Survey of Peer-to-Peer Botnets and Detection Frameworks

Ramesh Singh Rawat[1,2], Emmanuel S. Pilli[3] and R. C. Joshi[1]
(Corresponding author: Emmanuel S. Pilli)

Department of Computer Science and Engineering, Graphic Era University, Dehradun, India[1]
Department of Computer Science and Engineering, Uttarakhand Technical University, Dehradun, India[2]
Department of Computer Science and Engineering, Malaviya National Institute of Technology, Jaipur, India[3]
(Email: espilli.cse@mnit.ac.in)

(Received Feb. 18, 2017; revised and accepted June 25, 2017)

Abstract

Botnet is a network of compromised computers controlled by the attacker(s) from remote locations via Command and Control (C&C) channels. The botnets are one of the largest global threats to the Internet-based commercial and social world. The decentralized Peer-to-Peer (P2P) botnets have appeared in the recent past and are growing at a faster pace. These P2P botnets are continuously evolving from diverse C&C protocols using hybrid structures and are turning to be more complicated and stealthy. In this paper, we present a comprehensive survey of the evolution, functionalities, modelling and the development life cycle of P2P botnets. Further, we investigate the various P2P botnet detection approaches. Finally, discuss the key research challenges useful for the research initiatives. This paper is useful in understanding the P2P botnets and gives an insight into the usefulness and limitations of the various P2P botnet detection techniques proposed by the researchers. The study will enable the researchers toward proposing the more useful detection techniques.

Keywords: Botnet Architecture; Detection Frameworks; Hybrid Botnets; Peer-to-Peer Botnets; Traffic Analysis

1 Introduction

Botnet is a network of compromised computers that are illicitly controlled and secretly used by attackers for various malicious operations. The attacker controlling the botnet is called bot master or bot herder. The compromised computers in a botnet are called drones or zombies and the malicious software running on them is known as bot. The term "bot" is derived from the word "robot" and it is a program used to automate tasks [8].

Botnets comprise of large pool of thousand to millions of compromised computers which empower the attackers with huge computational power and large band-width to launch attacks at global scale. Botnets are the largest threat to the cyber security of government, industries, academia and critical infrastructure *etc.* [4]. These provide large distributed platform to perform various malicious activities such as distributed denial-of- service (DDoS), spamming, phishing, spying, click-fraud, bitcoin mining, brute force password attacks and compromising social media service [10].

Many papers have surveyed the research literature of botnets [13, 18, 20, 26], but to the best of our knowledge, there is no schematic, analytical & comprehensive survey on the emerging P2P botnets and the detection methods. In this paper, we exclusively discuss various aspects of P2P botnets including: evolution, characteristics, C&C architecture and various detections methods. The paper is useful in understanding the characteristics of P2P botnets and the classification of the various detection techniques. It has the following important contributions:

- The timeline of evolution and development life cycle of P2P botnets is presented;
- The characteristics, architecture and functionalities of P2P botnets are described;
- The taxonomy and analytical survey of the various detection frameworks and models is presented;
- A discussion of the research challenges in detection and defence of these botnets is also included.

The remainder of the paper is organized from Section 2 to Section 7. Section 2 covers the background and related surveys. The botnet architecture and C&C protocols are discussed in Section 3. In Section 4, we have categorized the various detection frameworks. Further, the detection techniques are investigated in Section 5. In Section 6, we have presented the identified research challenges. Finally, Section 7 presents the summary and directions for future research work.

2 Background

Attackers exploit number of vulnerabilities and use social engineering techniques to infect more and more computers, network/IoT devices with the malware to build the botnet [4]. Many robust or stealthy botnets have evolved which wrecked havoc on the cyber security. During the onset of botnets the centralized architecture was popular. This architecture is easy to implement and monitor, but suffers from the limitation of easy detection of the centralized servers. Therefore, attackers adopt the decentralized or hybrid botnet architectures to overcome the limitations of the centralized architecture.

The decentralized P2P or hybrid botnets belong to the third generation. In P2P botnets attackers can directly communicate the commands to any peer bot; who further communicate the commands to other neighboring peers. These botnets have many robust features and stay resilient, *i.e.*, even if a significant part of the botnet is taken down, remaining botnet works under control of bot masters [6, 34]. P2P botnets are the most prevalent in the cyber world since they have many merits in comparison to centralized (IRC or HTTP) botnets.

3 Peer-to-Peer Botnets

Many large P2P botnets have been discovered in the wild. This clearly show a tendency towards the decentralized P2P botnets. These botnets use P2P networks as a vector for infestation and the peer nodes as C&C channels [1]. The P2P botnets form complex overlay networks using either customized or standard P2P protocols [1]. Figure 1 shows the timeline of the P2P botnets' evolution. Table 1 lists the characteristics and applications of the well-known P2P botnets developed over a period of time.

3.1 Botnet Architecture

Botnets are usually characterized on the basis of C&C architecture. The model of a typical botnet can be understood by the analysis of its architecture, C&C mechanism and its development life-cycle. The various phases of the botnets are shown in Figure 2. Cooke *et al.* [8] presented three different botnet communication topologies: centralized (IRC-based and HTTP-based), decentralized (P2P-based) and random. Grizzard *et al.* [13] classified the architecture of decentralized P2P botnets as: structured, unstructured, super-peers and hybrid.

Structured P2P Botnets: Use the overlay structured P2P network and thus employ a globally consistent protocol for efficient routing and search. Storm uses Overnet structured P2P overlay network protocol to build its C&C infrastructure. Most of the P2P botnets are based on structured overlays such as Kademlia. The botnets using public protocols let them mix the C&C traffic with the standards P2P applications.

Unstructured Botnet: Use P2P overlay links established arbitrarily and maintain neighboring peers list to ensure connectivity. This architecture is inherently flexible in the selection of neighboring peers and routing mechanisms. The unstructured botnets are difficult to be crawled and probe, since there is no specific structure that can be exploited.

Superpeer Overlay Botnets: Select some of the globally accessible compromised systems to form the C&C architecture. The compromised peers behind NAT are the normal peer bots and connect to any of the superpeers to pull published commands. Although the design is more scalable, but the superpeers are vulnerable to be detected. Further, the detection and removal of a superpeer does not have any significant impact on the botnet, since communication can be redirected to the new super-peers.

Hybrid P2P Botnets: Overcame the limitations of centralized and decentralized architectures. Many of the real botnets discovered in the wild have multi-layered hybrid P2P architecture. This structure employs top layer bots as master C&C servers. The P2P network serves as relay bots at the second layer communicating with the top servers and the bottom client as peer bots.

3.2 Command and Control Mechanisms

The command and control (C&C) channels are used to command the bots from remote system. The C&C layer forms the multi-tier architecture of the botnets and differentiate them from other malware. The various C&C architectures are based on different C&C protocols including IRC, HTTP, P2P, DNS, Bluetooth, email, social networks and/or other custom protocols [26, 10]. The botnets can select either P2P protocols or non-P2P protocols for C&C communication.

The C&C operations of the botnets are categorized into - **pull** and **push** mechanisms. In pull mechanism botmasters publish commands at certain specific locations for which the peer bots subscribe and actively receive the commands. The bots execute the commands and also forward to other peers in their list [24]. In push mechanism C&C servers push/forward the commands to peer bots for execution. The peer bots passively wait for the commands and also forward the received commands to neighboring peer bots. The bots can receive commands to execute using either a push or pull mechanism.

The C&C activities can be detected by active monitoring of the hosts connecting to the external suspected hosts. Identification of botnet C&C traffic is an important approach to botnet defense, but identification of C&C traffic is difficult since botnets use standard protocols for communications. Moreover, the malicious traffic is similar to legitimate traffic and is fused with the benign P2P communication traffic of the legitimate P2P applications (*eg*, Skype); Trojan.Peacomm botnet and Stormnet are

Table 1: P2P botnets and their characteristics

Botnet Name	**Discovered**	**Architecture [a]/ Protocols**	**Functionalities [b]**	**Comments**
Sinit	Sep. 2003	D/P2P	DD	• Browser exploit, Random scan/probing, • Public key encryption
AgoBot	2003	C/IRC	DD, CS, SP, CF	• Spread using P2P Scanning, Disable AVs, • Uses SSL, Robust, Modular, Flexible
Slapper	2003	D/P2P	DD, CS	• Vulnerability exploit, • Difficult to monitor
Phatbot	Mar. 2004	D/IRC, P2P	DD, SP, MD, CS	• Multi exploit, Polymorphic, • Disable AVs, Not scalable
SpamThru	2006	H/Custom	SP	• Infection via e-mail, Encryption, • Spam templates server, 12,000 bots, 350 m spams/day
Nugache	Apr. 2006	D/Custom	DD	• Use random ports in C&C, E-mail infection, • Encryption, Resilient to take-down
Mega-D	2006	D/Custom	SP	• E-mail attachments, Polymorphic, • 10 billion spams/day, 250,000 infections,
Storm/Peacomm	Jan. 2007	D/P2P-Overnet	DD, SP	• Social engineering, C&C-Fast-flux, • Polymorphic, Hash encrypted, Disable AVs, 85,000-bots, 3 b spams/day
MayDay	Feb. 2008	H/HTTP,P2P	SP, PH	• Random anti-entropy based architecture, Hijacking browser proxy settings, • Encrypted ICMP, Limited C&C traffic, Web proxy,
Waledac	Dec. 2008	D/HTTP, P2P	DD, CS, SP, CS	• E-mail, social engg., Encryption, Packer, • Tunneling, Block DNS look-up, 7000 spams/day, shut down- Mar. 2010
Mariposa/Butterfly	Dec. 2008	D/Custom	DD, CS, SP, MD	• Code injection, Self-propagation, Obfuscation • Anti-debugging, 13,000,000 bots
Conficker C Conficker D, Conficker E,	Feb. 2009, Mar. 2009, Apr. 2009	H/HTTP, P2P	SP, DD	• Exploit-NetBIOS, Block DNS lookups, fast-flux, • Disables AVs & updates, 10,000,000+ bots, 10 b spams/day
Sality (v3, v4)	2009	D/Custom	SP	• Encryption, Resilient, Polymorphic, Disable AVs, • Custom P2P protocol over UDP
Miner	Dec. 2010	H/P2P	DD, MB, MD	• Social engineering, • Conceal C&C servers, Encryption
Kelihos	Dec. 2010	H/P2P-Custom	DD, SP, CS, MB	• Flash-drive, C&C proxies, Hidden-social networks, • 4 billion spams/day, Dismantled in Sep. 2011
ZeroAccess	Jul. 2011	D/P2P-Custom	CF, MB, PH, SP	• Exploit kit, Self healing P2P protocols, • Operation in user-mode, 9 m infections
TDL-4/ TDSS	2011	D/P2P-Kad	MD, CS	• Bootkit- infects MBR, DGA, Encryption, • Removes other malwares, 4.5 m infections
Gameover Zeus	Sept. 2011	H/HTTP, P2P-Kad	CS, SP, PH, DD	• Propagate-spams & phishing, RC4 encryption, • Anti-Crawling Technique, 3.6 m infections
Kelihos.B	Jan. 2012	D/P2P	MB, SB	• Spread via social networks, Encryption, • Sinkholed-March 2012
THOR	Mar. 2012	D/P2P	DD, SP, CS,	• Modules for sale, 256-AES encryption, • 8192-bit RSA instruction signing
Kelihos.C	Apr. 2012	D/P2P		• Stealing Internet browsers passwords, • Sinkholed in 2013
Wordpress/QBot	2013	-	CS, MD	• Crack administrative passwords, • 500,000+ infections, sniffed 800,000 transactions
newGOZ	Jul, 2014	D/DGA	SP,	• DGA generating 1,000 domains per day, • Use spam templates of Cutwail botnet,
Mac OS X botnet	Sep. 2014	-	CS, MD	• Request C&C servers list using MD5 hash of the current date • 17,000 unique IP addresses-Mac hosts

[a] Architecture: C: Centralized, D: Decentralized, H: hybrid,

[b] Functionalities: DD: DDoS, SP: Spamming, CS: Credential Stealing, MD: Malware Distribution, PH: Phishing, CF: Click Fraud, MB: Mining Bitcoins, SB: Stealing Bitcoins

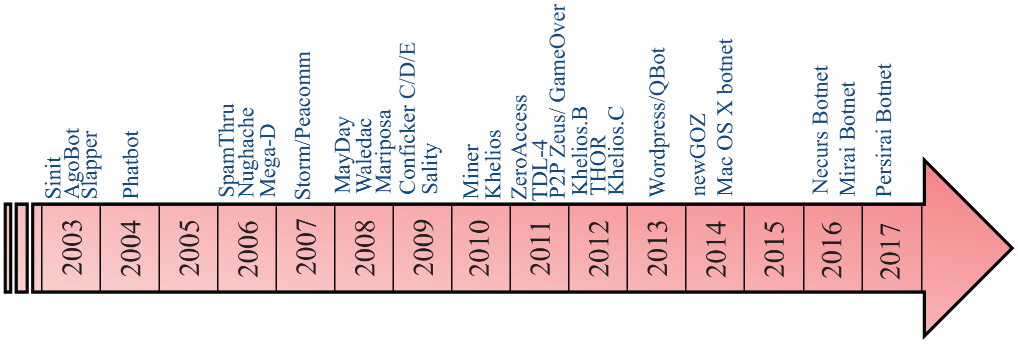

Figure 1: Development timeline of the P2P botnets

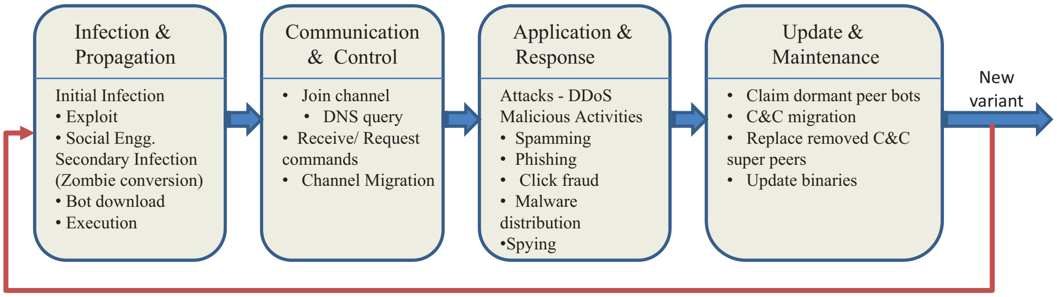

Figure 2: The development lifecycle of the botnets

two such examples [13]. Further, the C&C traffic of the botnets has low volume resulting into lower network latency. This challenges the threshold-based techniques to monitor and detect botnet activities.

4 P2P Botnet Detection Taxonomy

Botnet detection is the important step to combat the these threats. Researchers have analyzed the P2P botnets discovered in the recent past and identified the features which characterize them [13]. This has facilitated to develop various P2P botnet detection and mitigation techniques. Although, the available botnet detection and mitigation techniques have many great thoughts; but, these also suffer from many limitations to combat the real world botnet threats.

In this section, we present the taxonomy of the detection methods. The methods are classified into two major categories: (i) honeypot-based detection and (ii) Intrusion detection system (IDS)-based detection. These are again classified into further subtypes as explained in the subsections. The taxonomy of the detection methods is shown on Figure 3.

4.1 Honeypot-based Detection

A honeypot is a program that appears an attractive service, or an entire operating system to lure an attacker. It is a security resource designed to attract and detect the malicious operations and network attacks [3]. The group of honeypots is known as honeynet and are widely employed by the security communities to log bot activities and monitor the botnets. The logged data is analysed to discover the tools, techniques and motives of the attackers. Further, the obtained information is helpful to design the effective detection and mitigation techniques. This may also help to track the attackers for law enforcement [3].

Many researchers have discussed the use of honeypots/honeynets for botnet detection [8, 12, 26]. Cookie *et al.* [8] presented the use of honeypots to join botnet and monitor the activities. Freiling *et al.* [12] illustrated to prevent the DDoS attacks caused by the use of botnets. Further, honeypots can be deployed to join the botnets and serve as decoy system to provide valuable information about bots behavior and activities. Unfortunately, attackers employ anti-honeypot techniques to prevent any trap by the honeypots.

Honeypots trap the traffic directed to them only and cannot detect the real infected hosts in the enterprise network [3]. Moreover, honeypots need to be monitored to detect any anomalous behavior of the botnet. So, there is a need for developing more advanced honeypots for use

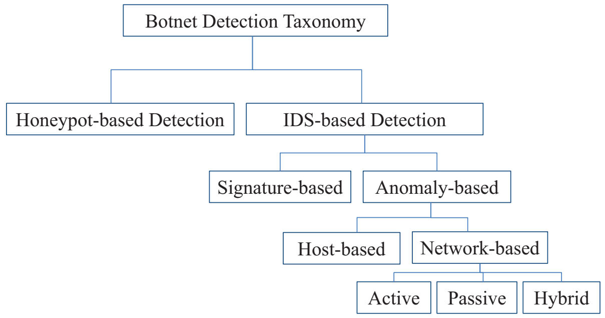

Figure 3: Botnet detection taxonomy

in complement of other botnet detection techniques.

4.2 IDS-based Detection

The IDS-based detection measures rely on the traffic collected at network-level using network sniffing intrusion detection tools and/or the network flow monitors. There are two major types of IDS-based detection techniques: (1) signature-based and (2) anomaly-based. The taxonomy shown in Figure 2.

4.2.1 Signature-based Detection

Signature-based methods use the knowledge of signatures and patterns for accurate botnet detection. The signature-based methods use Deep Packet Inspection (DPI) to inspect the malicious payloads, thus putting too much computational load on the system. These methods only detect the known botnets and are rendered useless by the unknown or even polymorphic botnets. Moreover, the signature-based systems require continuous update of the signature database. Botnet can also employ strong evasion mechanisms such as code obfuscation and encryption to bypass the signature-based detection methods [19, 34]. Therefore, anomaly-based botnet detection methods are proposed to keep up with the changing scenario of polymorphic botnet variants.

4.2.2 Anomaly-based Detection

Anomaly detection refers to finding the patterns that do not conform the normal behavior. These methods are based on the anomalies such as unusual system behavior, high traffic volume, high network latency and traffic on unusual ports. The security professionals and researchers have identified some of the inevitable P2P botnet characteristics such as high connection failure rate, out degree network connection and flow similarity. These characteristics suspect for anomalous behavior and are successfully exploited for the developing detection techniques.

The various anomaly-based botnet detection approaches can be classified into three major categories as: (i) Host-based, (ii) Network-based and (iii) Hybrid approaches.

Host-based Detection: Methods employ system calls monitoring for abnormal activities or data taint analysis techniques for detecting the malicious operations. The system also attempts to access system files to identify the suspected processes [29].

Wurzinger *et al.* [31] presented a host-based anomaly detection system based on aggregate network traffic features. The system inspect packet payloads to search for commonality likely to be C&C instructions from botmaster.

The host-based detection methods mostly inspect the packet payloads to identify a deviation from normal activity; so easily defended using encryption and obfuscation. These systems also suffer from the drawback that they need to be installed on every host. Such detection system demand lot of processing cycles, memory and storage space. Some malware can even disable the anti-malware product on the system it compromised. Therefore, these techniques typically suffer from the performance issues, scalability and effectiveness.

Network-based Detection: The network-based botnet detection methods focus on the network behavior of botnets and detect the C&C traffic between the servers and peers [14]. These detection approaches are generally based on either horizontal correlation (group behavior correlation) or dialog-based correlation (vertical correlation), which mainly utilize either network traffic analysis, aggregating network flows, or network behavior correlation analysis.

The network-based detection methods further employ either active detection or passive detection.

- Active Detection approach participate in the botnet operations. These approaches involve infiltration and C&C server hijack. Such approaches are based on the injection of test packets into the application, server or network for observing the reaction of the network. This produces extra traffic and sometime also violates the privacy.

- Passive Detection approaches silently observe, monitor and analyze the bots for their activities and do not make any efforts to participate in the operation. Passive traffic monitoring includes: behavior-based, DNS-based and data mining detection.

The network-based methods mainly monitor network traffic and can detect known and unknown botnets, but can be evaded by encryption and randomization. Further, these methods utilizing the behavioral characteristics of the bot produce visible network footprints as it works with other peer bots, communicates with botmaster and generates attack traffic. This lends itself exposed to the network-based detection.

Hybrid Detection: The researchers have proposed the host-network cooperative detection methods to overcome the limitations of individual host-level or network-level detection methods [29]. These methods combine the evidences collected from host-based method with the network-based malicious behaviors method [29].

The different P2P botnet detection approaches usually include two steps: 1) hosts with P2P traffic are identified and separated from the normal traffic and then 2) hosts with P2P botnet traffic are detected and filtered from hosts performing only legal action. The next section groups the various detection proposals according to the detection algorithms used in the techniques.

5 P2P Botnet Detection Techniques

The classical botnets detection techniques have three different points to characterize the attacks: (1) command and control (C&C) server, (2) botnet traffic and (3) bot infected computers. Many detection schemes have been proposed to target either one or more of these points, *i.e.*, to detect either individual bots [29], C&C communication traffic and/or C&C server(s) [5]. We classify the various proposals into the following categories (shown in Figure 3):

5.1 Traffic-based Detection

The P2P bots communicate with many other peer bots to push/pull commands, send harvested information and receive updates; thus continuously generating large traffic [5]. Various traffic-based detection techniques have been proposed, which examine the network traffic and focus to observe the traffic patterns.

Noh *et al.* [23] proposed a botnet detection technique based on multi-phased flow model. The method clusters the flows of communication traffic and then build Markov models. The clustering of TCP/UDP connections form the grouping and track packets to determine if they are normal transmissions or flooding attacks. Further, the approach uses an algorithm to construct transitions based matrix of flow modeling and detection engine. But, the method can only detect P2P C&C traffic similar to the traffic used for training. Moreover, malware may avoid detection by using traffic patterns similar to legitimate P2P network.

Another method proposed by Jiang and Shao [16] detect P2P botnets based on the flow dependency in C&C traffic. The method distinguishes the normal P2P application traffic from P2P botnets by assuming that the normal traffic has heterogeneous short time flow dependency. The method also relies on discovering frequent flow dependencies. If these flows rarely happen, the approach may have difficulty to discover the flow dependency. Moreover, the proposed algorithm extracting flow dependency is based on time information and needs large number of samples and the results are based-on limited synthetic P2P botnet traffic trace samples. Further, the method scales quadratically with flow numbers and hence not scalable.

A large-scale wide-area botnet detection system DISCLOSURE identifies groups of features from the Net Flow records [5]. Authors use the group features to distinguish C&C channels from benign traffic. The system is independent of any knowledge of particular C&C protocols and has the ability to perform real-time detection of both known and unknown C&C servers over large datasets.

Zhang *et al.* [34] proposed a system to detect the stealthy P2P botnets. The system exploits the statistical fingerprints and is scalable by parallelized computation. The approach can also be defeated by advanced evasion and obfuscation techniques.

Kheir *et al.* [17] proposed a behavior-based approach called PeerMinor to detect and classify the P2P bots inside corporate networks. The system combines misuse and anomaly-based detection techniques and uses statistical network features: flow size, chunk rate, periodicities and IP distribution. PeerMinor classify P2P signaling flows and use them for the detection. It detects only P2P bots in monitored network and can be challenged by modifying the statistical consistency in the malware P2P flows.

Table 2 presents the characteristics and limitations/challenges of the various detection proposals. These are listed according to the detection techniques used in each proposal.

5.2 Behavior-based Detection

A comprehensive analysis of botnet measurements by Rajab *et al.* [26] reveals the structural and behavioral properties of botnets. Bots may also possess many inherent features, maintain the persistent connections to communicate with other peer bots and receive the commands from botmaster via C&C server(s). It is observed that the network behavior characteristics of P2P botnets are closely tied to the underlying architecture and operation mechanisms [28]. The bots in network immediately execute received commands and show similar communication behavior unlike human behavior. The traffic-based detection techniques mainly analyze the network behavior characteristics.

The botnet detection systems proposed in [7, 11] focus on the network hosts?behavior analysis using Netflows, which avoid the individual packet or host inspection and do not raise the privacy issues.

Felix *et al.* [11] proposed a scheme to detect P2P botnet based on set of group behavior metrics. The metrics are derived from the three standard network traffic characteristics: topological properties, traffic pattern statistics and protocol sequence for identifying hosts which have similar communication patterns. The approach needs multiple bots to be infected in the monitored network. Moreover, the threshold based filtering in the group behavior graph can be evaded by botmasters launching threshold attack.

Shin *et al.* [29] proposed a host-network cooperative behavior-based bot detection framework called EFFORT. The system relies on the client and network characteristics of bots. Although the method is independence of topology and communication protocol, but can be evaded by choosing suitable evasion techniques.

Rodrguez-Gmez *et al.* [27] proposed a method to detect parasite P2P botnets based on resource sharing. The method relies on the assumptions that the bot peers look for popular resources like the command files issued by the botmaster and further share them with other peer bots in a short period of time. The system only focuses on the parasite P2P botnets?detection in the command communication stage by looking and sharing for popular resources. No, real network attack and any other malicious activity can be detected.

5.3 DNS-based Detection

The bots possess a group activity as a key feature and frequently use DNS to rally C&C servers, launch attacks and update their codes. Bots of same botnet contact the same domain periodically leading to similar DNS traffic which is distinct from legitimate users [7]. In this section, we describe and evaluate the DNS-based anomaly detection techniques.

Choi and Lee [7] proposed an online unsupervised botnet detection technique called BotGAD (Botnet Group Activity Detector). BotGAD is implemented based on DNS traffic similarity and its performance is measured using real-life network traces. Botnets can evade the detection methods by performing DNS queries only once in the lifecycle. Hence, the method can detect only the botnets that perform group activities in DNS traffic. The method can also be evaded by botnets employing multi-C&C servers to separates their domain names.

5.4 Graph-based Detection

The graphical structure is an inherent feature of the botnets and is useful to understand how botnets communicate internally. The graphical analysis of the botnet communication network can be used to find the characteristic patterns of the botnets. The P2P C&C communications graph exhibit the topological features useful for traffic classification and botnet detection.

Ha *et al.* [15] analyzed the structural characteristics of Kademlia-based P2P botnets from a graph-theoretical perspective. The study analyzed the scaling, clustering, reachability and various centrality properties of P2P botnets. The authors also discovered that P2P mechanism helps botnets to hide their communication effectively.

Nagaraja *et al.* [22] proposed BotGrep to detect P2P botnets using the analysis of network flows collected over multiple large networks (eg, ISP networks). BotGrep first identifies groups of hosts that form a P2P network in the global view of Internet traffic. The algorithm is based on the premise that many recent botnets use efficient P2P protocols such as Kademlia for implementing C&C communications. But, BotGrep requires additional information to bootstrap the detection algorithm. Further, acquiring global view of Internet communications to bootstrap the detection algorithm may be very challenging.

Venkatesh *et al.* [30] proposed BotSpot to detect the hosts that are part of the structured P2P botnets. The authors developed algorithms based on the differences in the assortativity and density properties of the structured P2P botnets. BotSpot is based on the analysis of the IP-IP graph. It is complementary to the traffic classification approaches that differentiate between the structured P2P botnets and the legitimate structured P2P applications.

5.5 Data Mining-based Detection

The data mining techniques can be used to detect an anomaly *i.e.*, the unusual or fraudulent behavior. Data mining techniques are used for malicious code detection and intrusion detection. Many authors has used classification and clustering techniques to efficiently detect botnet C&C traffic.

The network traffic is a continuous flow of data stream produced at fast rate, which is not practical to be stored and analyzed entirely. Moreover, botnet codes, features and commands are updated frequently leading to dynamic and temporal behavior. Masud *et al.* [21] proposed stream data classification technique to detect P2P botnets. The authors proposed multi-chunk multi-level ensemble classifier to classify concept-drifting streams data. The approach is tested on limited synthetic data and Nugache botnet data collected in small setup, therefore, does not represent the real characterization of numerous botnets in the malware zoo.

Dietrich *et al.* [9] proposed CoCoSpot to detect botnet C&C channels based on traffic analysis. The system uses the periodicity of messages to suspect for C&C operations. Then, it classifies the P2P applications running on the host as malicious or benign based on payload analysis. The approach will result with false negatives if several messages are sent only in one direction.

Rahbarinia *et al.* [25] proposed a system called PeerRush to detect the unwanted P2P traffic. It creates pplication profile?based on the network flow features and inter-packet delays. The system can be evaded by the introduction of noise (random padding and false packets) in communication of bots. Further, the system cannot have much accurate results with the polymorphic and ever evolving P2P botnets.

5.6 Soft Computing-based Detection

Saad *et al.* [28] proposed an technique to detect P2P botnets using network traffic behavior analysis. The authors tested five different machine learning techniques to check for adaptability, novelty and early detection and get promising results using the limited test dataset. The technique is useful only for detection of P2P bots. Further, the

Table 2: Strengths and limitations of detection proposals

Detection Systems	Characteristics	Limitations or Challenges
—Traffic Analysis-based Techniques		
Noh *et al.* [23]	– Behavior & Traffic Analysis, Multi-phased flows model – C&C Traffic detection	– Evasion by using a legitimate P2P network
Jiang and Shao [16]	– Flow dependencies, Independent of malicious traffic,	– High false +ve, Dependency discovery
DISCLOSURE [5]	– Structure & protocol independent, Pattern based features, – Real-time & large scale	– Higher false positives
Zhang *et al.* [34]	– Statistical fingerprints-profile P2P traffic, – Persistent peer clients-similar traffic	– Evasion by blended peer bots and randomization, – Evasion- traffic tunneling through Tor network,
Kheir *et al.* [17]	– Based on P2P Signaling Flow, – Statistical features: Flow size, Chunk rate, periodicities and IP distribution	– Detect P2P bots only in a monitored network,
Behavior Analysis-based Techniques		
Felix *et al.* [11]	– Exploit Traffic pattern, – Bots group behavior	– Multiple bots dependency, Vulnerable to threshold attack
EFFORT [29]	– Host-network cooperation, Independent of topology & protocol, – Resilient to encryption & obfuscation,	– Evasion by bots: using benign domains
Rodrguez-Gmez *et al.* [27]	– Temporal resource sharing model – Monitoring resource sharing behavior	– Used only for parasite P2P botnets, – Source should be popular & short life
DNS-based Techniques		
Choi and Lee [7]	– Group Activity Detector, Online unsupervised known, Unknown, – Scalable, Real-time	– Requires multiple bots
Graph Analysis-based Techniques		
Ha *et al.* [15]	– Reachability & centrality properties – C&C channels detection, Monitoring bot activities	– Vulnerable to random delay, – P2P protocols dependency, False negatives
BotGrep [22]	– C&C patterns in overlay topology – Large-scale, Clustering techniques	– Bootstrap information required
Data Mining-based Techniques		
Masud *et al.* [21]	– Mining Concept-Drifting Data Stream – Packet features are extracted and aggregated into Flow characteristics	– Requires monitoring traffic at each host – Sampling may miss useful communications patterns
CoCoSpot [9]	– Analysis of traffic features – Fingerprint botnet C&C channels	– Evasion by random message padding – Dependency on the dialog-like pattern
PeerRush [25]	– Created application profile from known P2P applications – Based on high-level statistical traffic features	– Deals with the signaling flows as a whole – Evasion by randomization of inter-packet delays
Soft Computing-based Techniques		
Saad *et al.* [28]	– Traffic behavior, Detection in C&C phase – Detection rate 98%	– Dependency on features selection – High computational requirement
Zhao *et al.* [33]	– Anomalous Network traffic, – Real-time detection in C&C phase & attack phase	– Sampling can skip botnet flows, – Vulnerable to obfuscation
General Techniques		
BotMiner [14]	– Anomaly-based-behavior, Traffic-based analysis – Independent to protocol and C&C structure, Real-time	– Detect only active bot(s) – Targets enterprise network only
Wurzinger *et al.* [31]	– Network traffic, Bot behavior, Detect Bots, – No prior information required	– Threshold attack – Content analysis required
PeerPress [32]	– Remote control process- analysis, – Active-informed probing, Fast, Scalable, Real time	– False positives- advanced encryption, – Delayed port binding

technique also needs novel machine learning techniques for more effective results and general botnet detection.

Zhao *et al.* [33] proposed a technique to identify P2P botnet activities. The authors examine the characteristics of the traffic flows in small time windows to achieve the real time detection. But, the reduced time interval to monitor the traffic can skip some flows related to botnets. Further, botnets can also complicate the network flow behavior of the bots and evade detection.

Alauthaman *et al.* [2] proposed a technique using classification and regression tree algorithm and neural network (CART-NN) to detect the P2P bot connections. The technique use the connection-based features extracted from the TCP control packet headers. The method assumes that bots communicate using TCP connections and hence unable to cover the botnet using UDP connections.?

Chen *et al.* [6] proposed a botnet detection framework based on supervised machine learning technique. The framework use the conversation-based features extracted by random forest-based learning. The paper results explain the conversation-based features are better than flow-features, further, random forest is better than other classification algorithms giving the results upto 93.6%.

5.7 Generic Frameworks

A number of general botnet detection frameworks have been proposed based on behavior monitoring and traffic correlation analysis. BotMiner is a general framework for botnet detection [14]. The system detect botnets based on network packets and flow analysis. It relies on behavior monitoring and traffic correlation analysis that is mostly applicable at a small scale and does not scale well, because it requires analysis of vast amounts of fine-grained information. In addition, if there are only small numbers

Table 3: Evaluation of the P2P detection proposals

Detection Proposal	Detection Methodology [a]			Detection Stage [b]			Detection type (KN/UK/B) [c]	Real Time	Scalability [d]
	SB	HB	NB	IP	CC	AT			
BotMiner [14]	✓	✓	✓	✓	✓	✓	B	✓	S/M
Wurzinger *et al.* [31]	✓	✓			✓		KN		S
Ha et el. [15]	✓	✓	✓		✓	✓	KN		S
BotGrep [22]				✓	✓	✓	B		M/L
Saad *et al.* [28]			✓		✓		B	-	M
Choi and Lee [7]		✓	✓		✓	✓	B	✓	M, L
Jiang and Shao [16]			✓		✓		B	-	M
DISCLOSURE [5]			✓		✓		B		L
Felix *et al.* [11]		✓	✓	✓	✓	✓	B		S
Zhao *et al.* [33]			✓		✓	✓	KN		M
PeerPress [32]		✓	✓	✓	✓	✓	B		M, L
EFFORT [29]		✓	✓		✓	✓	B		S, M
PeerRush [25]			✓	✓	✓		KN		S
Rodrguez-Gmez *et al.* [27]			✓	✓	✓		UN	✓	L
Kheir *et al.* [17]			✓	✓	✓		B	✓	S

[a] Detection methodology- SB: Signature-based, HB: Host-based anomaly, NB: Network-based anomaly

[b] Detection phase- IP: Infection/Propagation, CC: Command & Control, AT: Attack

[c] Detection type- KN: Known, UN: Unknown, B: Both

[d] Scalability- S: Small, M: Medium, L: Large

of bots instances in the edge network, it leads to failure of bot coordination and resulting in false negatives. Moreover, BotMiner requires the malicious activities to be visible, thus cannot detect botnets in an early stage and does not work in real-time.

Wurzinger *et al.* [31] proposed a general system to detect bots. The system relies on the characteristics that each bot receives commands and responds in a specific way. It examines the packet payloads to find commonality to be supposed as commands from botmaster. The work complements the existing network-based IDSs by automatically generating the inputs needed by these systems to detect infected machines. But, the approach can be rendered useless by simple encryption in the C&C communication as used by the advanced botnets.

PeerPress is a P2P malware detection framework based on dynamic binary analysis and network level informed probe [32]. First it extracts built-in remotely accessible mechanisms of botnets called Malware Control Birthmarks (MCB) and then performs informed probe at network-level. The framework relies on malware opening service port for communications and provide malicious binary download services on the new infected machines.

The various frameworks use different data-sets of either synthetic or real botnets for evaluation and testing; therefore, the comparative analysis of the techniques is difficult. Table 3 presents the analysis of the promising detection frameworks.

6 Observations and Challenges

Many P2P botnet detection approaches evolved significantly, but many open problems still exist. We have identified the following open problems and research challenges useful for future research:

- The P2P botnets easily evade the port and protocol based detections by using both P2P and non-P2P ports for covert C&C communication. Further, bots traffic may blend with legitimate P2P application traffic.
- The P2P botnets also use fast-flux techniques for anomalous communications and hide the C&C hosts. Further botnets also use Domain Generation Algorithms (DGA) to keep the identity of C&C servers anomalous.
- Botnets also randomize the behavior and stay hidden under the radar to continue the malicious operations. This challenges the early stage detection if few bots exit in monitored network.
- The real-time online botnet detection needs to process huge amount of network traffic streams. Therefore, it is a challenge to develop the fast stream mining and classification algorithms for network traffic analysis to infiltrate the botnet traffic.
- The proliferation of smart-phones and other mobile devices with fast internet access provide new platforms the attackers to build mobile botnets.
- The fast adoption of Cloud computing is likely to attract the development of cloud botnets and is expected to be a big challenge for the security of the cloud computing.

The continuous advances in the botnet technology have enabled the attackers to evade the various detection measures. The exhaustive practices and covert network of the attackers enable them to stay ahead in pursuit of their malicious operations.

7 Conclusions and Future Work

In this paper, we have presented a comprehensive survey of various aspects of P2P botnets. Although the detection techniques have some strengths and scope, but, no single technique can detect such evolving botnets. Further, most detection schemes rely on the offline clustering and classification and does not cope-up the requirements of real time detection. Therefore, there is a requirement to develop a real time clustering and classification of the botnet traffic and on-the-fly mining of the botnet traffic to meet the requirements of real time botnet detections.

There is also a requirement to broaden the scope of detection and cover multiple botnet perspectives and also develop a collaborative detection framework. In our future research work, we would create a model to analyze the latest botnet(s) and develop a generic framework for the detection and mitigation of botnets. Further, extend the model to address the issues of mobile, cloud, social network-based botnets.

References

[1] D. Andriesse, C. Rossow, B. Stone-Gross, D. Plohmann and H. Bos, "Highly resilient peer-to-peer botnets are here: An analysis of gameover zeus," in *8th IEEE International Conference on Malicious and Unwanted Software (MALWARE'13)*, pp. 116–123, 2013.

[2] M. Alauthman, A. Nauman, L. Zhang, R. Alasem and A. Hossain, "A P2P botnet detection scheme based on decision tree and adaptive multi-layer neural networks," *Neural Computing and Applications*, pp. 114, 2016.

[3] P. Bacher, T. Holz, M. Kotter and G. Wicherski. "Know your enemy: Tracking botnets," *Technical Report, The Honeynet Project*, 2005.

[4] E. Bertino and N. Islam, "Botnets and internet of things security," *Computer- IEEE Computer Society*, vol. 50, no. 2, pp. 76–79, 2017.

[5] L. Bilge, D. Balzarotti, W. Robertson, E. Kirda, and C. Kruegel, "Disclosure: Detecting botnet command and control servers through large-scale netflow analysis," in *Annual Computer Security Applications Conference (ACSAC12)*, pp. 129-138, 2012.

[6] R. Chen, W. Niu, X. Zhang, Z. Zhuoand and F. Lv, "An effective conversation-based botnet detection method," *Mathematical Problems in Engineering*, vol. 2017, pp. 9, 2017.

[7] H. Choi and H. Lee, "Identifying botnets by capturing group activities in dns traffic," *Computer Networks*, vol. 56, no. 1, pp. 20–33, 2012.

[8] E. Cooke, F. Jahanian, and D. McPherson, "The zombie roundup: Understanding, detecting and disrupting botnets," in *Proceedings of the USENIX Workshop on Steps to Reducing Unwanted Traffic on the Internet (SRUTI '05)*, pp. 39–44, 2005.

[9] C. J. Dietrich, C. Rossow and N. Pohlmann, "Cocospot: Clustering and recognizing botnet command and control channels using traffic analysis," *Computer Networks*, vol. 57, no. 2, pp. 475–486, 2013.

[10] J. Echeverria and S. Zhou, "Thestar wars' botnet with¿ 350k twitter bots," *ArXiv Preprint ArXiv:1701.02405*, 2017.

[11] J. Felix, C. Joseph, and A. A. Ghorbani, "Group behavior metrics for P2P botnet detection," in *Proceedings of the 14th International Conference on Information and Communications Security (ICICS'12)*, pp. 93–104, Jan. 2012.

[12] F. C. Freiling, T. Holz, and G. Wicherski, "Botnet tracking: Exploring a root-cause methodology to prevent distributed denial-of-service attacks," in *Proceedings of 10th European Symposium on Research in Computer Security (ESORICS'05)*, pp. 319-335, 2005.

[13] J. B. Grizzard, V. Sharma, C. Nunnery, B. B. Kang, and D. Dagon, "Peer-to-peer botnets: Overview and case study," in *Proceedings of First Workshop on Hot Topics in Understanding Botnets (HotBots'07)*, pp. 1–8, Apr. 2007.

[14] G. Gu, R. Perdisci, J. Zhang and W. Lee, "Botminer: Clustering analysis of network traffic for protocol- and structure-independent botnet detection," in *Proceedings of the 17th USENIX Security Symposium (Security'08)*, pp. 139–154, 2008.

[15] D. T. Ha, G. Yan, S. Eidenbenz, and H. Q. Ngo, "On the effectiveness of structural detection and defense against P2P-based botnets," in *Proceedings of 39th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN'09)*, 2009.

[16] H. Jiang and X. Shao, "Detecting P2P botnets by discovering flow dependency in C&C traffic," *Peer-to-Peer Networking and Applications*, pp. 1–12, 2012.

[17] N. Kheir, X. Han and C. Wolley, "Behavioral fine-grained detection and classification of P2P bots," *Journal of Computer Virology and Hacking Techniques*, pp. 1–17, 2014.

[18] C. Y. Liu, C. H. Peng and I. C. Lin, "A survey of botnet architecture and batnet detection techniques," *International Journal of Network Security*, vol. 16, no. 2, pp. 81–89, 2014.

[19] Y. D. Lin, Y. T. Chiang, Y. S. Wu and Y. C. Lai, "Automatic analysis and classification of obfuscated bot binaries," *International Journal of Network Security*, vol. 16, no. 6, pp. 477–486, 2014.

[20] M. Mahmoud, M. Nir and A. Matrawy, "A survey on botnet architectures, detection and defences.," *International Journal of Network Security*, vol. 17, no. 3, pp. 264–281, 2015.

[21] M. M. Masud, J. Gao, L. Khan, J. Han and B. Thuraisingham, "Mining concept-drifting data stream to detect peer to peer botnet traffic," *Technical Report, University of Texas at Dallas, Richardson, Texas, Technical Report UTDCS-05-08*, 2008.

[22] S. Nagaraja, P. Mittal, C. Y. Hong, M. Caesar and N. Borisov, "Botgrep: Finding P2P bots with structured graph analysis," in *Proceedings of 19th USENIX Security Symposium*, pp. 95–110, 2010.

[23] S. K. Noh, J. H. Oh, J. S. Lee, B. N. Noh and H. C. Jeong, "Detecting P2P botnets using a multiphased flow model," in *Proceedings of the Third International Conference on Digital Society (ICDS'09)*, pp. 247–253, 2009.

[24] P. Porras, H. Saidi, and V. Yegneswaran, "A multiperspective analysis of the storm (peacomm) worm," *Computer Science Laboratory, SRI International, Technical Report*, 2007.

[25] B. Rahbarinia, R. Perdisci, A. Lanzi and K. Li, "Peerrush: Mining for unwanted P2P traffic," *Journal of Information Security and Applications*, vol. 19, no. 3, pp. 194–208, 2014.

[26] M. A. Rajab, J. Zarfoss, F. Monrose and A. Terzis, "A multifaceted approach to understanding the botnet phenomenon," in *Proceedings of the 6th ACM SIGCOMM Conference on Internet Measurement (IMC'06)*, pp. 41–52, 2006.

[27] R. A. Rodrguez-Gmez, G. Maci-Fernndez, P. Garca-Teodoro, M. Steiner and D. Balzarotti, "Resource monitoring for the detection of parasite P2P botnets," *Computer Networks*, vol. 70, pp. 30–311, 2014.

[28] S. Saad, I. Traore, A. Ghorbani, B. Sayed, D. Zhao, W. Lu, J. Felix and P. Hakimian, "Detecting P2P botnets through network behavior analysis and machine learning," in *Ninth Annual International Conference on Privacy, Security and Trust (PST'11)*, pp. 174–180, 2011.

[29] S. Shin, Z. Xu and G. Gu, "Effort: A new host-network cooperated framework for efficient and effective bot malware detection," *Computer Networks*, vol. 57, no. 13, pp. 2628-2642, 2013.

[30] B. Venkatesh, S. H. Choudhury, S. Nagaraja and N. Balakrishnan, "Botspot: Fast graph based identification of structured P2P bots," *Journal of Computer Virology and Hacking Techniques*, vol. 11, no. 4, pp. 247–261, 2015.

[31] P. Wurzinger, L. Bilge, T. Holz, J. Goebel, C. Kruegel, E. Kirda, M. Backes, and P. Ning, "Automatically generating models for botnet detection," in *Proceedings of the 14th European conference on Research in computer security (ESORICS09)*, pp. 232–249, 2009.

[32] Z. Xu, L. Chen, G. Gu and C. Kruegel, "Peerpress: Utilizing enemies P2P strength against them," in *Proceedings of 19th ACM Conference on Computer and Communications Security (CCS'12)*, pp. 581-592, 2012.

[33] D. Zhao, I. Traore, A. Ghorbani, B. Sayed, S. Saad and W. Lu. "Peer to peer botnet detection based on flow intervals," in *Information Security and Privacy Research*, vol. 376, pp. 87–102, 2012.

[34] J. Zhang, R. Perdisci, W. Lee, X. Luo and U. Sarfraz, "Building a scalable system for stealthy P2P-botnet detection," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 1, pp. 27–38, 2014.

Biography

Mr. Ramesh Singh Rawat received his M. Tech degree in Computer Science and Engineering from Graphic Era University in Dehradun, India in 2010. Afterwards, he is serving as Assistant Professor in Department of Computer Science at Graphic Era University. He is working toward Doctoral degree in Computer Science & Engineering from Uttarakhand Technical University, Dehradun, India. He is a member of IEEE, ACM CSTA, UACEE and CSI India. His research interests include Internet Security and Privacy, IDS and Botnet Defense.

Dr. Emmanuel S. Pilli received his M. Tech (Computer Science) from BIT Ranchi in 2001 and Ph. D (Computer Science) from the Indian Institute of Technology, Roorkee in 2012. He has over 20 years of teaching and research experience and is presently an Assistant Professor in the Department of Computer Science of Engineering, Malaviya National Institute of Technology Jaipur, India. His areas of interest include Security and Privacy, Forensics, Cloud Computing, Big Data and IoT. Dr. Pilli was awarded the ISEA Fellowship in 2011 by the Department of Information Technology, Govt. of India, for his research in Information Security. He is a Senior Member of IEEE and an active participant in professional activities of ACM, CCICI, CSA and CSI.

Prof. R. C. Joshi received M. E., Ph. D from University of Roorkee (now IIT Roorkee) in 1970 & 1980 respectively. He has over 45 years of teaching and research experience. He is presently Chancellor of Graphic Era University Dehradun, India. He was formerly Professor & Head, Electronics & Computer Engineering Department at IIT Roorkee, India. Prof. Joshi has guided 27 Ph.Ds and more mthan 250 M.Tech. dissertations. He has worked as a Principal Investigator in number of Sponsored Projects of Ministry of ICT, DRDO, AICTE, UNDP, ISEA *etc.* He has written 8 Books/Book Chapters and several technical reports. Prof. Joshi has published more than 250 Research Papers in International Journals / Conferences. He was awarded Gold Medal by Institution of India for Best Research Paper.