SECURITY AND COMMUNICATION NETWORKS
Security Comm. Networks 2012; **5**:1094–1101
Published online 6 February 2012 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/sec.431

SPECIAL ISSUE PAPER

Detection of botnets before activation: an enhanced honeypot system for intentional infection and behavioral observation of malware

Young Hoon Moon[1], Eunjin Kim[2], Suh Mahn Hur[3] and Huy Kang Kim[1]*

[1] Center for Information Security Technologies (CIST), Graduate School of Information Security, Korea University, Seoul, South Korea
[2] Kyonggi University, Suwon, Gyunggi-do, South Korea
[3] Solution Operation Team, Saint Security Co, Ltd., Seoul, South Korea

ABSTRACT

As botnets have become the primary means for cyber attacks, how to detect botnets becomes an important issue for researchers and practitioners. In this study, we introduce a system that is designed to detect botnets prior to their activation. Pre-detection of botnets becomes available with our enhanced honeypot system that allows us to intentionally infect virtual machines in honeynets. For empirical testing, we applied our system to a major Internet service provider in Korea. After running our proposed system for 12 months, it was found that nearly 40% of blacklisted botnets were pre-detected by our system before their attacks begin. We expect that our system can be used to detect command-and-control servers and to screen them out during their propagation stage before they make harmful attacks. Copyright © 2012 John Wiley & Sons, Ltd.

KEYWORDS
botnet detection; malware; honeynets; intentional infection; behavioral analysis

***Correspondence**
Huy Kang Kim, Center for Information Security Technologies (CIST), Graduate School of Information Security, Korea University, Seoul, South Korea.
E-mail: cenda@korea.ac.kr

1. INTRODUCTION

Nowadays, botnets have become the primary means for attackers. The most promising value of botnets that lures attackers is the ability to provide anonymity to attackers through the use of a multi-tier command-and-control (C&C) architecture [1]. The attacks made through botnets are also known to be hardly tracable because individual bots are not physically owned by attackers and are located in several places [1,2]. Moreover, for formation and operation of botnets, attackers are not required to have high-level knowledge or skills regarding network or operating systems (OSs). For these reasons, botnets are widely used by attackers and now play a very important role in the Internet malware epidemic [2]. Well-known attacks made through botnets include distributed denial-of-service (DDoS), personal information theft, e-mail spam, and phishing [2].

While botnets are used for various attacks aforementioned, they are often used for DDoS attacks. DDoS attacks are known to disable the network services of a victim system by consuming its bandwidth [2]. Hence, they are very critical to Internet service providers (ISPs) who provide network services to their individual subscribers and enterprise users who lease network lines over converged network services [3]. Such damaging attacks have shifted the role of ISPs from simply providing network service to actively protecting themselves and their customers from the attacks [4–6]. To minimize the damage incurred by such attacks, ISPs are adopting various techniques for botnet detection.

As it is known, there exist several different methods for identifying and tracing botnets. These include the approach with honeynet, Internet Relay Chat (IRC)-based detection, and Domain Name System (DNS) tracking [2]. However, these botnet detection methods based on the aforementioned approaches can be applied only after attacks begin or only after significant symptoms occur; the timing of which is right before the attacks begin. Our study differs from previous researches on botnet detectection methods in that we focus on developing the system that detects botnets prior to their activiation. In this study, we propose an enhanced honeypot system that allows us to intentionally infect virtual machines in honeynets and to pre-detect botnets. For empirical testing, we applied our system to a major ISP in Korea. After running our proposed system for 12 months, it was found that among the C&C server IPs and URLs, which were identified and known in public, almost 40% of them were pre-detected by our system during their propagation stage.

This paper is organized as follows. In Section 2, related work is presented. We present our propsed system in Section 3. Section 4 presents our experiment results. We conclude our research in Section 5 and present directions for future work in Section 6.

2. RELATED WORK

2.1. Botnet detection

Botnet detection has been a major research topic in recent years, and several different approaches have been proposed by researchers. Major approaches are summarized as follows, which include the approach with honeynet, IRC-based detection, signature-based detection, anomaly-based detection, and DNS tracking.

2.1.1. Honeynet

Honeynet approach is well known by its strong ability to detect security threats, to collect malwares, and to understand the behaviors and motivations of attackers [2]. A honeypot, which comprises honeynet, is a system that lures attackers by pretending to have security vulnerabilities. That is, it is a system that awaits to be attacked. Honeynet consists of more than one honeypot on a network for large-scale network monitoring. Honeypots are classified into two different types: low-interaction honeypot (LIH) and high-interaction honeypot (HIH). LIH limits levels of interaction with attackers, commonly through simulation of network services or operation systems [7]. This makes LIH easily identifiable by human attackers; hence, LIH often only lures automated attacks [7]. An example of LIH is honeyd [8]. HIH, on the other hand, uses real systems for making interaction with attackers; hence, the levels of interaction is not limited [7]. However, more risks are involved when deploying HIH because an attacker can get complete control over the honeypot and can abuse it [7]. Levine *et al.* used HIH for collecting and analyzing rootkits manually [9]. Beyond these studies, many studies discussed how to use honeypot for botnet tracking and measurement [2]. Our proposed system is based on honeynet approach and is designed to overcome the weakness of both LIH and HIH. Besides, whereas previously developed honeynet-based systems passively await to be attacked, our system is designed to be active in malware collection and infection.

2.1.2. Internet Relay Chat-based detection

Internet Relay Chat-based botnet has long been studied, and therefore, several characteristics have been discovered for detection [2]. One of the well-known IRC-based detection approaches is sniffing the traffic on common IRC ports and checking the payloads [10]. However, to avoid being detected by such method, botnets can use random ports to communicate. Therefore, researchers proposed other approaches such as looking for behavioral characteristics of bots. Racine [11] found that IRC-based bots often remain idle and make response only when receiving a specific instruction. Thus, the network traffic that shows such features can be assumed to be made by IRC-based bots. However, such approach is known to show a high false positive rate. Another approach proposed by Rajab *et al.* [12] is using a modified IRC client called IRC tracker. Beyond these methods, many effective methods for IRC-based detection have been proposed. However, such IRC-based detection methods cannot be applied to the most of recent botnets, which are based on Hypertext Transfer Protocol (HTTP) or peer-to-peer (P2P).

2.1.3. Signature-based detection

The basic idea of signature-based detection is to extract feature information from the network traffic and to match the patterns with the ones registered in the knowledge base of existing bots [2]. Although it is easy to apply such detection method, such method is known to have several drawbacks [13]. To make such method to be more effective in detection, the knowledge base should be always updated with new signatures [13]. Even with frequently updating the knowledge base, it has limitation in detecting the new unknown bots [13].

2.1.4. Anomaly-based detection

Binkley and Singh [14] proposed an algorithm for anomaly-based detection. The algorithm observes and records a large number of Transmission Control Protocol (TCP) packets with respect to IRC hosts. It then computes the ratio of the total amount of TCP control packets over the total number of TCP packets. On basis of the ratio, called TCP work weight, it detects some anomaly activities [14]. High value of the ratio is assumed to imply a potential attack [14]. However, such anomaly-based detection mechanism might not work when the IRC commands are encoded [14].

2.1.5. Domain Name System tracking

Botnets are known to send DNS queries to make access with the C&C servers. Hence, if their domain names can be interpreted, the botnet traffic can be captured [15,16]. With the difference between DNS queries made by botnets and legitimate hosts, Choi *et al.* [15] developed an algorithm for identification of botnet DNS queries [15]. Beyond this, many other botnet detection methods are proposed, which are based on DNS tracking approach. However, there exists limitation in these methods. Nowadays, to avoid being detected by these methods, attackers frequently change the domain name of C&C servers. Besides, they let the bots send DNS queries not to the DNS servers owned by ISPs but to DNS servers owned by attackers. To further avoid the risk to be detected, they change the domain name of their DNS servers frequently and limit the number of queries sent to the servers.

2.2. Botnet life cycle

A typical life cycle of botnets is depicted in Figure 1. At first, the botnet malware is created by a bot author. Botnets are then formed and await instructions from the C&C servers through the following three stages: stage 1, recruiting bot members; stage 2, forming the botnets; and stage 3, standing by for instructions [17]. To recruit bot members, an attacker (sometimes, an attacker is not a bot author) spreads the botnet malware to infect vulnerable computer systems. For spreading the botnet malware, attackers may implant the malware in popular Web sites. In addition, the infected computers also spread the malware to other computer systems. After recruiting enough bot members, the attacker chooses some bots among them for formation of botnets. That is, the attacker can form multiple botnets with those recruited bot members. After a botnet is formed, all members of the botnet await orders from the C&C server as programmed. In this stage, bots make minimized communications with the C&C server to hide themselves.

Once attack commands and target information are delivered by the C&C server, the botnet starts to attack targets until it is blocked or detected by ISPs or antivirus (AV) companies. The general countermeasures for attacks include cleaning up infected PCs with AV program, blocking the C&C server's IP address, and shunting DDoS traffic with DNS sinkhole routing.

Previously proposed botnet detection algorithms or methods can be applied after botnets launch attacks or at

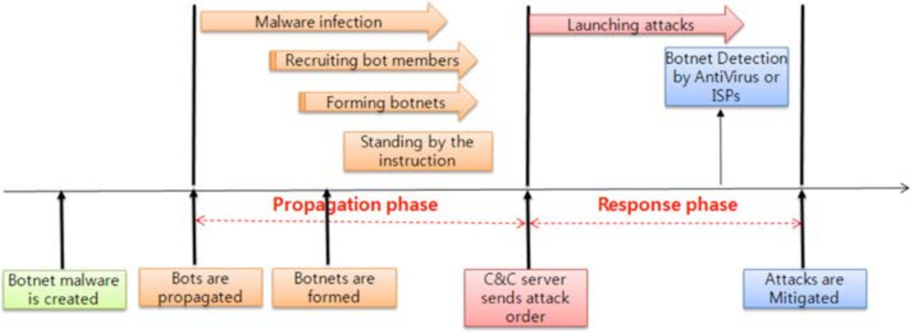
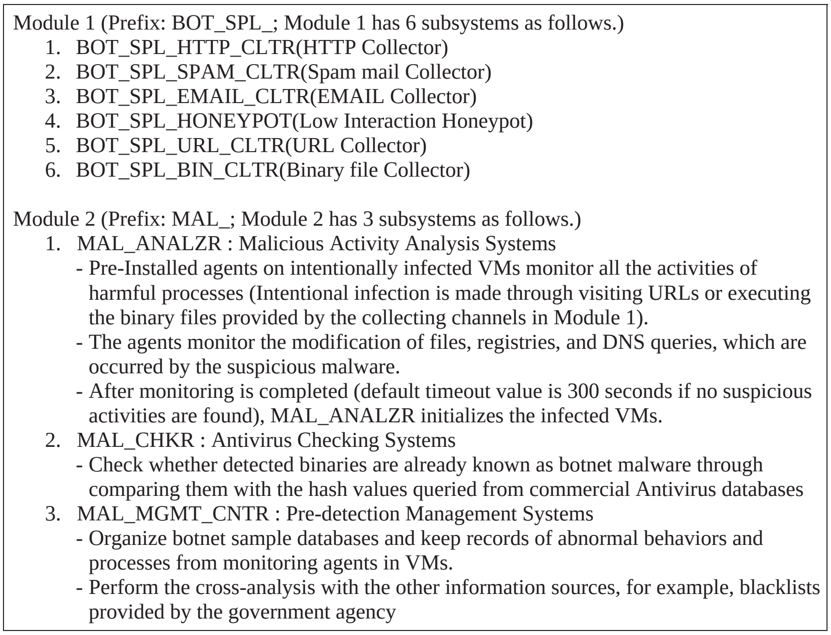


Figure 1. The life cycle of botnets. ISP, Internet service provider; C&C, command and control.


Module 1 (Prefix: BOT_SPL_; Module 1 has 6 subsystems as follows.)

1. BOT_SPL_HTTP_CLTR(HTTP Collector)
2. BOT_SPL_SPAM_CLTR(Spam mail Collector)
3. BOT_SPL_EMAIL_CLTR(EMAIL Collector)
4. BOT_SPL_HONEYPOT(Low Interaction Honeypot)
5. BOT_SPL_URL_CLTR(URL Collector)
6. BOT_SPL_BIN_CLTR(Binary file Collector)

Module 2 (Prefix: MAL_; Module 2 has 3 subsystems as follows.)

1. MAL_ANALZR : Malicious Activity Analysis Systems
   - Pre-Installed agents on intentionally infected VMs monitor all the activities of harmful processes (Intentional infection is made through visiting URLs or executing the binary files provided by the collecting channels in Module 1).
   - The agents monitor the modification of files, registries, and DNS queries, which are occurred by the suspicious malware.
   - After monitoring is completed (default timeout value is 300 seconds if no suspicious activities are found), MAL_ANALZR initializes the infected VMs.
2. MAL_CHKR : Antivirus Checking Systems
   - Check whether detected binaries are already known as botnet malware through comparing them with the hash values queried from commercial Antivirus databases
3. MAL_MGMT_CNTR : Pre-detection Management Systems
   - Organize botnet sample databases and keep records of abnormal behaviors and processes from monitoring agents in VMs.
   - Perform the cross-analysis with the other information sources, for example, blacklists provided by the government agency

Figure 2. System modules. VM, virtual machine; DNS, Domain Name System.

DOI: 10.1002/sec

the stage where botnets, which are already formed, await instructions from C&C servers. Our system differs from the previously proposed detection methods in that it is designed to detect C&C servers even at the stage of recruiting bot members or at the stage of forming botnets.

3. PROPOSED SYSTEM

3.1. Honeynet design

As mentioned in Section 2, honeypots can be generally distinguished into two types: LIH and HIH. LIH is designed to make limited interactions; hence, attacker activities are limited to the level of emulation by LIH. The advantage of LIH is that low risks are involved in its deployment. However, the major weakness of LIH is that it can be easily detected by attackers, hence can only lure limited automated attacks. The weakness of LIH can be overcome with using HIH. However, more risks are involved when deploying HIH because attackers can have complete control over the honeypot. In our system, because of the weakness of LIH that limits information gathering of malware behavior, we use real virtual machines for honeypots, which are HIHs. To avoid the risk associated with HIH, we control over the malware in the process of collection and selection. Besides, we install agents that restore the virtual machines to a previous clean state after the malware behavior observations are completed. The biggest difference between our enhanced honeynet and the previously implemented honeynet is that our honeynet does not passively wait to be attacked. Instead, our system actively collects malwares and determines what intentionally infects our honeynets.

3.2. System overview

Our proposed system consists of two major modules: Module 1 is the malware collection module, and Module 2 is the module for analyzing behaviors of the collected malware and for identifying suspicious malware for botnets.

Whereas traditional methods of botnet propagation intend to exploit known vulnerabilities on OSs or applications, recent propagation methods utilize various data transmission channels (e.g., URL links included in social networking application, URL links of P2P sites or Web-hard, postings on bulletin board system, e-mail with attached files, warez File Transfer Protocol servers, etc.). Therefore, collecting suspicious files from numerous sources is the first step for early detection. Module 1 is hence designed to maximize the number of collection channels of botnet samples. It gathers all the suspicious URL links and attached files from the six channels that we setup at both domestic backbone routers and international gateway routers. Because botnet malwares generally infect unpatched computers in several locations even across country boundaries, we setup two channels at international gateways and name them as BOT_SPL_HTTP_CLTR and BOT_SPL_SPAM_CLTR. Among these two channels, BOT_SPL_HTTP_CLTR gathers samples from four intrusion detection system (IDS) machines, the throughput of each of which is up to 2.5 Gbps, and is designed to get any downloadable files that are matched with the pre-defined regular expression. BOT_SPL_SPAM_CLTR, which is also located at international gateways, collects all the outgoing spam mails from the spam filter systems and extracts suspicious URLs or attachments from the contents in the collected spam mails. These two channels are used to inspect international Web and mail traffic. The rest of four sample

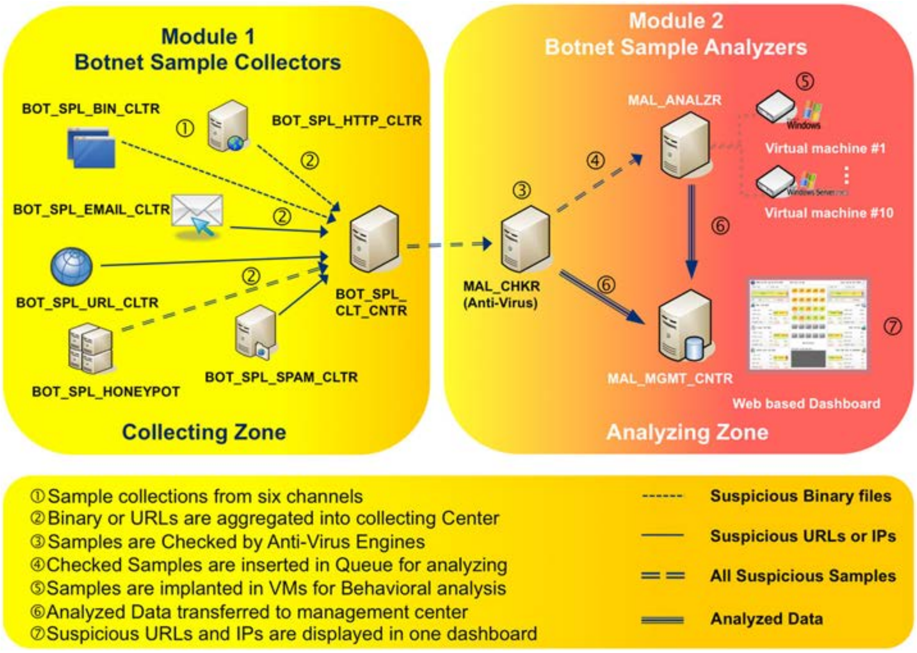


Figure 3. System overview. VM, virtual machine.

collecting channels are within the border of country for monitoring domestic traffic. Among these four channels, BOT_SPL_URL_CLTR and BOT_SPL_BIN_CLTR are operated manually relying on user input.

Module 2 comprises three subsystems, which are MAL_ANALZR (malicious activity analysis systems), MAL_CHKR (AV checking systems), and MAL_MGMT_CNTR (pre-detection management systems). Samples collected in Module 1 are delivered to MAL_CHKR system to check whether detected binaries are already known as botnet malwares through comparing them with the hash values queried from commercial AV databases. After checkups, the results are delivered to MAL_MGMT_CNTR system for recording and reporting. MAL_CHKR also delivers binary files or URL lists to MAL_ANALZR system for behavioral analysis. MAL_ANALZR consists of 20 Windows virtual machines, which can be intentionally infected. Each virtual machine has its own agent program to monitor system file, process, registry, and network modification of system. After analysis, MAL_ANALZR delivers the result to MAL_MGMT_CNTR system.

The proposed system modules are presented in Figures 2 and 3.

4. EXPERIMENT RESULT

From the beginning of 2010, we have tested our system with implementing it in KORNET, the biggest Korean backbone network. Tables I and II show the number of botnet samples collected from six channels, and the result of detection during 12 months of running the system. After running the system for 12 months, we compared the list of suspicious IPs and URLs detected by our system with the blacklist provided by the Korean government agency. From comparison, we could get that our system detect 36.2% of blacklisted IPs and 40.1% of blacklisted URLs before attacks begin. Notably, among the detected blacklisted IPs and URLs, 84% of them were pre-detected before they are completely forming botnets, and were blacklisted. We can infer from the experimental results that our system has strength in detecting botnets in their propagation stage.

Table I. Six channels for botnet sample collection.

Subsystems of Module1	Main functions	Analyzed samples	Rate (%)
BOT_SPL_HTTP_CLTR	Collecting suspicious URLs or executable binaries from international gateways with specific regular expressions of HTTP GET methods	35 422	3
BOT_SPL_SPAM_CLTR	Collecting and domestic spam mails and extracting malicious files or downloadable URLs on International gateways	826 111	69
BOT_SPL_EMAIL_CLTR	Collecting KORNET's Abuse e-mail messages and extracting malicious files or downloadable URLs included in e-mails	313 609	26
BOT_SPL_HONEYPOT	Collecting botnet samples by luring attackers with Pure Honeypots running Windows VMs	11 049	1
BOT_SPL_URL_CLTR	Collecting the list of suspicious Web sites' URL, which are reported by KORNET users	7837	1
BOT_SPL_BIN_CLTR	Collecting suspicious botnet sample files, which are reported by KORNET users	158	0

Table II. The number of collected and analyzed botnet samples from six channels.

Channels	Number of analyzed samples from channels (prefix: BOT_SPL_)						Results	
	International		Domestic		Domestic (manual)		Abnormal activity detection	Botnet activity detection
Months	HTTP_CLTR	SPAM_CLTR	EMAIL_CLTR	HONEYPOT	URL_CLTR	BIN_CLTR		
Jan	8620	98 111	44 276	—	1	87	4122	457
Feb	3526	68 114	13 918	1	—	2	2937	117
Mar	484	49 540	3614	1	3693	1	3499	691
Apr	6424	88 792	16 027	—	159	20	3972	98
May	3530	86 292	12 321	7	—	6	1296	16
Jun	2886	75 808	13 361	59	—	8	1317	19
Jul	2806	98 980	16 687	125	—	11	1222	15
Aug	1783	83 138	13 869	34	—	9	783	38
Sep	1393	82 355	12 913	70	—	—	716	14
Oct	924	65 321	34 740	5856	—	—	1156	16
Nov	2617	14 523	64 145	4896	3984	1	3225	137
Dec	429	15 137	67 738	—	—	13	4215	26
Total	**35 422**	**826 111**	**313 609**	**11 049**	**7837**	**158**	**28 460**	**1644**

4.1. Samples collected by multiple channels

We gathered suspicious botnet samples from six different channels. Among the channels, three channels—(i) EMAIL_CLTR: e-mail from the POP3 protocol; (ii) HTTP_CLTR: international IDS; and (iii) SPAM_CLTR: international spam filtering system, named KAIS—turned out to be the major input sources as shown in Table I. The channels that relied on user input—such as (iv) URL_CLTR: URL collector from manual user inputs and (v) BIN_CLTR: suspicious botnet sample files reported by users—contributed less in sample collection. The number of samples gathered through HONEYPOT channel differed greatly depending on the time of collection. At the beginning of the year, only few samples were gathered through this channel. However, the number of samples gathered increased significantly from June. This might be related with the increase of Common Vulnerabilities and Exposures Candidates between May and June in 2010.

4.2. Botnet collected by multiple channels

During the 12-month test period, we have found that most of Botnets detection was from HTTP_CLTR and SPAM_CLTR as shown in Figure 4. This implies that most botnet malwares are now disseminated via Internet URL links and SPAM mail payloads. By monitoring the URL links and SPAM payloads, we have found that they were often linked with compressed obscene media files that contained malicious binary executables, which were downloaded from P2P sites. Therefore, it is desirable to collect samples directly from P2P sites in our future work.

4.3. Pre-detection of botnets

To measure the performance of our system, we checked whether our system pre-detects suspicious URLs or IPs before they were known in public and before attacks began. We collected the lists of malicious URLs and IPs, which were known in public, from many agencies. First, we got malicious URL and IP list from another major ISP in Korea. We found that these URLs and IPs were included in the blacklist provided by the Korean government agency. To cope with large-scale DDoS attacks, the Korean Government has been operating a Botnet Response Team whose role is to analyze bot programs with dynamic and static analysis and to provide their results to major ISPs in Korea to make them not to route those IPs and not to respond to botnet DNS queries.

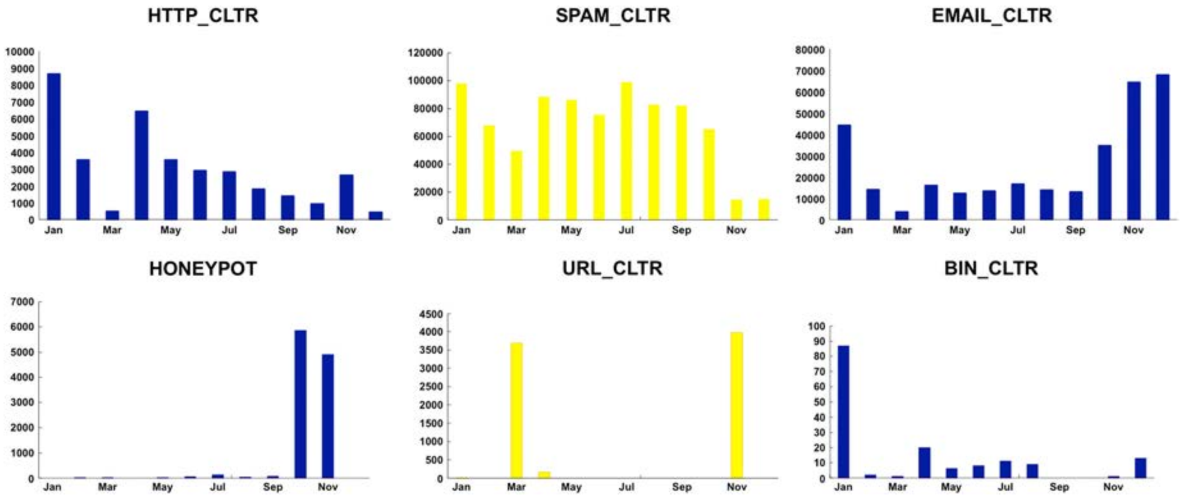


Figure 4. Analysis result of the collected samples from six channels.

Table III. Number of detected malicious URLs before/after Internet service providers' blocking.

	Days for discovering C&C URLs									
Discovering C&C URLs	30	60	90	120	150	180	210	240	Detected URLs/blocked URLs by KORNET	Ratio (%)
# of URLs before blocking	103	64	148	32	46	87	137	28	645/2020	31.9
# of URLs after Blocking	86	27	12	34	6	0	0	0	165/2020	8.2
Total	189	91	160	66	52	87	137	28	810/2020	40.1

C&C, command and control.

DOI: 10.1002/sec

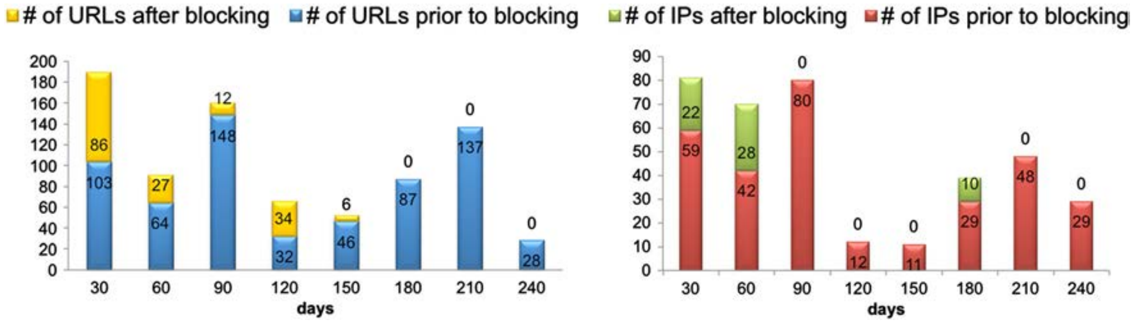


Figure 5. The time gap between our detecting of malicious URLs and IPs with Internet service providers' blocking.

The results of a comparison between the list of suspicious URLs and IPs detected by our system and the blacklist provided by the Korean government agency are shown in Table III and Figure 5. After comparison, we found that our system detected 370 IPs among 1021 blacklisted IPs, showing a 36.2% accuracy rate from a C&C URL point of view, and detected 810 blacklisted URLs, showing a 40.1% accuracy rate. Even though 40% seems to be a low accuracy rate, when we consider the large scale of DDoS attacks that is made at around 100–200 Gbps of bandwidth, ISPs are expected to get great benefits from this system.

5. CONCLUSION

Nowadays, botnet detection becomes a critical issue to government, ISPs, and business companies. However, even though there exist various methods for botnet detection, currently available methods can be applied only after attacks begin or only after significant symptoms occur. In this study, to further reduce the damage incured by botnet attacks, we propose a system that is designed to detect botnets prior to their activation or massive propagation. Our system is also designed to mitigate the shortcoming of signature-based detection and anomaly-based detection with making hybrid approach.

With a 12-month field test, we show that our system detects 40% of botnets before activation with comparing the list of suspicious URLs and IPs detected by our system to the blacklist provided by the Korean government agency. Besides, 84% of those botnets detected prior to their activation are pre-detected before they are blacklisted. There also exists a list of suspicous URLs and IPs that are detected by our system but are not listed in the blacklist of the Korean government agency. Considering this list, we expect that detection rate of our system is higher than the percentage driven by comparing the suspicious list produced by our system to the blacklist provided by the Korean government agency. We believe that these show evidences that our system is valid for pre-detection of botnets.

Our proposed system can be also deployed by other companies or government agencies. Even though our system is designed for ISPs, our system can be also adopted by other government agencies or companies with slight modification of the system. With utilizing our proposed system, companies or government agencies can proactively screen out botnets by isolating the C&C URLs and IPs.

6. FUTURE WORKS

In our future work, the number of sample collection channels will be increased to improve botnet detection accuracy rate. As mentioned in Section 4, botmasters are using P2P or social network as ways to disseminate botnet malwares. Hence, adding a channel that directly collects samples from those sites can greatly increase the accuracy rate. In addition, to better measure the performance of the system, we need to develop an efficient performance measure matrix that indicates the false negative and false positive rates of botnet detection.

ACKNOWLEDGEMENTS

This work was supported by the Agency for Defense Development(ADD) under contract no. UD110051ED. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies and endorsements of the ADD.

REFERENCES

1. Feily M, Shahrestani A, Ramadass S. A survey of botnet and botnet detection. *Third International Conference on Emerging Security Information, Systems and Technologies*, 2009.
2. Liu J, Xiao Y, Ghaboosi K, Deng H, Zhang J. Botnet: classification, attacks, detection, tracing, and preventive measures. *Fourth International Conference on Innovative Computing, Information and Control*, 2009.
3. Xie Y, Yu F, Achan K, Panigrahy R, Hulten G, Osipkov I. Spamming botnets: signatures and characteristics. *ACM SIGCOMM 2008 Conference on Data Communication*, 2008.
4. Kryvinska N, van Thanh D, Strauss C. Integrated management platform for seamless services provisioning in converged network. *International Journal of Information Technology, Communication and Convergence* 2010; **1**(1):77–91.
5. El-Semanry AM, Gadal-Haqq M, Mostafa M. Distributed and scalable intrusion detection system based on agents and intelligent technique. *Journal of Information Processing Systems* 2010; **6**(4):481–500.

6. Ponomarchuk Y, Seo D. Intrusion detection based on traffic analysis and fuzzy inference system in wireless sensor networks. *Journal of Convergence* 2010; **1** (1):35–41.
7. Zhuge J, Holz T, Han X, Song C, Zou W. Collecting autonomous spreading malware using high-interaction honeypots. *Second International Conference on Innovative Computing, Information and Control*, 2007; 438–451.
8. Provos NA. Virtual honeypot framework. *13th USENIX Security Symposium*, 2004.
9. Levine J, Grizzard J, Owen H. Application of a methodology to characterize rootkits retrieved from honeynets. *5th Information Assurance Workshop*, 2004.
10. Cooke E, Jahanian F. The zombie roundup: understanding, detecting, and disrupting botnets. *Steps to Reducing Unwanted Traffic on the Internet Workshop*, 2005.
11. Racine S. Analysis of Internet Relay Chat usage by DDoS zombies. *Master's thesis*, Swiss Federal Institute of Technology Zurich, 2004.
12. Rajab MA, Zarfoss J, Monrose F, Terzis A. A multifaceted approach to understanding the botnet phenomenon. *6th ACM SIGCOMM Internet Measurement Conference*, 2006.
13. Kugisaki Y, Kasahara Y, Hori Y, Sakurai K. Bot detection based on traffic analysis. *International Conference on Intelligent Pervasive Computing (IPC '07)*, 2007.
14. Binkley JR, Singh S. An algorithm for anomaly-based botnet detection. *Steps to Reducing Unwanted Traffic on the Internet Workshop*, 2006.
15. Choi H, Lee H, Lee H, Kim H. Botnet detection by monitoring group activities in DNS traffic. *7th IEEE International Conference on Computer and Information Technology*, 2007.
16. Dagon D. Botnet detection and response, the network is the infection. *Operation Analysis and Research Center Workshop*, 2005.
17. Wang P, Wu L, Aslam B, Zou CC. A systematic study on peer-to-peer botnets. *18th International Conference on Computer Communications and Networks*, 2009.