July. 2015

Mode of Operation: It refers to the method of operation used. In online method, botnets are detected in real time environments when the monitored host or network is carrying out its normal operations whereas in offline method, botnet detection is performed on log files or network traffic dumps. **Depth of Analysis:** It defines the depth of data analysis in detecting botnets. In DPI, detection methods perform fine grained analysis of data. Partial DPI is performed only for suspicious data instead of analyzing the whole data indiscriminately. Header based analysis is performed on flow level data which incurs minimum complexity. **Direction of Analysis:** It defines the sequence of data analysis. In Top-down approach, upstream network components are observed. In Bottom-up approach, the behavior of an individual bot is analyzed. **Degree of Automation:** It defines the degree of human participation in the detection process. Manual approach requires significant human effort to detect botnets. Semi-Automated approach involves little human intervention and an Automated detection requires no human intervention after initial development. **Location of Deployment:** It can host based or network based deployment. In Host based deployment, Host behavior is analyzed which can announce botnet infection only on individual machines or bot. In network based deployment, Network traffic is analyzed and can be deployed anywhere in the network hierarchy. **Specificity:** It defines the detection methods can detect any specific botnet or all the kinds of botnet. Specific detection methods can only detect some specific types of botnet whereas general methods are applicable for detecting any kind of botnets. **Perspicacity:** It defines the ability of detection mechanism to differentiate between different bot families. **Discriminate** mechanisms for botnet detection not only identify the infected machine but also provide information about bot family. Indiscriminate mechanisms can only detect botnet infected machines but are unable to distinguish between different bot families. **Detection:** It defines the entire botnet detection or single bot detection. Entire botnet detection methods can detect the whole family of botnet whereas single bot detection techniques can only detect a single bot or a single infected machine. **Date from Dataset:** It shows the period of collected data for experimental results.

SPOT maintains the logarithm value of the corresponding probability ratio for each observed IP address, whose value is updated when the message arrives from the IP address. The algorithm determines that the corresponding machine is compromised, normal or decision cannot be reached and need additional observations on the basis of calculated values. The authors have also presented two more algorithms, Count Threshold (CT) detection algorithm which counts the number of spam messages sent and Percentage Threshold (PT) detection algorithm which determines the percentage of spam messages sent from an internal machine and compared them with their SPOT algorithm. The evaluation is performed using a two month Email trace collected on a large US campus network. The SPOT algorithm outperforms both CT and PT algorithms and the results show that there were 440 FSU internal IP addresses in the Email trace and SPOT identifies 132 of them to be associated with compromised machines.

Carlo Schafer [158] has presented two methods for detecting compromised accounts called Country Counting and Theoretical Geographical Travelling Speed which do not need access to email content to detect an abused account since these compromised accounts are primarily used by botnets for sending new spam or phishing emails. The proposed method uses data that is extracted from the log files of the SMTP server and the detection is succeeded by observing the incoming metadata from the from the SMTP connections. All this information is stored and enhanced with MaxMind, an IP Geolocator which is the main component of the proposed method. From the available connection information, the source IP and the connection time are extracted to detect an anomaly In order to detect an anomaly of an abused account, the author first differentiates between a normal case (human users and the accounts used by devices like a server or other hardware for sending notifications) and a special case (compromised account used by spam botnets). In the first method of Country Counting (CC), the countries for a specified time interval from where the same accounts are used to connect to the SMTP server are counted and the available metadata is expanded with the geolocations, so as to get information about the country of origin and geographical location of the authentication. The second method Theoretical Geographical Travelling Speed (TGTS) is based on a concept that an email account is primarily used by a human person who can only be in one place at a time and if the TGTS is too high, an anomaly is detected.

The quantitative analysis of all the passive detection techniques is shown in Table 2. It demonstrates various technical aspects of these techniques that express the depth analysis of the strategies followed by these techniques in detecting spamming botnets.

3.2 Active Techniques for Email Spamming Botnet Detection

The more recent adopted approach to study the spamming botnets is to actively participate in the botnet operation by creating a client that mimics' the C&C protocol and after joining the botnet, they can accurately estimate the size of the botnet and can dismantle or take down the entire botnet [55]. Active detection techniques can be further categorized into three types namely, Infiltration based. Controlled Environment based and Sink-holing based. In infiltration based mechanisms, these approaches necessitate the deeper knowledge and understanding of the targeted C&C protocol and the botnets architecture. Thus it requires a priori knowledge about the botnet which makes it difficult to operate on a new and [56]. In Sink-holing (also called BGP black-holing) based techniques, the malicious traffic is re-directed from the spamming botnet to sinkholes which record and analyze the spamming activities. In such approaches the existence of backup channels for C&C processes can be challenging [57]. In controlled environment based techniques, the researchers establish a virtual machine environment or a botnet monitoring platform, execute spam bots or Email clients in virtual machines and after applying their techniques and further continuously monitoring botnet spamming activities they detect origin of spam, identify the active botnets, number of hosts controlled by botnets or group the incoming spam into ongoing spam campaigns.

We have observed a considerable collection of several studies [58-62][63][126] whose goal was to measure and understand the botnet phenomenon but more recently the researchers have either actively infiltrated [39][64-67][82][127] the spamming botnets or relied upon spamming botnet taken down operations [68-71]. Although, the take down operations involve delicacy concerning the effects caused on infected machines, they are enormously thriving equally [72]. Some legal constrains also complicate the take down operations [57]. The quantitative analysis of all the active detection techniques is shown in Table 3.

3.2.1 Infiltration Based

Kreibich et al. [64] have investigated distribution infiltration which is a new approach for measuring spam campaigns esoterically. An initial analysis of Storm botnet spam campaigns is presented by infiltrating the data captured at its distribution platform. The components of the system that are supporting spam campaigns are studied which include a modular campaign framework, a work queue model to distribute the load through the botnet, a template language to introduce per-message polymorphism, delivery feedback for target list pruning and per bot address harvesting for acquiring new targets. The authors have run 16 instances of Storm bots in controlled environments (from late Dec. 2007 through early Feb. 2008) using virtual machines that are hosted on VMware ESX 3 servers for instrumentation and have used crawler for probing Storm proxies and analyze different forms of Storm communication traffic. Depending upon the machine's configuration, these bots may possibly run as either workers or proxies. The outgoing C&C requests from the workers are analyzed to discover their communication with the external proxies. A custom crawler acquires the latest spamming instructions by mimicking the presence of additional workers and using these proxy identities as a feed to enquire each active proxy repetitively. The dictionaries are studied which can be used to construct messages programmatically, the pervasiveness of several categories of campaigns, address harvesting behavior and the existence of test accounts employed by the botmaster. In the end the total size of a campaign mailing list is estimated by using a small sample of the global botnet activity.

Chao et al. [65] have conducted an infiltration of the MegaD botnet for 4 months which continuously provides the data regarding MegaD's complex and its C&C architecture along with its spam operations. The botmasters' operations and malicious activities are analyzed with the collection of ample confirmation that multiple botmasters are managing the MegaD's infrastructure. Two infiltration techniques are used namely Google hacking, which can locate C&C servers when fingerprinting is applied on the web pages that are supplied on the visiting of non-bots, and the milkers are used for extracting the series of commands sent to bots as well as C&C overall structure by mimicking the interactions of a bot's network and probing different C&C components. The milkers analyze the ongoing activities of the botmasters by "milking" MegaD C&C components and its operations. FireEye propelled a harmonized effort to takedown MegaD on Nov. 6 2009 and MegaD trickled to halt after a successful takedown but after 16 days its share of world's spam climbed to 17%. The authors observed an interesting finding about the botmasters take down response that the botmasters didn't have backup domains and ISPs and thus they took weeks to find a new ISP for hosting their infrastructure and setting up the new C&C servers.

Kreibich et al. [66] have presented an infiltration of spamming campaigns of the Storm botnet which is an extension of their previous work [64]. They have analyzed C&C flow of Storm botnet in upward direction by injecting target addresses into the Email address harvests gathered from infected machines and analyze the spam campaigns for a longer period of time. Two platforms are used for conducting the measurements, a C&C crawler which is used for collecting updated messages that contain spamming information by tapping into Storm's network, a C&C rewriting engine for observing the activity of real worker bots of Storm botnet by infiltrating the botnet at the proxy bot level. The authors have also performed harvest injection experiment to confirm that the addition of harvested Email addresses from compromised machines to the spammer's distribution list. Large variances have been observed in size domain distribution and Email address overlap between the target lists of the spam campaigns.

Stone Gross et al. [67] has comprehensively analyzed botmaster's perspective on a large scale by highlighting orchestrating spam campaigns in detail. They have investigated the effectiveness of IP based blacklisting, quality of Email address list, and the reliability of bots. The authors have gained access to 13 C&C servers and 3 development servers of Pushdo/Cutwail botnets in August 2010 and performed an analysis of modus operandi of cyber criminals and lively demonstration of the sophisticated spam operations. The tool ANBIS is used for runtime analysis of binary programs like file system modifications and network activity. The authors have also analyzed an underground web forum called Spamdot.biz which is committed to spam operations.

3.2.2 Controlled Environment Based

Andreas Pitsillidis et al. [73] have developed a system called Botnet Judo which produces regular expression signatures by processing spam generated or sent by bots (individual botnet instances) that are executed in a controlled environment. A black-box approach is used to collect spam messages which are then analyzed. A template inference algorithm is used to generate a matching regular expression signature and the set of signatures are updated immediately on the arrival of new messages as the system operates in real time. For experiments the authors have used real templates and dictionaries collected during their 2008 study of Storm botnet campaign orchestration [125] covering three campaigns including a self- propagation campaign, a pharmaceutical campaign and several low priced stock campaigns. After evaluating a case of spam generated using a single template, the authors have also analyzed multiple template inference. The real time experiments confirmed that Judo works well in both cases. Besides, spammers can use technical means and more complex spam generation languages to complicate the execution of bots within controlled environments and in turn perplexes the template inference approach. Also the spammers can manage the distribution of templates in a more adversarial fashion to make this system unable to generalize to spam issued from another bot in the same botnet.

G. Stringhini et al. [74] have presented a technique for identifying and tracking bots that send spam. This approach can track online IP addresses of all active hosts that belong to every spamming botnet, participating in spam campaigns. In the first step the transaction logs are searched for those senders IP addresses entries which are one of the IP addresses in the seed pool i.e. the known spambots. After analyzing them a number of behavioral profiles are generated. In the second step, the whole transaction logs are searched for similar behavioral patterns which are similar to the spambot behavior that is previously learned from the seed pools, and the hosts which behave in a similar manner are flagged as possible spamming bots. The IP addresses of these hosts are added to the corresponding magnified pool. The third step involves applying heuristics to reduce the false positives and assigning the spam campaigns to specific botnets. The technique is implemented in a tool called BotMagnifier in which each seed and a magnified pool are associated with a label for identifying the name of the botnet that is carrying out the parallel spam campaign. The authors have built an environment and execute bot binaries in a controlled setup and the system is run on a large set of real world data for a period of four months to successfully track the growth of large botnets.

John P. John et al. [75] have designed a platform called BotLab which monitors a botnet and after analyzing all incoming spam, arriving at the University of Washington, it provides real time information about the botnet activity. After precisely observing the outgoing spam feeds, multiple captive sandboxed nodes which belong to different botnets are then executed. BotLab has analyzed botnet behavior by using URLs that are found in its incoming spam feed and the outgoing spam sources. It also tracks down that how the spam is coming into and going out from these ongoing spam campaigns of botnets by determining the number of active hosts within each botnet. BotLab provides updated information regarding the spamming botnets which includes information about the currently ongoing spam campaigns of these botnets, their bots and C&C servers. For identifying the spamming bots, a network fingerprint is produced for each binary that BotLab considers, by obtaining more reliable behavioral signatures. These signatures seizure the information regarding binary initiated network connections. Network fingerprints are comprised of a set of flow records which are then analyzed. After examining the actions of the bots running in the BotLab and analyzing the properties of the outgoing spam feed produced by these bots, they are characterized on the basis of their behavior. Correlation is performed on the incoming and outgoing spam feeds to first classify spam based on botnet source and then identify their spam campaigns. The botnet partitioning is analyzed to estimate the botnet sizes and botnet membership lists are produced after analyzing and enlisting new spam victims.

Jan Gobal et al.[98] have introduced a proactive approach for directly and efficiently collecting spam messages after the interaction with the controllers of spam botnets. The authors have executed spambots (malicious software responsible to send Email spam) in a controlled environment and collected all the Emails sent by those bots. All the spam mails are collected at the gateway and a machine is reset to a clean system by using some software based restoring mechanism after some definite time and then the next spambot is executed for further collecting spam Emails. On the basis of the collected information, the detection rules are generated and a model is created to define the spam Email's overall structure. All the incoming messages are then checked against this model and if matches exist then it is classified as spam message. In order to evaluate the template generation process, Emails from a Casino advertising spam campaign collected during June and November 2008 are used. It is shown from the results that a good template can be formed by using a large number of Emails that depending upon the spam campaign diversity and the time span for which the template is used without update.

Gianluca Stringhini et al. [76] have presented a system for filtering spam by observing and analyzing how messages are sent by the spammers, precisely focusing on the Email delivery mechanism. Two complementary techniques are proposed namely, SMTP dialects and Server feedback manipulation. The concept of SMTP dialects is based on the observation that different Email clients (bots) implement the SMTP protocol in slightly different ways. Dialects, capturing these small variations, help in distinguishing legitimate Email clients and spam bots. The technique is implemented to automatically learn the SMTP dialects of both legitimate clients and spam bots. The Server feedback manipulation is based on the observation that the botmasters take into account the server feedback to improve the performance of a spam campaign. The spammers remove the non-existent recipient addresses from their Email lists which prevent them from sending useless messages during subsequent campaigns. By exploiting this, authors have manipulated the responses from the mail server to a bot by providing incorrect feedback to bots which negatively affects the spamming effectiveness of a botnet. The algorithm uses both passive and active probing to efficiently generate models to distinguish between different Email engines (botmasters). This approach is implemented in a tool called B@bel which is able to correctly identify spam bots in a real world scenario by running Email clients in virtual machines and applied the learning techniques to learn SMTP dialects of each client. The learned dialects are then used to build a decision machine which performed malware classification and spam mitigation. The results demonstrate that B@bel is successful in detecting current spambots but the bot authors can evade dialects detection by using an existing SMTP engine which is used by a legitimate client or adopting a well-known SMTP implementation for their bots. Spammers can also evade server feedback manipulation technique by guessing whether the receiving mail server is performing feedback manipulation.

Gianluca Stringhini et al. [159] have presented an analysis of the relations among email harvesters, botmasters, and spammers. The authors set up a large number of email addresses, each pointing to a mail-server under their control and advertise them on web pages. Then, they recorded the accesses to those web pages, to fingerprint the email harvesters. Then, they log the connections that they received on their mail-server. Since the email addresses that they disseminated on the web are not used for legitimate purposes, the authors have assumed that any connection that they received was generated by a botnet (or by a mail-server operated by spammers). After that the authors applied a technique known as SMTP dialects [76] to assess which botnet or mail-server generated each connection. As a last step, the authors have analyzed the content of the spam emails that they received, and group them into spam campaigns. The different datasets are compared in order to check whether the same spammer has rented multiple botnets, and whether multiple spammers share the same email list or botnet. From the experiments the authors have observed that the main activity concentrates on a small set of IPs and particularly four IP addresses harvested 70% of the email addresses, which ended up receiving 74% of the total spam. The proposed system identified seven different dialects among the clients that sent emails to the mail-server which was targeted by three of the largest active spamming botnets (Cutwail, Lethic, and Kelihos). After applying clustering technique to the emails the authors received a total of 63 spam campaigns. The authors have observed that all the botnets were used by a single spammer each.

Table 3: Qualitative Analysis of Active Detection Mechanisms for Botnet Generated Spam

Active Techniques	Mode of Operation	Depth of Analysis	Direction of Analysis	Specificity	Degree of Automation	Location of Deployment	Perspicacity	Detection	Date From Dataset
Infiltration based Techniques									
Kreibich et al. [64]	Online	Deep Packet Inspection	Diffuse	Specific (Storm)	Manual	Network Based	Discriminate	Bot detection	Dec. 26, 2007 to Feb. 4, 2008
Chao et al. [65]	Online	Deep Packet Inspection	Diffuse	Specific (Mega-D)	Manual	Network Based	Discriminate	Bot detection	Oct. 27, 2009 to Feb. 18, 2010
Kreibich et al. [66]	Online	Deep Packet Inspection	Diffuse	Specific (Storm)	Manual	Network based	Discriminate	C&C detection	Apr. 26, 2008 to May. 6, 2008
Stone Gross et al. [67]	Online	Deep Packet Inspection	Diffuse	Specific (Pushdo/Cutwail)	Manual	Network based	Discriminate	Bot detection	Aug 2010
Sink-holing based Techniques									
Ramachandran et al.[78]	Online + Offline	Partial Deep Packet Inspection	Bottom Up	General	Semi-Automated	Network based	Discriminate	Entire botnet detection	Aug 2004 to Dec 2005
Shin et. al. [77]	Cross analysis of two existing Techniques								
Controlled Environment based Techniques									
Bot Magnifier [74]	Online + Offline	Partial Deep Packet Inspection	Bottom up	General	Manual	Network based	Discriminate	Entire botnet detection	Sep. 1, 2010 to Feb.10, 2011
BotLab [75]	Online	Deep Packet Inspection	Bottom up	Specific	Semi-Automated	Network based	Discriminate	Entire botnet detection	March 2008 to April 2008
Botnet Judo [73]	Online + Offline	Header	Top Down	General	Manual	Network based	Discriminate	Entire botnet detection	Feb 2003 to Feb 2007 and Aug 2000 to April 2009
Gobel et al. [98]	Online	Deep Packet Inspection	Top down	General	Automated	Host based	Indiscriminate	Single bot detection	June 2008 to April 2009
B@bel [76]	Online	Deep Packet Inspection	Bottom Up	General	Semi-Automated	Network based	Discriminate	Entire botnet detection	June. 18, 2011 to Aug. 30, 2011
Stringihini et al. [159]	Online + Offline	Deep Packet Inspection	Bottom Up	Specific (Lethic, Cutwail, Kelihos)	Semi-Automated	Network based	Discriminate	Entire botnet detection	Dec. 14, 2013 To May. 15, 2013

Mode of Operation: It refers to the method of operation used. In online method, botnets are detected in real time environments when the monitored host or network is carrying out its normal operations whereas in offline method, botnet detection is performed on log files or network traffic dumps. **Depth of Analysis:** It defines the depth of data analysis in detecting botnets. In DPI, detection methods perform fine grained analysis of data. Partial DPI is performed only for suspicious data instead of analyzing the whole data indiscriminately. Header based analysis is performed on flow level data which incurs minimum complexity. **Direction of Analysis:** It defines the sequence of data analysis. In Top-down approach, upstream network components are observed. In Bottom-up approach, the behavior of an individual bot is analyzed. **Degree of Automation:** It defines the degree of human participation in the detection process. Manual approach requires significant human effort to detect botnets. Semi-Automated approach involves little human intervention and an Automated detection requires no human intervention after initial development. **Location of Deployment:** It can host based or network based deployment. In Host based deployment, Host behavior is analyzed which can announce botnet infection only on individual machines or bot. In network based deployment, Network traffic is analyzed and can be deployed anywhere in the network hierarchy. **Specificity:** It defines the detection methods can detect any specific botnet or all the kinds of botnet. Specific detection methods can only detect some specific types of botnet whereas general methods are applicable for detecting any kind of botnets. **Perspicacity:** It defines the ability of detection mechanism to differentiate between different bot families. **Discriminate** mechanisms for botnet detection not only identify the infected machine but also provide information about bot family. Indiscriminate mechanisms can only detect botnet infected machines but are unable to distinguish between different bot families. **Detection:** It defines the entire botnet detection or single bot detection. Entire botnet detection methods can detect the whole family of botnet whereas single bot detection techniques can only detect a single bot or a single infected machine. **Date from Dataset:** It shows the period of collected data for experimental results.

Lethic and Kelihos carried out a single spam campaign, while Cutwail carried out two different campaigns, at two distinct points in time.

3.2.3 Sink-holing/Redirecting Malicious Traffic Based

Seungwon Shin et al. [77] have analyzed a large amount of infected data obtained from three major spamming botnets, Conficker, MegaD and Srizbi. A cross analysis is performed between these botnets which involves an in-depth passive and active measurement study viewing the similarities and differences for these botnets. These botnets are categorized on the basis of their infection patterns. The Conficker botnet data is collected by setting up sink-holing servers and that of MegaD and Srizbi botnet through the Botlab project. Some interesting viewpoints regarding the three selected spamming botnets are observed. First, the geographical distribution of infected networks has helped in identifying more or less vulnerable locations. Second, the IP address density has helped in understanding relationships between the number of assigned IP address to the country and the number of infected networks of the country. Third, the remote accessibility of networks has shown the openness of the networks and finally, the dynamism of IP addresses has presented the usage of more dynamic IP addresses for vulnerable networks.

Ramachnadran and Feamster [78] have studied network level characteristics of spam Emails and after observing the network level behavior of spammers it is characterized by performing a joint analysis of the data collected at spam sinkholes for two domains. An analysis is performed on the packet traces, an archive of BGP route advertisements heard from the receiving network, trace routes from the receiving mail relay to the spammer's mail relay at the time the relay sent the mail, traces from the botnet C&C of the Bobax worm and traces of legitimate Email from the border router of a large campus network. The properties of the sender like the IP addresses that made connections to their mail server, packet traces of those connections passive TCP fingerprints, corresponding rote announcements etc. are also analyzed which cannot which cannot be forged by the spammers.

IV. FUTURE TRENDS & CHALLENGES IN DETECTING EMAIL SPAMMING BOTNETS

Email spam is not only irritant and annoying for users as it inundates and clogs user inboxes, but it also wastes resources and slows down the servers. It also wears down all the productivity gains that are resulted from the initiation of any information technology.

In order to alleviate the impact of spam, the most common countermeasures heavily rely on the filtering techniques that use the content of spam messages to distinguish them from ham messages and have high success rates of detecting Email spam but they also suffer from some limitations. First, the rate of false positives (legitimate Emails classified as spam) is as high as 15% [37]. Spammers have adopted techniques like image spam and they have also designed Emails to mislead filters that learn certain keyword patterns [47]. They devise new contents and formats to circumvent filters [52]. Second, the spammers compromise the lifetime of existing techniques by forging their Email addresses and misspelling spam messages. Third, these techniques are vulnerable to adversarial chaff and poising attacks [73]. Even more sophisticated filtering techniques are ineffective against new types of spamming methods. Fourth, these techniques are expensive to both maintain and scale. The techniques of IP based blacklisting are also losing their potency because spammers can evade these techniques by stealing IP addresses on the same local network or by stealing IP address blocks with BGP route hijacking. IP blacklists need to be updated continuously, which creates a nuisance for the administrators. Also IP addresses and prefix blacklists contain millions of entries that incur a large storage overhead [80].

Although researchers are acquainted with the existence of botnets several years ago but the research efforts which focus on combating and defending against botnet generated spam are still in their early stage. It is because the botnet traffic is similar to normal traffic and may contain encrypted communication thus, making it very hard to detect. Also, most of the internet users are unable to protect their computers which aid the spammers to evade protection techniques, thus spamming botnets grow rapidly by adopting root kit and employing fast flux and domain flux hosting in order to hide their origins and remain active for longer periods. The detection mechanisms [42] [43] [46] for botnet generated spam based on Email content or header features (URLs), can be evaded by the spammers. This is because the spammers randomly add legitimate URLs to the content of Emails with the intention to appear the Emails as legitimate and standard software is used to generate URLs to be used in HTML-based Emails. The approaches followed by the existing solutions [102-104] which contain pre-classified Email traffic of legitimate and suspicious pools are not feasible to adopt because the spammers have started to mix legitimate and spam URLs in the Email content. Moreover spammers are able to evade detection by extensively using URL obfuscation techniques. In addition, they often use customized URLs in order to reflect the recipient's Email address for tracking those users which visit spamming websites [42].

An important issue is to detect individual spam bots as differentiating a bot-user from a legitimate user individually is a difficult task because of two reasons; firstly, both the bot-users and the legitimate users may share a common computer and secondly each bot-user sends only a few spam Emails. The bot users are programmed in way that they receive and read Emails in order to look like legitimate users. Detecting bot-users as an aggregate can be a promising approach but it has two challenges. Firstly, it is challenging to differentiate between a bot user and a normal user behavior algorithmically and elusively and finding out subtle correlations among bot-user activities. Secondly, finding out that how a large volume of data can be efficiently analyzed in order to reveal the correlations among hundreds of millions of users. It may need to process hundreds of gigabytes or terabytes of user activity logs [48]. The techniques like [73] can be evaded if the spam is not based on templates and when spam uses multiple interleaving templates generated by different bots. Most of the detection mechanisms in this category need a huge amount of data which cannot be analyzed in real-time.

Detection of spamming botnets can be performed effectively by exploiting various properties of these botnets. One important property of spamming botnets is that they are involved in one or more spamming campaigns. Since the Email messages in a spam campaign share similarities in content, or links to the same target URL, this feature of spam campaigns can be exploited by clustering or correlating them to track down the spamming botnets. Another common property of botnets is the collective behavior or group activities of bot members of the same botnet which include sending/receiving control traffic, downloading new codes, migration of the communication channel, and performing malicious behaviors like launching spam campaigns for pushing same spam to millions of Email-boxes [105]. By accurately clustering such communal behavioral characteristics of these spamming bots belonging to the same botnet, spam signatures can be created which can characterize a campaign immediately. Also some distinguishable features of Botnet DNS and Legitimate DNS can be exploited to detect spamming botnets [106].

Most of the efforts towards spamming botnet detection are on the receiver's end meaning that they are reactive defense mechanisms in which the key challenge is that how well we can measure the activities of the spammers, since, they are using intelligent spam delivery infrastructure by targeting and harvesting addresses in a more lively fashion that it is invisible to a spam recipient. But on the other hand, detecting, measuring and mitigating botnet generated spam can also be accomplished by analyzing it at the source or sender's level in a proactive manner. This can be comprehended by using a feature of botnets that they have to permit new machines to become the part of their spamming process. An ideal solution involves infiltration [72] in which one can join the network and after performing investigations while being part of the botnet infrastructure, he might be able to contain the botnet or take it down from the inside. It facilitates us to collect current spam messages sent by a specific botnet by directly interfering with the botnet control servers.

Infiltration approaches can be more promising and can be mitigating, manipulating or exploiting. In mitigating strategies, the aim is to slow down the spamming botnets by consuming resources e.g. by attempting DoS attacks against C&C servers. In Manipulation strategies, the knowledge about the command protocols is required to manipulate and inject commands. The Exploitation strategy exploits the use of bugs which are found in the spamming bots. These bugs can be used to perform actions on the infected machines. A combination of actions on the addressing and command layer of botnet is essential to launch a botnet infiltration attack. In addition, redirecting bots to a controlled server or to command them to perform a self-removal can be the most effective countermeasure against spamming botnets. The most important challenge in this regard is that these approaches are performed stealthy and can put at risk the security of the very hosts they inquire about to protect. If this offensive detection is discovered by the botmaster, there is a possibility that he/she will direct attack traffic to the responsible host(s) in defy defense [31].

Spam sending botnets use several techniques in their malware and infrastructure to make them robust to typical spam detection techniques [107]. According to [107], the new inclination in Email spam consists of such Botnet attachments which include attached files like images and even more sophisticated types of graphical images in which like background noise is created in pictures, letters are replaced by images, the elements are rotated at different angles unique fonts or fonts of different sizes, animated spam and GIF images are used. In order to handle these kinds of Email spam, techniques based on low-level image features [109-113] and a combination of OCR with low-level image features [114-116] are used. Another technique called Image Texture Analysis-Based Image Spam Filtering (ITA-ISF) [117] is proposed which can filter image spam based on image texture analysis. But with the passage of time, software developers are trying to swindle anti-spam at a new intensity by switching from image and graphic spam to excel spreadsheets, pdf documents, or even mp3 files and Zip file attachments which are mostly infected with malware [118].

Since centralized architectures offer a single point of failure for the botnet, the botmasters prefer the more stable architectures like P2P based architecture which is more resilient to defense methods and countermeasures than the traditional centralized architecture. The appearances of new P2P techniques have caused some challenges for the current detection against botnets. Botmasters are now leveraging three kinds of C&C mechanisms of P2P architecture i.e. Pull, Push and Pull and Push. Most of the existing spamming botnet detection techniques concentrate on specific protocol characteristics and network behavior characteristics of a known botnet. But to handle these new categories of P2P C&C mechanisms, the characteristics of each category of C&C mechanism should be explored. This can be done by first extracting the traffic and behavioral characteristics of each category of P2P C&C mechanisms in a controlled environment and then train the Support Vector Machine (SVM) by using the extracted characteristics.

Another important challenge in detecting P2P spamming botnets is that the current active probing detection methods can only obtain a part of P2P botnet peers. To resolve this problem the passive probing algorithm should be integrated into active probing algorithm so as to improve the capacities of the topology discovery. Also in the meantime some passive monitoring points should also be established on the network for collecting the information about bots passively.

The botnet designers are now drifting towards the use of existent and legal P2P networks (also named as leeching botnets) [19] for instance, Storm bonet [10][64] uses Overnet network to find and communicate with its bots. Thus, it is recommended that leeching botnets should be circumspectly considered while designing protocols and systems for the detection of spamming botnets in future. As most of the botnet studies are based on their size estimation i.e. estimating number of compromised bots within a particular botnet, thus it is suggested by [123] that future research should focus on analyzing malicious functionalities and the impact of specific botnets of particular importance to society.

Recently there are two future challenges for botnet detection as spamming botnets are moving towards Cloud and Mobile infrastructures and thus there is need to protect them from botnets. First challenge comes with a possibility of spamming attacks originating from Cloud based botnets or Botclouds. One instance of this can be found in [121]. Botclouds are more advantageous for botmasters because they are always accessible, require less time to converge and maximize utilization of the cloud resources. The above mentioned techniques both active and passive detection techniques are not useful in this regard because deployment of these techniques on clouds is complicated as porting honey pots or performing infiltration on a cloud requires service level agreements (SLAs) with CSPs to monitor the activity logs of machines used by customers. According to [122] a common solution is to blacklist the range of IP addresses from which the spam is being sent. But it might block access to many legitimate services, such as the customers who are hosting their Email servers in the Cloud.

Second challenge comes with that the Botmasters are now starting to avail new opportunities for sending Email spam like smart devices which incorporate new features, software and applications. These smart devices which include smart phones and smart TVs are guaranteed to have connections to Email and Social networks since internet access is almost mandatory. These smart devices come with inevitable vulnerabilities and are attractive targets to botnet spammers because of their sophisticated Operating systems and the absence of efficient host-level security software. A study in [153] presents the proof of concept for an Android botnet, which is able to turn smartphones that are running Android OS into SMS spamming zombies. The research in the area of mobile botnets and their detection is scant and only some of the researchers have explored it yet [136-139]. In addition, future botnets can be highly strong and robust against the existing defensive solutions developed for Email spam as they will become potentially massive and more harmful with multiple small botnets being joined together into one large "super-botnet" [119][120].

In real time, a number of commercial solutions have also been developed for detecting botnet generated spam like SonicWALL by Dell [140], IronPort S-Series, Secrity Agent, NetFlow and Security MARS by Cisco [141], OSG (Outbound Spam Guard) by PineApp [142], SWG (Symantec Web Gateway) by Symantec [143] etc.

The intensification and increase in the amount of Email spam remained rampant and the problem of Email spam was growing in the past years [144][108] (botnet generated spam volume on yearly basis are shown in Figure 9) but recently according to Internet Security Threat Report 2014 [145], it is analyzed that ongoing efforts against spamming botnets have assisted in the gradual decline of botnet spamming activities. For example 76% of spam Email was distributed by spam-sending botnets as compared to 79% in 2012 [146]. Thus most of the spam has reduced dramatically by a combination of several takedowns against peak spamming botnets (like Rustock, Storm, Grum, etc), better anti-spam protection, and shifting interests of botnet spammers from Email spamming to DDoS attacks. But all the Email users still have to contend with the significant amount of spam i.e. illegitimate Emails and its increasing volume intimidates to devastate the user's ability to recognize useful legitimate Emails.

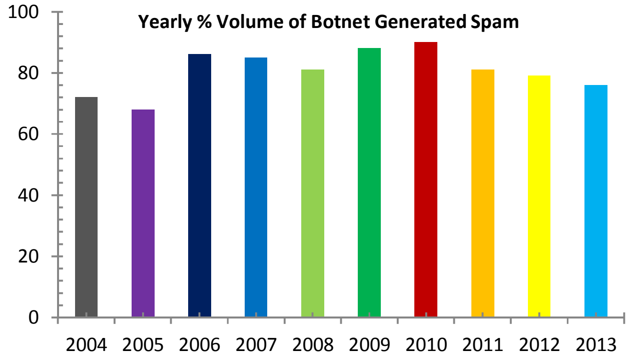


Figure 9: Yearly Percentages of botnet generated Spam

However, the spamming botnets form an endless, continuous and stirring chain of events as when one botnet goes offline, another come up with more aggressive spamming activities to fill the gap. Also the spamming botnets used to leverage dispersed sending mechanism with large-scale sending capability and non-obvious network statistic features (e.g., relatively low network traffic, dynamic network addresses, and rapid changes of spam signatures) which make anti-spam solutions ineffective. Moreover the blacklisting and spam filtering techniques are unable to uncover the invariant spamming and significant bot activity features. Thus, the security industry must keep on designing innovative ideas without only resting on their laurels in order to beat the cybercriminals since there exist a never ending arms race between the operators of spam botnet and the defenders of the internet security.

V. CONCLUSION

Botnet generated Email Spam is increasingly becoming a continual menace for the Internet Security as it is sent from compromised infected devices which form botnet armies and are controlled by one or more botmasters. Typically spamming botnets are involved in carrying out different spam campaigns. It is challenging to track and identify spam bots as spammers are becoming more agile and nimble and are expecting to use more effectively encrypted C&C communication and functionality. In this paper we have investigated the most recent solutions proposed typically for the detection of botnet generated Email spam and spamming botnets categorizing them according to their nature of defense. The traditional content based filtering and IP Blacklisting solutions are now ineffective to combat botnet generated spam and cannot decontaminate their factual origins. It is thus becoming critical to track down and launder these spam botnets. The passive or reactive approaches perform detection of spam bots on receiving the Email spam at the receiver's end. These approaches are successful to some extent but they require a plethora of Email spam messages to accurately identify botnet spam which may not be a practical solution. More recently the researchers and developers have turned their attention towards exploiting active or proactive countermeasures for identifying botnet Email spam and taking down spamming botnets as measuring and detecting these botnets at the source can be more effective and practical due to the heterogeneous nature of these spam bots. This comprehensive study presents a holistic analysis of the whole Email spamming arena by assembling almost all the research efforts in this regard, assisting the researchers in this field to gain a better understanding of the existing attempts to thwart botnet generated spam for producing effective solutions. Our future work targets the botnet phenomena in mobile devices and its detection comprehensively.

ACKNOWLEDGMENT

This Project was funded by the National Plan for Science, Technology and Innovation (MAARIFAH), King Abdulaziz City for Science and Technology, Kingdom of Saudi Arabia, Award Number 09-INF883-02.

REFERENCES

[1] Meng Weng Wong,"Sender Authentication What to do", A Messaging Anti-Abuse Working Group Whitepaper. www.openspf.org/blobs/sender-authentication-whitepaper.pdf [Last visited on 28/03/2014].

[2] Loshin, Pete, "Essential Email standards: RFCs and protocols made practical", John Wiley & Sons, Inc., 1999.

[3] Myers, J. and M. Rose, "Post Office Protocol - Version 3", STD 53, RFC 1939, May 1996, <http://www.rfc-editor.org/info/rfc1939>.

[4] Cormack G.V. and Lynam T.R. Spam Corpus Creation for TREC, Proc. CEAS 2005 - The Second Conference on Email and Anti-Spam, Palo Alto, July 2005.

[5] When Was the First Spam Email Sent? What did it advertise? http://Email.about.com/od/Emailtrivia/f/first_spam.htm [Last visited on 04/05/14].

[6] Crispin, M., "Internet Message Access Protocol - Version 4rev1", RFC 2060, December 1996, <http://www.rfc-editor.org/info/rfc2060>.

[7] Messaging Anti-Abuse Working Group, "Email metrics program: Report #15 – first, second and third quarter 2011", Tech. rep.

[8] Dominik Schatzmann, Martin Burkhart, Thrasyvoulos Spyropoulos, "Inferring Spammers in the Network Core", PAM '09, Proceedings of Passive and Active Network Measurement 10th International Conference, Pages 229-238, April 1-3 2009, Seoul, Korea.

[9] Alex Brodsky, Dmitry Brodsky, "A Distributed Content Independent Method for Spam Detection", HotBots'07, Proceedings of the first conference on First Workshop on Hot Topics in Understanding Botnets, Pages 3-3, April 10, 2007, Cambridge, USA.

[10] J. B. Grizzard, Sharma, V., Nunnery, C., Kang, B. B., Dagon, D, "Peer-to-peer botnets: overview and case study", HotBots'07, Proceedings of the first conference on First Workshop on Hot Topics in Understanding Botnets, Pages 1-1, April 10, 2007, Berkeley, USA.

[11] N. Feamster, "Open problems in BGP anomaly detection", WISP'04, ISMA Workshop on Internet Signal Processing, San Diego, CA, November 11-14, 2004.

[12] Emre Yuce, "A Literature Survey about Recent Botnet Trends", GÉANT Network, ULAKBIM, Turkey, Rep.JRA2 T4, 2012.

[13] Chao Li , Wei Jiang , Xin Zou, "Botnet: Survey and Case Study", ICICIC'09, Fourth International Conference on Innovative Computing, Information and Control, December 7-9, 2009. Kaohsiung, Taiwan.

[14] Sergio S.C. Silva, Rodrigo M.P. Silva Raquel C.G. Pinto, Ronaldo M. Salles, "Botnets: A Survey", Computer Networks: The International Journal of Computer and Telecommunications Networking, Pages 378-403, Volume 57 Issue 2, February, 2013.

[15] N. Ianelli, A. Hackworth, "Botnets as a Vehicle for Online Crime", Coordination Center, CERT cMellon University, Canegie CERT, 2005.

[16] P. Bacher, T. Holz, M.Kotter, G. Wicherski, "Know Your Enemy: Tracking Botnets (using honeynets to learn more about bots)", Technical Report, the Honeynet Project, 2008.

[17] R. Puri, "Bots & Botnet: An overview", SANS Institute InfoSec Reading Room, 2003.

[18] Ping Wang, Lei Wu, Baber Aslam, and Cliff C. Zou. "A Systematic Study on Peer-to-Peer Botnets", ICCCN'09, Proceedings of 18th International Conference on Computer Communications and Networks, Pages 1-8, August 3-6, 2009, IEEE Computer Society, Washington, DC, USA.

[19] Zhaosheng Zhu, Guohan Lu, Yan Chen, Zhi Fu, Roberts P, Keesook Han, "Botnet Research Survey", COMPSAC '08, Computer Software and Applications, 32nd Annual IEEE International, Pages 967-972, July 28-Aug 1 2008, Turku.

[20] Maryam Feily, Alireza Shahrestani, Sureswaran Ramadass, "A Survey of Botnet and Botnet Detection", SECURWARE'09, Proceedings of Third International Conference on Emerging Security Information, Systems and Technologies, Pages 268-273, June 18-23, 2009, Athens, Glyfada.

[21] Yun Ho Shin, Eul Gyu Im, "A survey of botnet: Consequences, Defenses and Challenges", Joint Workshop on Internet Security, Pages 1-11, 2009.

[22] Jing Liu, Yang Xiao, Kaveh Ghaboosi, Hongmei Deng, and Jingyuan Zhang, "Botnet: Classification, Attacks, Detection, Tracing, and Preventive Measures", Hindawi Publishing Corporation, EURASIP Journal on Wireless Communications and Networking, 11 pages, July 19, 2009.

[23] Michael Bailey, Evan Cooke, Farnam Jahanian, Yunjing Xu, Manish Karir, "A Survey of Botnet Technology and Defenses", CATCH '09, Proceedings of the 2009 Cyber security Applications & Technology Conference for Homeland Security, Pages 299-304, March 3-4, 2009, Washington, DC, USA.

[24] Areej Al-Bataineh and Gregory White, "Detection and Prevention Methods of Botnet-generated Spam", MIT Spam Conference, Pages 1-10, March 2009, MIT, Cambridge, Massachusetts.

[25] Tyagi, Amit Kumar; Aghila, G, "A Wide Scale Survey on Botnet", International Journal of Computer Applications, Page 9, Vol. 34, November 2011.

[26] Gianluca Stringhini, The Spammer, the Botmaster, and the Researcher: on the Arms Race in Spamming Botnet Mitigation -Major Area Exam, December 5, 2011.

[27] Haritha S Nair, Vinodh Ewards S E, "A Study on Botnet Detection Techniques", International Journal of Scientific and Research Publications, Volume 2, Issue 4, April 2012,

[28] Erdem Alparslan, Adem Karahoca and Dilek Karahoca, "BotNet Detection: Enhancing Analysis by Using Data Mining Techniques", INTECH, Advances in Data Mining Knowledge Discovery and Applications, Chapter 17. September 12, 2012

[29] Lei Zhang, Shui Yu, Di Wu, Paul Watters, "A Survey on Latest Botnet Attack and Defense", TrustCom'11, IEEE 10th International Conference on Trust, Security and Privacy in Computing and Communications, Pages 53-60, November 16-18 2011.

[30] Sheharbano Khattak, Naurin Rasheed Ramay, Kamran Riaz Khan, Affan A. Syed and Syed Ali Khayam "A Taxonomy of Botnet Behavior, Detection and Defense", Journal of Communications Surveys & Tutorials, IEEE, Pages 1 – 27, Vol. PP, Issue 99, October 2013.

[31] Ahmad Karim, Rosli Bin Salleh, Muhammad Shiraz, Syed Adeel Ali Shah, Irfan Awan, Nor Badrul Anuar, "Botnet detection techniques: review, future trends and issues", Journal of Zhejiang University-SCIENCE C-Computers & Electronics, February 2014.

[32] R. Beverly and K. Sollins, "Exploiting Transport-Level Characteristics of Spam", CEAS'08, Proceedings of the Fifth Conference on Email and Anti-Spam, August 2008, Mountain View, CA.

[33] Anna Sperotto, Gert Vliek, Ramin Sadre and A Pras,"Detecting Spam at the Network Level", EUNICE'09, Proceedings of 15th Open European Summer School and IFIP TC6.6 Workshop on The Internet of the Future, Pages 208-216, September 7-9, 2009, Barcelona, Spain.

[34] Miao Ye, Tang Tao, Fan-Jin Mai, Xiao-Hi Cheng "A Spam Discrimination Based on Mail Header Feature and SVM," WiCOM'08, Proceedings of 4th International Conference on Wireless Communications, Networking and Mobile Computing, October 12-14, 2008, Dalian, China.

[35] Chih-Hung Wu, "Behavior-based spam detection using a hybrid method of rule-based techniques and neural networks", Expert Systems with Applications, Pages 4321-4330, Volume 36 Issue 3, April, 2009

[36] Jyh-jian Sheu, "An Efficient Two-phase Spam Filtering Method Based on E-mails Categorization", International Journal of Network Security, Pages 34-43, Volume 9, July 2009.

[37] Henrique Gomes, Christiano Cazita, Jussara M. Almeida, Virgillo Almeida, Wagner Meira J, "Characterizing a Spam Traffic", IMC '04 Proceedings of the 4th ACM SIGCOMM conference on Internet measurement, Pages 356-369, October 25–27, 2004 Taormina, Sicily, Italy.

[38] Son T. Vuong and Mohammed S. Alam," Advanced Methods for Botnet Intrusion Detection Systems", Dr. Pawel Skrobanek (Ed.), ISBN: 978-953-307-167-1, InTech, March 2011.

[39] Chris Kanich, Kirill Levchenko, Brandon Enright, Geoffrey M. Voelker, Stefan Savage, "The Heisenbot Uncertainty Problem: Challenges in Separating Bots from Chaff", LEET'08, Proceedings of the first USENIX workshop on Large Scale Exploits and Emergent Threats, Article 10, April 2008, USENIX Association, Berkeley, CA, USA.

[40] S. Lab. March 2011 intelligence report. Symantec Report 2011, 2011.

[41] Anirudh Ramachandran, Nick Feamster and David Dagon, "Revealing Botnet Membership Using DNSBL Counter-Intelligence", SRUTI'06, Proceedings of 2nd USENIX Steps to Reducing Unwanted Traffic on the Internet, Volume 2, Pages 49-54, July 7, 2006, USENIX Association, Berkeley, CA, USA.

[42] Y. Xie, F. Yu, K. Achan, R. Panigrahy, G. Hulten and I. Osipkov, "Spamming botnets: signatures and characteristics," SIGCOMM'08, Proceedings of ACM SIGCOMM conference, Pages 171-182 August 17-22 2008, Seattle, Washington, USA.

[43] Li Zhuang, John Dunagan, Daniel R. Simon, Helen J. Wang, J. D. Tygar, "Characterizing Botnets from Email Spam Records", LEET'08, Proceedings of the 1st Usenix Workshop on Large-Scale Exploits and Emergent Threats, Article no 2, April 2008, USENIX Association, Berkeley, CA, USA.

[44] P. Sroufe, S. Phithakkitnukoon, R. Dantu, J. Cangussu, "Email shape analysis for spam botnet detection", CCNC'09, Proceedings of 6th IEEE Consumer Communications and Networking Conference, Pages 1-2, January 10-13, 2009, Las Vegas, NV.

[45] Tatsuya Mori, Holly Esquivel, Aditya Akella, Akihiro Shimoda, Shigeki Goto, "Understanding Large-Scale Spamming Botnets From Internet Edge Sites", CEAS 2010 - Seventh annual Collaboration, Electronic messaging, Anti-Abuse and Spam Conference July13-14, 2010, Redmond, Washington, USA.

[46] Masaru Takesue, "Lightweight Detection of Spamming Botnets", SECURWARE 2011, Proceedings of the Fifth International Conference on Emerging Security Information, Systems and Technologies, August 21-27, 201, France.

[47] A. Ramachandran, N. Feamster and S. Vempala, "Filtering spam with behavioral blacklisting", CCS'07, Proceedings of the 14th ACM conference on Computer and Communications Security, Pages 342-351, Oct 2007, Alexandria, Virginia, USA.

[48] Y. Zhao, Y. Xie, F. Yu, Q. Ke, Y. Yu, Y. Chen, E. Gillum, "BotGraph: large scale spamming botnet detection", NSDI '09, Proceedings of the 6th USENIX symposium on Networked systems design and implementation, Pages 321-334, April 2009, Boston, Massachusetts.

[49] Abhinav Pathak, Feng Qian, Y. Charlie Hu, Z. Morley Mao, Supranamaya Ranjan, "Botnet spam campaigns can be long lasting: evidence, implications, and analysis", SIGMETRICS '09, Proceedings of the eleventh international joint conference on Measurement and modeling of computer systems, Pages 13-24, June 15-19 2009, ACM New York, NY, USA.

[50] Orestis Tsigkas, Olivier Thonnard, Dimitrios Tzovaras, "Visual spam campaigns analysis using abstract graphs representation", VizSec '12, Proceedings of the Ninth International Symposium on Visualization for Cyber Security, Pages 64-71, October 15, 2012, Seattle, USA.

[51] Smith K, Al-Shaer, E Elbadawi, K, "Information Theoretic Approach for Characterizing Spam Botnets Based on Traffic Properties", ICC '09, Proceedings of IEEE International Conference on Communications, Pages 1-5, June 14-18, 2009, Dresden.

[52] W. K. Ehrlich, A. Karasaridis, D. Liu and D. Hoeflin, "Detection of spam hosts and spam bots using network flow traffic modeling", LEET 2010, Proceedings of the 3rd USENIX Conference on Large-scale Exploits and Emergent Threats: botnets, spyware, worms, and more, Pages 7-14, April 27, 2010, San Jose, California, USA.

[53] Ping-Hai Lin, Po-Ching Lin, Pin-Ren Chiou, Chien-Tsung Liu, "Detecting Spamming Activities by Network Monitoring with Bloom Filters", ICACT 2013, Proceedings of 15th International Conference on Advanced Communication Technology, Pages 163-168, January 27-30, 2013, Pyeong Chang, Korea.

[54] Z. Duan, P. Chen, F. Sanchez, Y. Dong, M. Stephenson, J. M. Barker, "Detecting spam zombies by monitoring outgoing messages", Proceedings of IEEE Transactions on Dependable and Secure Computing, Pages 198–210, Volume 9, Issue 2, March-April 2012,

[55] Diogo Mónica, Carlos Ribeiro, "Leveraging honest users: stealth command-and-control of botnets", WOOT '13, Proceedings of the 7th USENIX Workshop on Offensive Technologies, Pages 7-7, August 2013, Washigton, D.C.

[56] Zhitang Li; Jun Hu; Zhengbing Hu; Bingbing Wang; Liang Tang; Xin Yi, "Measuring the botnet using the second character of bots", Journal of Networks (JNW), Pages 98-105,Volume 5, Issue 1, Jan 2010.

[57] Czosseck, C, Tallinn, Estonia, Klein, G Leder F, "On the arms race around botnets-Setting up and taking down botnets", ICCC 2011, Proceedings of 3rd International Conference on Cyber Conflict, Pages 1-11, June 7-10, 2011, Tallinn, Estonia.

[58] Evan Cooke, Farnam Jahanian, 'The Zombie Roundup: Understanding, Detecting, and Disrupting Botnets", SRUTI '05, Proceedings of the Steps to Reducing Unwanted Traffic on the Internet Workshop, Pages 39-44, July 2005, Cambridge, MA.

[59] Moheeb Abu Rajab, Jay Zarfoss, Fabian Monrose, Andreas Terzis. "A multifaceted approach to understanding the botnet phenomenon", IMC'06, Proceedings of the 6th ACM SIGCOMM conference on Internet measurement, Pages 41 – 52, New York, USA.

[60] Thorsten Holz, Moritz Steiner, Frederic Dahl, Ernst Biersack, Felix Freiling, "Measurements and Mitigation of Peer-to-Peer-based Botnets: A Case Study on Storm Worm", LEET'08, Proceedings of the 1st Usenix Workshop on Large-Scale Exploits and Emergent Threats, Article No. 9, Pages 1-9, April 15, 2008, San Francisco, CA, USA.

[61] Olivier Thonnard, Marc Dacier, "A strategic analysis of spam botnets operations", CEAS '11, Proceedings of the 8th Annual Collaboration, Electronic messaging, Anti-Abuse and Spam Conference, Pages 162-171, September 1-2, 2011, ACM New York, NY, USA.

[62] Jianwei Zhuge, Thorsten Holz, Xinhui Han, Jinpeng Guo1, Wei Zou, "Characterizing the IRC-based Botnet Phenomenon', Peking University & University of Mannheim Technical Report, December 3, 2007.

[63] MengjunXie, Heng Yin, Haining Wang, "An effective defense against Email spam laundering", CCS '06, Proceedings of the 13th ACM conference on Computer and communications security, Pages 179-190, October 30-November 3, 2006, Alexandria, Virginia, USA.

[64] Christian Kreibich , Chris Kanich , Kirill Levchenko , Brandon Enright , Geoffrey M. Voelker , Vern Paxson , Stefan Savage, On the spam campaign trail, Proceedings of the 1st Usenix Workshop on Large-Scale Exploits and Emergent Threats, Pages 1-9, April 15, 2008, San Francisco, California, USA.

[65] Chia Yuan Cho, Juan Caballeroy, Chris Grier, Vern Paxsonz, Dawn Song, "Insights from the Inside: A View of Botnet Management from Infiltration", LEET'10, Proceedings of the 3rd USENIX Workshop on Large-Scale Exploits and Emergent Threats: botnets, spyware, worms, and more, Pages 2-9, April 27, 2010, San Jose, California, USA.

[66] Christian Kreibich, Chris Kanich, Kirill Levchenko, Brandon Enright, Geoffrey M Voelker, Vern Paxson, Stefan Savage, "Spamcraft: An inside look at spam campaign orchestration", LEET'09, Proceedings of 2nd USENIX Workshop on Large-Scale Exploits and Emergent Threats, Pages 4-12, April 21, 2009, Boston, MA, USA.

[67] Brett Stone-Gross, Thorsten Holzz, Gianluca Stringhini, Giovanni Vigna, "The Underground Economy of Spam: A Botmaster's Perspective of Coordinating Large-Scale Spam Campaigns", LEET'11, Proceedings of the 4th USENIX Workshop on Large-Scale Exploits and Emergent Threats, Pages 4-11, March 29, 2011, Boston, MA, USA.

[68] Mushtaq A, "Smashing the Mega-d/Ozdok botnet in 24 hours", http://blog.fireeye.com/research/2009/11/smashing-the-ozdok.html (2009) [Last visited on 23/04/14]

[69] Krebs B, "Takedowns: The Shuns and Stuns That Take the Fight to the Enemy", McAfee Security Journal, Pages 5-8, 2010.

[70] Cranton T, "Cracking Down on Botnets", http://blogs.technet.com/b/microsoft_blog/archive/2010/02/25/cracking-down-on-botnets.aspx (2011) [Last visited on 01/05/14]

[71] Krebs B, "Researchers Kneecap Pushdo Spam Botnet", http://krebsonsecurity.com/2010/08/researchers-kneecap-pushdo-spam-botnet/ (2010) [Last visited on 01/05/14]

[72] Felix Leder, Tillmann Werner, and Peter Martini, "Proactive Botnet Countermeasures- An Offensive Approach", CCDCOE'09, Proceedings of Conference on Cyber Warfare, June 17-19, 2009, Tallinn, Estonia.

[73] A. Pitsillidis, K. Levchenko, C. Kreibich, C. Kanich, G. M. Voelker, V. Paxson, N. Weaver and S. Savage, "Botnet Judo: Fighting Spam with Itself", NDSS'10, Proceedings of the 17th Annual Network and Distributed System Security Symposium, February 28- March 3, 2010, San Diego, California, USA.

[74] G. Stringhini, T. Holz, B. Stone-Gross, C. Kruegel, G. Vigna, "BotMagnifier: locating spambots on the internet", SEC'11, Proceedings of the 20th USENIX conference on Security, August 2011, Pages 28-43, San Francisco, CA, USA.

[75] John P. John, Alexander Moshchuk, Steven D. Gribble, Arvind Krishnamurthy, "Studying Spamming Botnets Using Botlab", NSDI'09, Proceedings of the 6th ACM/USENIX Symposium on Networked Systems Design and Implementation, Pages 291-306, April 2009, Boston, MA.

[76] Gianluca Stringhini, Manuel Egele, ApostolisZarras, Thorsten Holz, Christopher Kruegel, Giovanni Vigna, "B@bel: Leveraging Email Delivery for Spam Mitigation", Proceedings of the 21st USENIX Security Symposium, Pages 16-32,August 8-10, 2012, Bellevue, WA.

[77] Seungwon Shin, Raymond Lin, GuofeiGu, "Cross-Analysis of Botnet Victims: New Insights and Implications", Recent Advances in Intrusion Detection Lecture Notes in Computer Science, Pages 242-261,Volume 6961, 2011.

[78] Anirudh Ramachandran, Nick Feamster, "Understanding Network-Level Behavior of Spammers", SIGCOMM '06, Proceedings of the 2006 Conference on Applications, Technologies and Architectures and Protocols for Computer Communications, Pages 291-302, September 11-16, 2006, Pisa Italy.

[79] Zeidanloo, Hossein Rouhani, and Azizah Abdul Manaf. "Botnet Command and Control Mechanisms." In Proceedings of the 2009 Second International Conference on Computer and Electrical Engineering-Volume 01, Pages 564-568. IEEE Computer Society, December 28-30, 2009, Dubai, UAE.

[80] H. Esquivel, T. Mori, and A. Akella, "Router-Level Spam Filtering Using TCP Fingerprints: Architecture and Measurement-Based Evaluation", CEAS '09, Proceedings of Sixth Conference on Email and Anti-Spam, July 16-17, 2009, Mountain View, California, USA.

[81] Alex Brodsky, Dimitry Brodsky, "A Distributed Content Independent Method for Spam Detection", In 1st USENIX Workshop on Hot Topics in Understanding Botnet (2007).

[82] C. Kanich, C. Kreibich, K. Levchenko, B. Enright, G. Voelker, V. Paxson, and S. Savage, "Spamalytics: an empirical analysis of spam marketing conversion", CCS '08, Proceedings of the 15th ACM conference on Computer and communications security, Pages 3-14, October 27-31, 2008, Alexandria, Virginia, USA.

[83] The history of the botnet- Part II, http://countermeasures.trendmicro.eu/2010/09/ [Last visited on 23/04/14]

[84] Bagle (computer worm), http://en.wikipedia.org/wiki/Bagle [Last visited on 23/04/14]

[85] Top Spam Botnets Exposed, http://www.secureworks.com/cyber-threat-intelligence/threats/topbotnets/ [Last visited on 25/04/14]

[86] Rustock Botnet, http://en.wikipedia.org/wiki/Rustock_botnet [Last visited on 25/04/14]

[87] Paul Royal, DAMBALLA "On the Kraken and Bobax Botnets", April 2008, http://bandwidthco.com/whitepapers/compforensics/malware/bots/OntheKrakenandBoBaxBotnets.pdf [Last visited on 26/04/14]

[88] Botnets and Spam Development, http://www.eleven.de/tl_files/timeline/index-en.html [Last visited on 26/04/14]

[89] Storm Botnet, http://en.wikipedia.org/wiki/Storm_botnet [Last visited on 27/04/14]

[90] Srizbi Botnet, http://en.wikipedia.org/wiki/Srizbi_botnet [Last visited on 26/04/14]

[91] Srizbi Botnet Sending Over 60 Billion Spams a Day, http://www.darkreading.com/risk/srizbi-botnet-sending-over-60-billion-spams-a-day/d/d-id/1129480? (2008) [Last visited on 29/04/14]

[92] The Kraken Botnet Returns, http://www.darkreading.com/risk-management/the-kraken-botnet-returns/d/d-id/1090438? (2010). [Last visited on 29/04/14]

[93] IT Security & Network Security News & Reviews: The Rise and Fall of the Srizbi Botnet, www.eweek.com/c/a/Security/The-Rise-and-Fall-of-the-Srizbi-Botnet/ [Last visited on 29/04/14]

[94] Festi Botnet spins up to become one of the main spamming botnets, http://www.symantec.com/connect/blogs/festi-botnet-spins-become-one-main-spamming-botnets (November 2009) [Last visited on 29/04/14]

[95] Boscovich, R; "Taking Down Botnets: Microsoft and the Rustock Botnet", http://blogs.technet.com/b/microsoft_on_the_issues/archive/2011/03/18/taking-down-botnets-microsoft-and-the-rustock-botnet.aspx (March 2011) [Last visited on 29/04/14]

[96] Rafael A. Rodríguez-Gómez, Gabriel Maciá-Fernández, Pedro García-Teodoroc, Moritz Steiner, Davide Balzarotti, "Resource monitoring for the detection of parasite P2P botnets", International Jornal of Compter networks.

[97] How Kaspersky Lab and Crowd Strike Dismantled the Second Hlux/Kelihos Botnet: Success Story, http://www.kaspersky.com/au/about/news/virus/2012/How_Kaspersky_Lab_and_CrowdStrike_Dismantled_the_Second_Hlux_Kelihos_Botnet_Success_Story [Last visited on 01/05/14]

[98] Jan Göbel, Thorsten Holz, Philipp Trinius, "Towards Proactive Spam Filtering", DIMVA '09 Proceedings of the 6th International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment Lecture Notes in Computer Science, Pages 38-47, Volume 5587, July 9-10, 2009, Como, Italy.

[99] Grum botnet http://en.wikipedia.org/wiki/Grum_botnet [Last visited on 01/05/14]

[100] McAfee Threats Report: Fourth Quarter 2012, http://www.mcafee.com/au/resources/reports/rp-quarterly-threat-q4-2012.pdf [Last visited on 01/05/14]

[101] https://www3.trustwave.com/support/labs/spam_statistics.asp [Last visited on 01/05/14]

[102] Hyang-Ah Kim and Brad. Karp, "Autograph: Toward automated, distributed worm signature detection", SSYM'04, Proceedings of the 13th conference on USENIX Security Symposium, Pages 271- 286, Volume 13, 2004, San Diego, CA.

[103] James Newsome, Brad Karp, and Dawn Song, "Polygraph: Automatically generating signatures for polymorphic worms", SP '05, Proceedings of the 2005 IEEE Symposium on Security and Privacy, Pages 226-241, Washington, DC, USA.

[104] Zhichun Li, Manan Sanghi, Yan Chen, Ming Yang Kao, and Brian Chavez. Hamsa, "Fast signature generation for zero-day polymorphic worm with provable attack resilience", SP '06, Proceedings of IEEE Symposium on Security and Privacy, Pages 32-47, May 21-24, 2006, Oakland, California, USA.

[105] Hyunsang Choi, Heejo Lee, Hyogon Kim, "BotGAD: detecting botnets by capturing group activities in network traffic", COMSWARE '09, Proceedings of the Fourth International ICST Conference on COMmunication System softWAre and middleware, Article no 2, June 15-19, 2009, Dublin, Ireland.

[106] Hyunsang Choi, Hanwoo Lee, Heejo Lee, Hyogon Kim, "Botnet Detection by Monitoring Group Activities in DNS Traffic", Proceedings of the 7th IEEE International Conference on Computer and Information Technology, Pages 715-720, October 16-19, 2007, Aizu-Wakamatsu, Fukushima.

[107] Zhang, "A Sublexical Unit Based Hash Model Approach for Spam Detection", Ph.D. Dissertation 2009. The University of Texas at San Antonio.

[108] Fossi, Marc, Gerry Egan, Kevin Haley, Eric Johnson, Trevor Mack, Téo Adams, Joseph Blackbird et al. "Symantec internet security threat report trends for 2010." Volume 16 (2011): 20.

[109] Qiao Liu, Zhigang Qin, Hongrong Cheng, and Mingcheng Wan, "Efficient Modeling of Spam Images," IITSI '10, Proceedings of the third International Symposium on Intelligent Information Technology and Security Informatics, Pages 663-666, April 2-4, 2010, Jinggangshan.

[110] B. Mehta, S. Nangia, M. Gupta, and W. Nejdl, "Detecting Image Spam Using Visual Features and Near Duplicate Detection", WWW 08, Proceedings of the 17th international conference on World Wide Web, Pages 497-506, April 21 - 25, 2008, Beijing, China.

[111] Mark Dredze, Reuven Gevaryahu, Ari Elias-Bachrach, "Learning Fast Classifiers for Image Spam," CEAS 07, Proceedings of the 4th Conference on Email and Anti-Spam, Pages 487-493, August 2-3, 2007, Mountain View, California, USA.

[112] Zhe Wang, William Josephson, Qin Lv, Moses Charikar, Kai Li, "Filtering Image Spam with Near-Duplicate Detection," CEAS 07, Proceedings of the 4th Conference on Email and Anti-Spam, August 2-3, 2007, Mountain View, California, USA.

[113] B. Biggio, G. Fumera, I. Pillai, and F. Roli, "Image Spam Filtering Content Obscuring Detection," CEAS 07, Proceedings of the 4th Conference on Email and Anti-Spam, August 2-3, 2007, Mountain View, California, USA.

[114] G. Fumera, I. Pillai, F. Roli, and B. Biggio, "Image Spam Filtering Using Textual and Visual Information," Proceedings of the MIT Spam Conference 2007, Cambridge, MA, USA.

[115] P. Klangpraphant and P. Bhattarakosol, "PIMSI: A Partial Image Spam Inspector," Future Tech 2010, Proceedings of 5th International Conference on Future Information Technology, Pages 1-6, 21-23 May 2010, Busan.

[116] F. Gargiulo and C. Sansone , "Combining Visual and Textual Features Filtering Spam Emails," ICPR 08, Proceedings of the 19th International Conference on Pattern Recognition, Pages 1-4, December 8-11, 2008, Tampa, FL.

[117] Basheer Al-Duwairi, Ismail Khater and Omar Al-Jarrah, "Detecting Image Spam Using Image Texture Features", International Journal for Information Security Research (IJISR), Pages 334-356, Volume 2, Issue 3/4, September/December 2012.

[118] Markus Selinger, "A New AV-TEST Study: Dangerous Spam E-Mails, Spam-More Dangerous than Ever Before", The Independent IT Security Institute, Magdeburg, Germany, April 11, 2011. http://www.av-test.org/…/pdf/avtest_2013-04_spam_english.pdf [Last visited on 18/04/14].

[119] R. Vogt, J. Aycock, and M. Jacobson, J, "Army of botnets", NDSS'07, Proceedings of the 14th Annual Network and Distributed System Security Symposium, Pages 111–123, 28th February - 2nd March, 2007, San Deigo, CA, USA.

[120] Ping Wang, S herri Sparks, and Cliff- C Zou, "An advanced hybrid peer-to-peer botnet", HotBots'07, Proceedings of the first conference on First Workshop on Hot Topics in Understanding Botnets, 2007, Cambridge, CA, USA.

[121] Washington Post http://blog.washingtonpost.com/securityfix/2008/07/amazon_hey_spammers_get_off_my.html. [Last visited on 20/04/14].

[122] Kassidy Clark, MartijnWarnier, Frances M. T Brazier,"BOT-CLOUDS, The Future of Cloud-based Botnets?" CLOSER 2011, Proceedings of the 1st International Conference on Cloud Computing and Services Science, May 2011, Noordwijkerhout.

[123] Daniel Plohmann, Elmar Gerhards, Felix Leder, "Botnets: Detection, Measurement, Disinfection & Defence", ENISA 2011, Proceedings of European Network and Information Security Agency, Pages 21-26, March 7, 2011,

[124] C.Associates.GTBot1. http://www3.ca.com/securityadvisor/pest/pest.aspx?id=453073312 (1998) [Last visited on 21/04/14].

[125] Technical Brief Understanding Botnets And Their Security challenges http://www.zscaler.com/pdf/technicalbriefs/tb_understanding_botnets.pdf [Last visited on 22/04/14].

[126] Gianluca Stringhini, Oliver Hohlfeldy, Christopher Kruegel, Giovanni Vigna, "The Harvester, the Botmaster, and the Spammer: On the Relations Between the Different Actors in the Spam Landscape", ASIA CCS '14, Proceedings of ACM Symposium on Information, Computer and Communications Security, Pages 353-364, 2014, Kyoto, Japan.

[127] Gianluca Stringhini, Manuel Egele, Christopher Kruegel, and Giovanni Vigna, "Breaking the Loop: Leveraging Botnet Feedback for Spam Mitigation", GSWC 2012, Proceedings of the Seventh Annual Graduate Student Workshop on Computing, October 5, 2012, Santa Barbara, California.

[128] Storm e-card malware keeps on coming https://www.virusbtn.com/blog/2007/08_17.xml [Last visited on 12/04/14].

[129] 9 Million E-card Spam Mails Pushed out in 48 Hours, http://www.spamfighter.com/News-8979-9-Million-E-card-Spam-Mails-Pushed-out-in-48-Hours-Sophos.htm [Last visited on 13/04/14]

[130] Alice Decker, David Sancho, Loucif Kharouni, Max Goncharov, Robert McArdie, "A study of the Pushdo / Cutwail Botnet", TrendMicro, May 2009.

[131] Ben Stock, Jan Göbel, Markus Engelberth, Felix C. Freiling, Thorsten Holz, "Waledac -Aanalysis of a Peer-to-Peer Botnet", EC2ND '09, Proceedings of the 2009 European Conference on Computer Network Defense, Pages 13-20, November 9-10, 2009, Milano, Italy.

[132] Greg Sinclair, Chris Nunnery, and Brent Byunghoon Kang, "The Waledac Protocol: The How and Why", MALWARE '09, Proceedings of the 4th International Conference on Malicious and Unwanted Software, Pages 69-77, October 13-14, 2009, Montreal, Quebec, Canada.

[133] Chris Nunnery, Greg Sinclair, and Brent ByungHoon Kang, "Tumbling Down the Rabbit Hole: Exploring the Idiosyncrasies of Botmaster Systems in a Multi-Tier Botnet Infrastructure", LEET'10, Proceedings of the 3rd USENIX conference on Large-scale exploits and emergent threats: botnets, spyware, worms, and more, Pages 1-1, April 2010, San Jose, California, USA.

[134] Chiang, K., Lloyd, L,"A case study of the rustock rootkit and spam bot", HotBots'07, Proceedings of the First Workshop on Hot Topics in Understanding Botnets, Pages 10-10, April 10, 2007, Cambridge, MA.

[135] Richard Boscovich, "Taking down Botnets: Microsoft and the Rustock Botnet", http://blogs.technet.com/b/microsoft_blog/archive/2011/03/17/taking-down-botnets-microsoft-and-the-rustock-botnet.aspx (May 17, 2011) [Last Visited on 01/05/14].

[136] Ickin Vural, Hein Venter, "Mobile Botnet Detection Using Network Forensics" Future Internet - FIS 2010, Proceedings of the Third Future Internet Symposium, Lecture Notes in Computer Science, Pages 57-67, Volume 6369, September 20-22, 2010, Berlin, Germany.

[137] Cui Xiang, Fang Binxing, Yin Lihua, Liu Xiaoyi, Zang Tianning, "Andbot: Towards Advanced Mobile Botnets", LEET'11, Proceedings of the 4th USENIX conference on Large-scale exploits and emergent threats, Pages 11-17, March 29, 2011, Boston, MA, USA.

[138] Byungha Choi, Sung-Kyo Choi, Kyungsan Cho, "Detection of Mobile Botnet Using VPN", IMIS 2013, Proceedings of the Seventh International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, July 3-5, 2013, Taichung.

[139] Ickin Vural, Hein Venter, "Combating Mobile Spam through Botnet Detection using Artificial Immune Systems", Journal of Universal Computer Science, Pages 750—774, Volume 8, Issue 6, March 28, 2012.

[140] The Anti-Spam & Email Security Platform, "Zombie Detection & Prevention", http://www.sonicwall.com/us/en/products/358.html [Last visited on 25/11/14].

[141] "Botnets: The New Threat Landscape" Cisco White paper, http://www.cisco.com/c/en/us/solutions/collateral/enterprise-networks/threat-control/networking_solutions_whitepaper0900aecd8072a537.pdf [Last visited on 25/11/14].

[142] PineApp™ Outbound Spam Guard™ , "An Anti-Botnet Solution", http://www2.pineapp.com/files/files/OSG_Datasheet_2011_print.pdf [Last visited on 25/11/14].

[143] "Botnet detection on Symantec Web Gateway (SWG)" http://www.symantec.com/docs/TECH134542 [Last visited on 25/11/14].

[144] Joanne Pimanova, "Email spam trend at a glance", http://www.Emailtray.com/blog/Email-spam-trends-2001-2012/ [Last visited on 1/05/14].

[145] Symantec Corporation Internet Security Threat Report 2014, Volume 19, 2014.
[146] Symantec Corporation Internet Security Threat Report 2013, Volume 18, 2013.
[147] Aleksandr Matrosov, Eugene Rodionov, "Festi botnet Analysis and Investigation", Proceedings of the 15th AVAR conference, Novemeber 12-14, 2012, Hang Zhou.
[148] Kelihos Botnet, https://community.infoblox.com/blogs/2014/02/13/kelihos-botnet [Last visited on 01/04/14].
[149] Grum Botnet http://en.wikipedia.org/wiki/Grum_botnet [Last visited on 01/04/14].
[150] Spamhaus, 2007, http://www.spamhaus.org/ [Last visited on 01/04/14].
[151] Caruana, Godwin, and Maozhen Li. "A survey of emerging approaches to spam filtering." ACM Computing Surveys (CSUR), Volume 44, Issue no. 2, Pages 1-27, February 2012.
[152] Spirin, Nikita, and Jiawei Han. "Survey on web spam detection: principles and algorithms." ACM SIGKDD Explorations Newsletter, Volume 13, Issue no. 2, Pages 50-64, 2012.
[153] K. Fogarty, "Just what we need: Malware to slave your android to a botnet", 2011, IT World.
[154] Kim, Won, Ok-Ran Jeong, Chulyun Kim, and Jungmin So. "The dark side of the Internet: Attacks, costs and responses." ,Information systems Pages 675-705, Volume 36, Issue no. 3, May 2011.
[155] S. Webb, J. Caverlee, and C. Pu, " Introducing the web spam corpus: Using email spam to identify web spam automatically", In Proceedings of the Third Conference on Email and Anti-Spam (CEAS), July 27-28, 2006, Mountain View, California USA.
[156] Chen, Jianxing, Romain Fontugne, Akira Kato, and Kensuke Fukuda. "Clustering Spam Campaigns with Fuzzy Hashing." In Proceedings of the AINTEC 2014 on Asian Internet Engineering Conference, Pages 66-7, ACM, November 26–28, 2014, Chiang Mai, Thailand.
[157] Dinh, Son, Taher Azeb, Francis Fortin, Djedjiga Mouheb, and Mourad Debbabi. "Spam campaign detection, analysis, and investigation." Digital Investigation, Pages 12-21, 12 (2015), 2015.
[158] Schäfer, Carlo. "Detection of Compromised Email Accounts used by a Spam Botnet with Country Counting and Theoretical Geographical Travelling Speed Extracted from Metadata." IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW), Pages 329 – 334, 2014, Naples.
[159] Stringhini, Gianluca, Oliver Hohlfeld, Christopher Kruegel, and Giovanni Vigna. "The harvester, the botmaster, and the spammer: on the relations between the different actors in the spam landscape." In Proceedings of the 9th ACM symposium on Information, computer and communications security, Pages 353-364. ACM, 2014, Kyoto, Japan.
[160] Haider, Peter, and Tobias Scheffer. "Finding Botnets Using Minimal Graph Clusterings." in Proceedings of the 29 th International Conference on Machine Learning, Pages 847-854, June 26–July 1 2012, Edinburgh, Scotland, UK.
[161] Mao, Ching-Hao, Chang-Cheng Lin, Jia-Yu Tim Pan, Kai-Chi Chang, Christos Faloutsos, and Hahn-Ming Lee. "EigenBot: foiling spamming botnets with matrix algebra." In Proceedings of the ACM SIGKDD Workshop on Intelligence and Security Informatics, Pages 1-8, ACM, August 12, 2012, Beijing, China.

Wazir Zada Khan is currently with Faculty of Computer Science and Information System, Jazan University, Kingdom of Saudi Arabia. He is also a PhD Scholar at Electrical and Electronic Engineering Department, Universiti Teknologi PETRONAS, Malaysia. He received his MS in Computer Science from Comsats Institute of Information Technology, Pakistan. His research interests include network and system security, sensor networks, wireless and ad hoc networks. His subjects of interest include Sensor Networks, Wireless Networks, Network Security and Digital Image Processing, Computer Vision.

Dr. Muhammad Khurram Khan is currently working at the Center of Excellence in Information Assurance, King Saud University, Saudi Arabia. He has edited seven books and proceedings published by Springer-Verlag and IEEE. He has published more than 225 papers in international journals and conferences and he is an inventor of 10 U.S./PCT patents. Dr. Khan is the Editor-in-Chief of a well-reputed journal 'Telecommunication Systems' (Springer). He is also on the editorial boards of several International journals, including the Journal of Network and Computer Applications (Elsevier), IEEE Communications Magazine, IEEE Access, Security and Communication Networks (Wiley), PLOS ONE, Electronic Commerce Research (Springer), and Scientific World Journal. Dr. Khurram is one of the organizing chairs of several top-class international conferences and he is also on the program committee of dozens of conferences. He is a recipient of several national and international awards for his research contributions. In addition, he has been granted several national and international funding projects in the field of Cybersecurity. His current research interests include Cybersecurity, biometrics, multimedia security, and digital authentication. He is a Fellow of the IET, Fellow of the BCS, Fellow of the FTRA, Senior Member of the IEEE, a member of the IEEE Technical Committee on Security & Privacy, and a member of the IEEE Cybersecurity community.

Dr. Fahad T. bin Muhaya, is Associate Professor at Management Information Systems (MIS) Department, Business Administration College at King Saud University, Riyadh, Saudi Arabia. Bin Muhaya has received his B.S. in Information System from Computer Science and Information Systems College at King Saud University in 1986, and his Master degree in Information System from American University at Washington D.C, United States in 2000. In addition, he has obtained the Ph.D., in Information Technology from George Mason University, United States, in 2005. From 2006 to 2008 he was a faculty member at Computer Science College in IMAM University at Riyadh. During that period he was appointed as a Chairman of Information System Department till he left the University. He joined King Saud University in the spring of 2008 as faculty in Management Information Systems Department at the Business School. By the time he was joined KSU, Dr. Fahad co-founded the Center of Excellence in Information Assurance (CoEIA) and was appointed as a vice director of the Center. Year Later, Dr. Fahad was appointed to be the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/COMST.2015.2459015, IEEE Communications Surveys & Tutorials

Director of His Royal Highness Prince Muqrin Chair (PMC) for IT Security, which is the first research Chair in IT Security in the region. Meanwhile, Dr. Fahad was appointed as Chairman of Management Information Systems Department in Business Administration College. After Bin Muhaya has served for two full terms as a chairman, he was appointed to be a dean for College of Applied Studies and Community Services (CASCS) for full term. Dr. Fahad is a part–time Information Security Consultant for several government departments and private companies. Dr. Bin Muhaya is member of several scientific societies and founder and board council members of others. Moreover, Bin Muhaya has published tens of scientific papers in referred conferences and Journals and is working an editor- in-Chief for special edition in referred Journals and as referee for several conferences.

Dr. Mohammed Y Aalsalem is currently Dean Faculty of Computer Science and Information System, Jazan University, Kingdom of Saudi Arabia. He received his PhD in Computer Science from Sydney University. His research interests include real time communication, network security, distributed systems, and wireless systems. In particular, he is currently leading in a research group developing flood warning system using real time sensors He has served as the PC member for many international conferences such as CAINE2011, ICRAMET 2015. He serves as the Guest Reviewer of Journal of Network and Computer Applications (Elsevier, JNCA), King Saud University Journal (CCIS-KSU Journal) and King Abdulaziz City for Science and Technology (KACST).

Han-Chieh Chao is a joint appointed Chair Professor of the Department Computer Science & Information Engineering and Electronic Engineering of National Ilan University (NIU), I-Lan, Taiwan, the Department of Electrical Engineering, National Dong Hwa University, Hualien, Taiwan, and the School of information Science and Engineering, Fujian University of Technology, Fuzhou, China. He is serving as the President since August 2010 for NIU as well. He was the Director of the Computer Center for Ministry of Education Taiwan from September 2008 to July 2010. His research interests include High Speed Networks, Wireless Networks, IPv6 based Networks, Digital Creative Arts, e-Government and Digital Divide. He received his MS and Ph.D. degrees in Electrical Engineering from Purdue University in 1989 and 1993 respectively. He has authored or co-authored 4 books and has published about 400 refereed professional research papers. He has completed more than 100 MSEE thesis students and 4 PhD students. Dr. Chao has been invited frequently to give talks at national and international conferences and research organizations. Dr. Chao is the Editor-in-Chief for Journal of Internet Technology, International Journal of Internet Protocol Technology and International Journal of Ad Hoc and Ubiquitous Computing. Dr. Chao has served as the guest editors for Mobile Networking and Applications (ACM MONET), IEEE JSAC, IEEE Communications Magazine, IEEE Systems Journal, Computer Communications, IEE Proceedings Communications, the Computer Journal, Telecommunication Systems, Wireless Personal Communications, and Wireless Communications & Mobile Computing. Dr. Chao is an IEEE senior member and a Fellow of IET (IEE).