

Detecting New P2P Botnet with Multi-chart CUSUM

Jian Kang, Jun-Yao Zhang, Qiang Li, Zhuo Li

Department of Computer Science & Technology

Jilin University

Changchun, China

kj885788@gmail.com

Abstract—Botnets have been recognized as one of the most important threats to the Internet security. They are engaged in DDOS attacks, email spamming and other malicious activities likewise. Traditional botnets usually organized themselves in a hierarchy architecture, which offers professionals opportunities to detect or defend the botnets in their servers. However, newly-appeared P2P botnet such as Storm botnet, are revealing a decentralized feature, which brought difficulties in detection and mitigation. We believe that it is the very trend of future botnet development—adopting more sophisticated methods from being detected. Thus, in this paper, we analyze the basic principles and mechanism of this decentralized P2P botnet, and present a novel detecting method using Multi-chart CUSUM.

Keywords- P2P Botnet; Storm; Multi-chart CUSUM; Detection

I. INTRODUCTION

Botnets are formed by compromised machines which are infected by worms or Trojans. They are engaged in malicious behaviors like DDOS, Spam and etc., which largely imperil the Internet security. From SANS security report 2008[1], the Botnet is listed at the 2nd out of 10 crucial risks in the new millennium.

In recent years the newly-evolved P2P botnets are flourishing in the Internet. Because the “traditional” botnets(IRC and HTTP botnets) are now easily detected or controlled by taking measurements on the Command and Control(C&C) servers. While in P2P network, every peer serves as both server and client, thus little harm would be caused by single-point (server) failure. Also, the new P2P botnets are using new techniques like rootkits, Fast-flux and etc which made them hard to be detected. One example is Storm Botnet, appeared in early 2007[2] and quickly developed to the “biggest” botnet for the Internet world. It indicates a more sophisticated P2P approach by using an embedded decentralized architecture. As its threats to the Internet Security increasing, searching for detecting and mitigating methods is becoming urgent. Therefore, in this paper, 1) we present a brief overview of storm’s mechanism.

2) We propose a novel detecting method using the Multi-chart CUSUM test in detecting this botnet. Kaulfman algorithm is also applied on the dynamic threshold adjusting to improve the detecting precision. The results prove that the method can detect the storm botnet in a relatively high precision with both low false-positive and false-negative rate. We are planning to work on ways of mitigating and preventing those botnets in the future.

II. RELATED WORK

Since Decentralized P2P botnets are discovered not long ago, research works on this issue are still at the beginning stage.

Julian B. Gizzard et al. [3] studied and analyzed the storm’s mechanism thoroughly, including the infection steps, communication methods and so forth, which is valuable for later studies.

Antti Nummipuro et al. [4] presented some of the behavior characteristics of the P2P botnets, and offered thoughts of controlling bots on hosts, such as using the System Service Table (SST) Hooking and etc.

Matthew STEGGINK et al. [5] analyzed the botnets’ net flow, found some unique characteristics of storm when comparing them with other software’s net flows.

Reference [6] offered a novel botnet mitigation thought—joining into the Overnet network as a peer itself and publishing large number of keys to delay the communication between bots.

In SRI technical Report [7], Phillip Porras et al. presented a penetrating analysis in storm’s logic, and provoked a Dialog-based Detecting Method—using Snort to discover the dialogs and BotHunter for dialog matching.

Above all, researches on decentralized botnets detecting are still in a new phase. Many studies conclude the simple checks such as single net flow characteristic, host behaviors, and etc can distinguish between the botnets and regular activities, and thus leads to imprecise results. In this paper, we propose a method of integrating several factors as our detecting characters, testing them by improved change-point detecting algorithm—Multi-chart CUSUM, and adjusting the thresholds by Kaulfman algorithm. This prototype system effectively improves the precision in new P2P botnet detection and got relatively satisfied experiment results.

Supported by NSFC (No.60703023); the Science and Technology Development Plan of Jilin Province of China (No.20080108)

III. P2P BOTNET AND STORM BOTNET

Recent years have witnessed an increasingly appearance of various P2P botnets-trying to “borrow” the P2P networks as their communication tools. Storm, also called Peacomm, Zhelatin or NuWar, is the new P2P botnet based on Kademia algorithm [8] and Overnet/eDonkey protocol, and used the complete decentralized architecture for C&C. It is hard to detect partly because its communication channel is encrypted. More importantly, new techniques like fast-flux, rootkit and even anti-reverse engineering are used into this botnet.

The Storm infection and communication could be briefly divided into two different phases-the “Bot Initialization” and the “Secondary Injection”.

In the initializing step, the worm’s executable binary is downloaded by users from their email boxes. Then it makes several configurations to prepare for the “secondary injection” part, such as opening up several ports for later communication. The worm tries to contact the other peers from an encoded list, which is in the initial downloaded worm binaries. If successful, it joins into the Overnet/eDonkey net as well as steps into the second phase. In this phase, the bot uses hard coded keys to search on Overnet and download a value, which is an encrypted URL that points to the location of a secondary injection executables. The bot decrypts the value with its hard coded key and follows the executable directions such as upgrading, email spamming or others likewise.

Within the two phases, some unique net flow characteristics are worth noticing—1) When initializing, the bot randomly sends requests to connect to the other peers, thus leads to an unusual amount of ICMP “Destination unreachable” packets, which are rarely seen in regular circumstances. 2) UDP packets number are also detected sharply increasing because storm are using them for publishing itself in the Overnet, peer discovery, and other functions, as depicted in Section V Figure 2. 3) As in [5], the number of SMTP packets is also in a rising trend when botnets are activated. So we propose a detecting method based on the characteristics, as illustrated in section IV.

IV. DETECTION WITH MULTI-CHART CUSUM

A. Multi-chart CUSUM

Internet traffic could be viewed as a complex random model; any abnormality in the traffic will bring obvious changes. However, since the observation series are blurred in the Internet security issues, it is hard to build up a specific model. For this reason, a nonparametric CUSUM (NP-CUSUM) that uses minimum a priori information is needed for abnormality detection. [9, 10] Moreover, more details in net flow changing are needed in the botnet detection, so multi-chart NP-CUSUM detection algorithm is used for multi-factor detection in the network. [11]

For random series $\{X_1, \dots, X_n\}$, let H_k^i denotes the abnormality happens at time k , and detected at channel i . ($i \in \{1, \dots, N\}$), H_∞ denotes no abnormality happens. Let

$\sum_{s=k}^n g_{i,s}(X_i(1), \dots, X_i(s))$ stand for the score function that measures “likelihood” when H_k^i is true. So the detection statistics $S_n(i)$ is

$$S_n(i) = \left\{ \max_{1 \leq k \leq n} \sum_{s=k}^n g_{i,s}(X_i(1), \dots, X_i(s)) \right\}^+ \quad (1)$$

where $i = 1, \dots, N$ and $x^+ = \begin{cases} x, & x > 0 \\ 0, & \text{otherwise} \end{cases}$. Then each $g_{i,s}(X_i(n))$ could be changed into

$$g_{i,s}(X_i(n)) = X_i(n) - \mu_i - c_i \quad (2)$$

where $\mu_i = E_\infty X_i(n)$ means the average data in normal state, and c_i is a turning positive constant for turning the $g_{i,s}(X_i(n))$ into a negative value so the positive ones could accumulate. So (1) could be represented in a recursive way

$$S_n(i) = \{S_{n-1}(i) + X_i(n) - \mu_i - c_i\}^+ \quad (3)$$

$$S_0(i) = 0$$

Finally, the definition of judging function is

$$d_M(S_n(i)) = \begin{cases} 1, & S_n(i) > M \\ 0, & S_n(i) \leq M \end{cases} \quad (4)$$

where M is a threshold, and it adjusts dynamically as introduced in section IV.C.

B. Algorithm application

As aforementioned in section III, Storm botnet causes abnormalities in network flows. The number of UDP, ICMP, and SMTP packets are all increasing because of communication between bots, spamming or some other behaviors. Thus, we choose these 3 abnormal changes, turn them into proportion of net flow and consider them as the detecting input factors of Multi-chart NP-CUSUM algorithm. That is,

- 1) Retrieve data from monitoring device, and turn them into proportions— C_{UDP} , C_{ICMP} , and C_{SMTP} .
- 2) Then adopt the multi-chart CUSUM algorithm on them and output $S_i(UDP)$, $S_i(ICMP)$, $S_i(SMTP)$ and $d(S_i(UDP))$, $d(S_i(ICMP))$, $d(S_i(SMTP))$.
- 3) Synthesize these outputs and make the judgment. The integration method is

$$D_t = \alpha_i * d(S_n(UDP)) + \beta_i * d(S_n(ICMP)) + \gamma_i d(S_n(SMTP)) \quad (5)$$

$$\alpha_i + \beta_i + \gamma_i = 1$$

where α_i , β_i and γ_i are the weight values generated by Exponential Weighted Moving Average (EWMA) algorithm. If $D > K$ (K is a constant decided by different network situation), it is judged abnormal, and consider that botnet exists, otherwise not.

C. Dynamic threshold adjusting

Enlightened by Load-Shedding method and using the Kaufman algorithm, we adjust the threshold dynamically to improve the detection precision.

Let the $\Gamma[i]$ denotes the mapping variable of the system effective payload and detection algorithm threshold in the $(i+1)^{th}$ time span. Defined $\Gamma[0]=1$ and $\Gamma[i]$ values in $[\Gamma[\min], 1]$, where $\Gamma[\min]$ is a rather small but not 0 constant. This is because if $\Gamma[\min]$ is 0, no data flows are allowed to pass through. Hypothesize that right at the i^{th} time over, the actual payload in the system is $\rho[i]$, and $\rho[\text{target}]$ is the maximum payload, so we get $\phi[i] = \rho[\text{target}] / \rho[i]$. Thus, $\Gamma[i]$ could be presented in a recursive way

$$\Gamma[i] = \Gamma[i-1] * \phi[i] \quad (6)$$

And since $\Gamma[i] \in [\Gamma[\min], 1]$, we can get the final equation of $\Gamma[i]$, that is

$$\Gamma[i] = \max \left\{ \min \left\{ \Gamma[0] * \prod_{j=1}^i \phi[j], 1 \right\}, \Gamma[\min] \right\} \quad (7)$$

Where $i=1, \dots, n$. In this way, threshold could be computed out by $\Gamma[i]$.

D. Sequence graph of detecting system

Above all, the system's sequence is as follows (Fig. 1).

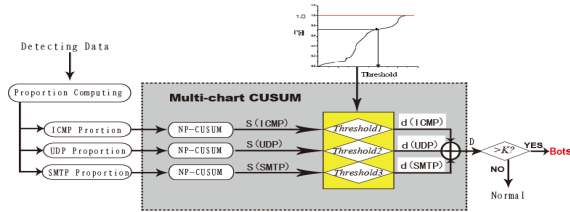


Figure 1. Prototyping system design

V. PERFORMANCE EVALUATION

A. Net flow comparison

The environment is set up following [5]. It consists of a protected net. Several computers installed with Windows

XP SP2, are all connected through a hub to a firewall. One host is logging the traffic with the Wireshark. Some of the hosts are served as storm bots.

In experiment A, we monitor the net flows under different situations. The sample packets are gathered every 10 seconds. After running normally for a while, storm bots are injected into the network.

Shown in Fig. 2, when bots began to communicate the number of UDP packets rapidly increased to nearly 20 times larger. Meanwhile, the ICMP packets also showed a skyrocketing trend when bots were activated (from 100 to 1000).

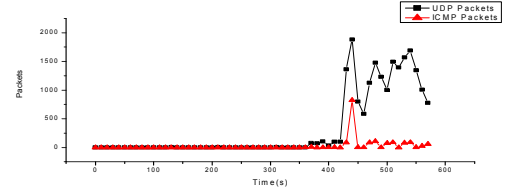


Figure 2. UDP and ICMP Packets Comparison

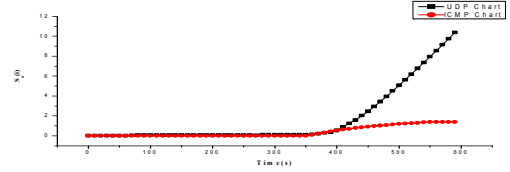


Figure 3. UDP and ICMP Packets in MNP-CUSUM

B. Multi-chart CUSUM detecting experiment

In experiment B, we verify the performance of Multi-chart CUSUM. After running normally for some time, we could compute the value of μ (UDP) and μ (ICMP), (Few SMTP packets were found in our experiment due to the time delay for spamming, so here we denote it as 0). Let $c(\text{UDP})$ be 0.1 and $c(\text{ICMP})$ be 0.05 in Formula (3). The result is Fig. 3.

From Fig. 2, at $t=360s$, the UDP and ICMP packets began sharply increase, while in Fig. 3, the MNP-CUSUM detected it with little delay. It proves that the algorithm is sensitive enough in detecting the activated bots in the network. Also, mentioned in [11], the detection delay of Multi-chart CUSUM is qualified for using in real work.

C. False-positive and False-negative comparison experiment

Experiment C is mainly to compare the algorithm's false-positive and false-negative rate with other methods. We randomly select six groups of data, which are using different combination of protocols and net flow rates. The first 3 samples are in the environment without worms and botnets. And the later 3 samples are in the situation when worms are

injected exactly under the first 3 environment respectively (Table 1).

From Table 1, the detecting precision of prototyping system is desirable, as its false-positive and false-negative rate in sample 1, 2 and 4, 5(normal office environment) approximates to the real ones. Even under the extreme situation of ARP attacks where the abnormalities are blurred by the flooding packets, the false-negative rate in sample 6 could be kept in a relatively low proportion. It proves that bots can be detected by our method.

TABLE I. FALSE-POSITIVE AND FALSE- NEGATIVE

	No.1	No.2	No.3	No.4	No.5	No.6
UDP	12	18	2	89	91	24
ICMP	6	7	1	71	84	30
SMTP	2	4	1	20	24	16
MNP-CUSUM	2	2	1	96	95	33
Real	0	0	0	100	100	50

VI. CONCLUSIONS

In this paper, we describe the newly-evolved decentralized P2P botnet, Storm Botnet; briefly analyze its infection and communication mechanism. To detect this botnet, a Multi-chart CUSUM algorithm with its input on UDP, ICMP, and SMTP abnormally increasing is proposed. And we use the dynamic thresholds to improve the precision. The results is satisfied within detecting , fault-positive and fault negative experiments, which prove that proposed method has its own advantages in detecting this new botnets on experiment network platform. We are planning to improve the detecting precision in large-scale network environment, and try to mitigate its harm to the Internet world.

REFERENCES

- [1] S. Northcutt, E. Skoudis, M. Sachs, J. Ullrich, T. Liston, E. Cole, E. Schultz, R. Dhamankar, A. Yoran, H. Schmidt, W. Pelgrin, and A. Paller, "Top Ten Cyber Security Menaces for 2008", SANS Institute, SANS Press Room, 2008.
- [2] J. Stewart, "Storm Worm DDOS Attack", SecureWorks, Inc, Atlanta GA, 2007.
- [3] J. Grizzard, V. Sharma, C. Nunnery, B. Kang and D. Dagon, "Peer-to-Peer Botnets: Overview and Case Study", In HotBots '07 conference, Usenix, 2007.
- [4] A. Nummipuro, "Detecting P2P-Controlled Bots on the Host", Seminar on Network Security, Espoo, Helsinki, 2007.
- [5] M. STEGGINK and I. IDZIEJCZAK, "Detection of peer-to-peer botnets", University of Amsterdam, Netherlands, 2007
- [6] T. Holz, M. Steiner, F. Dahl, E.W. Biersack and F. Freiling, "Measurements and Mitigation of Peer-to-Peer-based Botnets: A Case Study on Storm Worm", 1st USENIX Workshop on Large-Scale Exploits and Emergent Threats, Usenix, San Francisco, 2008.
- [7] P. Porras, H. Saidi and V. Yegneswaran, "A Multi-perspective Analysis of the Storm (Peacomm)Worm", Computer Science Laboratory, SRI International, CA, 2007.
- [8] P. Maymounkov and D. Mazières, "Kademlia: A peer-to-peer information system based on the XOR metric", 1st International Workshop on Peer-to-Peer Systems (IPTPS'02), Springer, NY, 2002.
- [9] A.G. Tartakovsky and V. Veeravalli, "Change-point detection in multichannel and distributed systems with applications", Applications of Sequential Methodologies, Marcel Dekker, Inc., pp. 339–370, New York, 2004.
- [10] A.G. Tartakovsky, "Asymptotic properties of CUSUM and Shiryaev's procedures for detecting a change in a nonhomogeneous Gaussian process", Mathematical Methods of Statistics, No. 4, pp. 389–404, 1995.
- [11] A.G. Tartakovsky, B. Rozovskii and K. Shah, "A Nonparametric Multichart CUSUM Test for Rapid Intrusion Detection", Proceedings of Joint Statistical Meetings, Minneapolis, MN, 2005.