A Real-time Method for Detecting Internet-wide SYN Flooding Attacks

Lihua Miao, Wei Ding, Jian Gong
School of Computer Science and Engineering
Key Laboratory of Computer Network Information Integration, Ministry of Education
Southeast University, Nanjing, China
{lhmiao, wding, jgong}@njnet.edu.cn

***Abstract*—Reports show that DDoS attacks are ubiquitous on the Internet and may jeopardize networks' stable operation. In order to understand the nature of this threat and further to enable effective control and management, a whole picture of the Internet-wide attacks is a necessity. Traditional methods use darknets to this end. However, with the IPv4 address space exhaustion, darknets become hard to acquire. In this paper, we seek to detect Internet-wide attacks using a *live* network. In particular, we focus on the most prevalent SYN flooding attacks. First, a complete attack scenario model is introduced according to the positions of the attacker, the victim and the attacking address. Then, after discussing the features of all scenarios, an algorithm named WSAND is proposed to detect Internet-wide SYN flooding attacks using Netflow data. In order to evaluate it, the algorithm is deployed at 28 main PoPs (Points of Presence) of the China Education and Research Network (CERNET) and the total internal address space is up to 200 /16 blocks. A large quantity of Internet-wide SYN flooding attacks detected in March 2014 is discussed in detail. With the help of the detected attacks, a case study of detecting an internal zombie is presented.**

***Index Terms*—Internet-wide SYN flooding attack, large-scale deployment, live network, Netflow data, real-time detection**

I. Introduction

Distributed Denial of Service (DDoS) attacks have a history of about 25 years up to now since the first ping flooding attack appeared in 1989 [1]. In order to avoid detection, DDoS attacks are usually launched by botnets, which are groups of zombies remotely controlled by attackers [2]. Zombies are usually coded to use spoofed source addresses and each address is only used to send part of the attack traffic. These techniques make DDoS attacks hard to detect and defend. The Arbor Networks' 9th Worldwide Infrastructure Security Report published in April 2014 shows that DDoS attacks remain to be a worldwide threat [3]. Among all forms of DDoS attacks, SYN flooding attack is the most prevalent one according to the Prolexic's 4th Quarterly Global DDoS Attack Report in 2013 [4].

As introduced above, DDoS attacks pose a serious worldwide threat to the Internet infrastructure and key services. A comprehensive observation of their characteristics, such as attack locations, attack numbers, attack durations, and attack rates, can help network operators to analyze their trend and evaluate their possible impact on the Internet. However, it's a significant challenge to monitor at enough sites to obtain a representative measure of these attacks [5].

978-1-4673-6762-2/15/$31.00

In order to understand the prevalence of DDoS attacks on the Internet, Moore etc. utilize darknet traffic to estimate worldwide attacks at a single observation point [5]. A darknet is composed of blocks of dark addresses, i.e. unused but routable addresses. When dark addresses are spoofed to launch attacks, they might receive responding traffic from victims. The responding traffic is called *backscatter*. Through analyzing backscatter, a large quantity of DDoS attacks can be observed, among which SYN flooding attacks are the most prevalent ones [6]. Nevertheless, the dark addresses are hard to obtain since the IPv4 address space is almost exhausted. In another word, most of the IPv4 address space is composed of live networks [7] instead of dark ones.

In this paper, we seek to address the question: *can we observe worldwide DDoS attacks using a live network just like what Moore did with a darknet?*

As a start, we focus on detecting SYN flooding attacks. In fact, all routable addresses can be targeted by backscatter. Hence, backscatter packets destined to live networks can also be used to detect attacks if they can be differentiated from communication ones. However, corresponding to SYN flooding attacks, backscatter SYN+ACK packets targeting active addresses can hardly be identified using DPI methods. The reason is that they are mixed with normal SYN+ACK packets and there is little difference between them. Flow data, on the other hand, are suitable for distinguishing them because in flow data backscatter SYN+ACK packets become one-way flows while normal ones are grouped into two-way flows. In addition, flow records need less computing resources and thus can ensure the scalability of our algorithm.

Under normal conditions, bidirectional SYN and SYN+ACK traffic can be observed at the network borders. This symmetric relationship could be violated under SYN flooding attacks. Besides backscatter from external victims, we also exploit this symmetric relationship to detect attacks targeting or initialized by internal hosts. These attacks also belong to the whole worldwide attack set. Especially, WSAND possesses the following advantages compared with works based on darknets.

- Darknets are usually fixed and can be known by attackers. Therefore, attackers might evade them to avoid detection.

Live networks on the contrary are hard to avoid and thus stand a better chance to observe certain attacks.
- It detects attacks targeting inside hosts and then measures can be taken to protect them.
- It detects attacks internal hosts participated in and the detection results can be utilized to understand botnet activities.

To summary, the value of our work is twofold. One is detecting Internet-wide SYN flooding attacks at a live network border. To the best of our knowledge, we are the first effort to this end. We first introduce a complete attack scenario model. Based on the model, a classification method and a Netflow based detection algorithm called WSAND are then proposed. WSAND has three advantages listed above. The other is characterizing a large quantity of Internet-wide SYN flooding attacks detected by a large-scale live network. We deploy WSAND at 28 main PoPs of CERNET. The total address space is up to 200 /16 blocks, whose size is close to the UCSD Network Telescope used in [5]. WSAND can detect attacks in a real-time fashion. We characterize the detected attacks in March 2014 and compare part of the characteristics with [5].

II. Related Work

References [5], [8] use a method called "backscatter analysis" to detect worldwide DDoS attacks. Based on 22 traces captured by a /8 darknet during 2002 and 2004, 68000 attack events are detected [5]. These attacks are analyzed from several aspects such as attack count, attack duration, attack type, and victim type. Inspired by this work, we seek to do the same work at a live network border.

Different from darknets, some hosts inside live networks can be attacked or participate in distributed attacks. These threats should also be detected because they are part of the worldwide attack set. This paper tries to fulfill this aim at the network border. Several edge router-based SYN flooding attack detection methods have been proposed over the past few years [9]–[12]. They all exploit the different characteristics between the attack and normal conditions. Most of them focus on detecting attacks targeting their own network and are based on the characteristics of TCP control packets. According to the characteristic they use, these works can be roughly classified into three categories.

1. SYN-FIN/RST method. References [9]–[11] detect SYN flooding attacks by exploiting the difference between the numbers of inbound SYN and FIN/RST packets. However, this characteristic can be useless when attackers send SYN and FIN/RST packets simultaneously.

2. SYN-SYN+ACK method. Reference [11] also introduces a detection method based on the SYN-SYN+ACK pair, which utilizes the difference between the numbers of incoming SYN and outgoing SYN+ACK packets.

3. SYN-CliACK method. Reference [12] exploits the difference between the numbers of inbound SYN and CliACK packets. Here, CliACK stands for the ACK packet sent by the client in the TCP handshake process. The key is to match the CliACK packets with the right SYN packets and [12] uses bloom filters to this end. However, this characteristic is not suitable for flow data.

The aim of our work is to detect *Internet-wide* SYN flooding attacks at a live network border using Netflow data. According to the above analysis, we choose the SYN-SYN+ACK pair.

III. The Complete Senario Model

A. Premises and Assumptions

This paper is edge router-based and utilizes traffic in two directions. For a network that has multiple edge routers, traffic from different routers should be merged first. On this basis, we assume that traffic destined to and sent by the live network can altogether be obtained at its border. In addition, we exploit the TCP control information and thus flows with TCP flags are required. Finally, we define a position function as follows:

Definition 1. For an arbitrary address g and a live network N, g can be either outside N or inside it. Let $P(g, N) = 0$ denote $g \in U - N$ and $P(g, N) = 1$ mean $g \in N$, where U represents the entire IPv4 address space.

All addresses' position information is stored in a table called IP geolocation table. With the IP geolocation table, an address can be identified as either an internal or external IP address of N.

B. The Complete Scenario Model

We first define address-normal and address-abnormal flows are follows:

Definition 2. If a flow's source and destination addresses have the same position, i.e. $P(src, N) = P(dest, N)$, the flow is *address-abnormal*. Otherwise, it's *address-normal*.

Thus, flows crossing the network border can be classified into address-abnormal and address-normal ones.

When observing them at the network border, SYN flooding attacks can be further classified into eight scenarios if the source addresses are spoofed. The division is conducted according to the position combinations of the three-tuple (attacker, victim, source address (*SIP*)). Eight scenarios are presented below and illustrated in Fig. 1.

- S_0: the position combination is (0,0,0), i.e. $attacker, victim, SIP \in U - N$. No attack traffic passes the border of N and the attack cannot be observed.
- S_1: (0,0,1), i.e. $attacker, victim \in U - N$, $SIP \in N$. Only address-normal inbound single SYN+ACK packet flows can be observed and they are backscatter from the external victim.
- S_2: (0,1,0), i.e. $attacker, SIP \in U - N$, $victim \in N$. Address-normal inbound single SYN and outbound single SYN+ACK packet flows can both be observed.
- S_3: (0,1,1), i.e. $attacker \in U - N$, $victim, SIP \in N$. Only *address-abnormal* incoming single SYN packet flows can be observed.
- S_4: (1,0,0) and it's the opposite of S_3. Only *address-abnormal* outgoing single SYN packet flows can be observed.

- S_5: (1,0,1) and it's the opposite of S_2. Address-normal outbound single SYN and inbound single SYN+ACK packet flows can both be observed.
- S_6: (1,1,0), the opposite of S_1. Only address-normal outbound single SYN+ACK packet flows can be observed and they are backscatter from the internal victim.
- S_7: (1,1,1) and is the opposite of S_0. Similar to S_0, the attack cannot be observed.

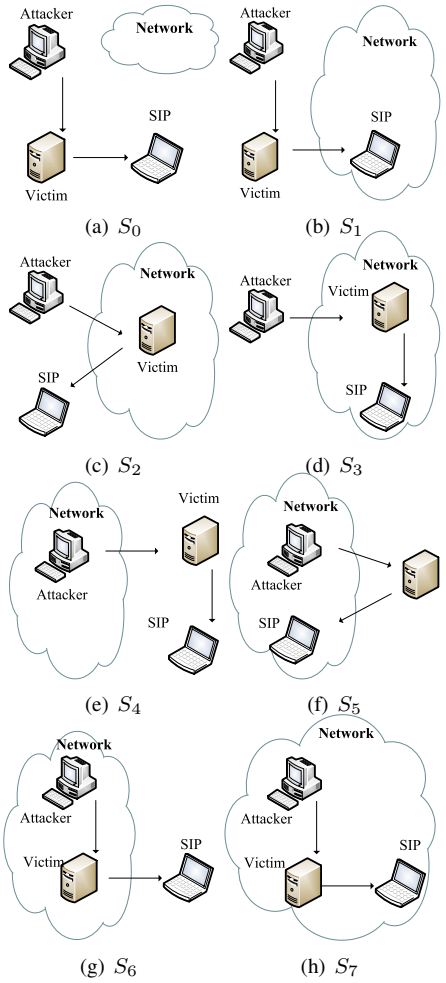

Fig. 1. Eight Basic Scenarios

Since S_0 and S_7 cannot be detected, we only discuss six scenarios $S_1 \sim S_6$ in the rest of this paper. When the source address is not spoofed, the attacker and *SIP* are the same and there are only two scenarios left: S_2 and S_5. Note that scenarios S_3 and S_4 are special because only flows in them are *address-abnormal*. Traffic observed in scenarios S_1 and S_6 is *backscatter*. All observed flows are one-way flows.

IV. SYN Flooding Attack Detection Algorithm

A. Scenario Features and Attack Types

Under normal conditions, bidirectional SYN and SYN+ACK traffic can be observed at the network borders. As introduced above, this symmetric relationship could be disturbed under SYN flooding attacks. In order to describe the behavior of the SYN-SYN+ACK pair, three detection metrics are introduced below.

We break time into discrete intervals $B_1, B_2, \ldots$ and every interval is called a detection period, which is set 300 seconds long like [13]. For all one-way single SYN and SYN+ACK packet flows in a detection interval B_i, the following metrics are recorded: X_S^g represents the packet rate of address-normal one-way single SYN packet flows g received, X_{SA}^g stands for the packet rate of address-normal one-way single SYN+ACK packet flows g sent and X_{AS}^g stores the packet rate of *address-abnormal* one-way single SYN packet flows destined to g. At the end of the interval B_i, the following anomaly detection function is used to detect anomalies:

$$d_{TH}(X^g(i)) = \begin{cases} 0 & if X^g(i) \leq TH \\ 1 & if X^g(i) > TH \end{cases} \quad (1)$$

Here, i means the i^{th} detection period. When a metric is larger than the threshold *TH*, $d = 1$ and this indicates an anomaly.

Let $\vec{D} = (d_{TH}(X_S^g(i)), d_{TH}(X_{SA}^g(i)), d_{TH}(X_{AS}^g(i)))$ be a detection vector. It is based on the symmetry relationship of address-normal one-way single SYN and SYN+ACK packet flows and the number of address-abnormal one-way single SYN packet flows. According to the last section, every scenario has its own features and can be uniquely identified by $\vec{D}$ and $P(V, N)$. For instance, the features of S_1 are that $P(V, N) = 0$ and $\vec{D} = (0, 1, 0)$. All scenarios' features are shown below in Table I.

TABLE I
Scenario Features

Scenario	$P(V,N)$	$\vec{D}$
S_1	0	(0,1,0)
S_2	1	(1,1,0)
S_3	1	(0,0,1)
S_4	0	(0,0,1)
S_5	0	(1,1,0)
S_6	1	(0,1,0)

According to $P(V, N)$, the victims of scenarios S_1, S_4, and S_5 are all external and thus they can coexist. Similarly, S_2, S_3, and S_6 can coexist too. Therefore, there are altogether $(2^3 - 1) \times 2 = 14$ attack types, shown in Table II.

For two scenarios that can coexist, if their detection vectors are not orthogonal, their combined attack type cannot be identified by only using the detection vector. Taking S_1 and S_5 for example, if an IP address's detection vector $\vec{D} = (1, 1, 0)$, it may indicate a type T_5. However, if $X_{SA}^g - X_S^g > TH$, this means that the correct type is $T_8 = \{S_1, S_5\}$. Thus, T_8 can only be identified by using an additional metric $X_{SA}^g - X_S^g$. On the other hand, the detection vectors of scenarios S_1 and S_4 are orthogonal and $\vec{D} = (0, 1, 1)$ only indicates a type T_7. The other combined attack types are also checked using the same method. For simplicity, we only detect six basic types $T_1 \sim T_6$ and two combined types T_8 and T_{12}, marked in Table II. The others can be combined by the above eight types. For instance, type T_7 is the combination of T_1 and T_4.

B. The Detection Algorithm WSAND

According to the above analyses, the attack types of an arbitrary address g in the period B_i can be decided using the

TABLE II
ATTACK TYPES

Type	Content	Detect?	Type	Content	Detect?
T_1	$\{S_1\}$	Yes	T_8	$\{S_1, S_5\}$	Yes
T_2	$\{S_2\}$	Yes	T_9	$\{S_4, S_5\}$	No
T_3	$\{S_3\}$	Yes	T_{10}	$\{S_1, S_4, S_5\}$	No
T_4	$\{S_4\}$	Yes	T_{11}	$\{S_2, S_3\}$	No
T_5	$\{S_5\}$	Yes	T_{12}	$\{S_2, S_6\}$	Yes
T_6	$\{S_6\}$	Yes	T_{13}	$\{S_3, S_6\}$	No
T_7	$\{S_1, S_4\}$	No	T_{14}	$\{S_2, S_3, S_6\}$	No

following Boolean expressions shown in Table III.

TABLE III
BOOLEAN EXPRESSIONS

Type	Boolean Expression
T_1	$\neg P(g,N) \wedge \neg d_{TH_1}(X_S^g(i)) \wedge d_{TH_1}(X_{SA}^g(i))$
T_2	$P(g,N) \wedge d_{TH_2}(X_S^g(i)) \wedge d_{TH_2}(X_{SA}^g(i))$ $\wedge \neg d_{TH_6}(X_{SA}^g(i) - X_S^g(i))$
T_{12}	$P(g,N) \wedge d_{TH_2}(X_S^g(i)) \wedge d_{TH_2}(X_{SA}^g(i))$ $\wedge d_{TH_6}(X_{SA}^g(i) - X_S^g(i))$
T_3	$P(g,N) \wedge d_{TH_3}(X_{AS}^g(i))$
T_4	$\neg P(g,N) \wedge d_{TH_4}(X_{AS}^g(i))$
T_5	$\neg P(g,N) \wedge d_{TH_5}(X_S^g(i)) \wedge d_{TH_5}(X_{SA}^g(i))$ $\wedge \neg d_{TH_1}(X_{SA}^g(i) - X_S^g(i))$
T_8	$\neg P(g,N) \wedge d_{TH_5}(X_S^g(i)) \wedge d_{TH_5}(X_{SA}^g(i))$ $\wedge d_{TH_1}(X_{SA}^g(i) - X_S^g(i))$
T_6	$P(g,N) \wedge \neg d_{TH_6}(X_S^g(i)) \wedge d_{TH_6}(X_{SA}^g(i))$

Similar to flow monitoring [14], we use a hash table to store the detection metrics in each interval and call it the sketch table. At the end of every interval, the sketch table is travelled through to identify attacks, which are stored in another hash table called the attack table. Attack statistics, such as the victims, attack types, attack durations, and attack rates are stored in the attack table. Based on the analyses provided, WSAND is proposed to detect the eight types of attacks, shown in Table IV.

TABLE IV
WSAND

Algorithm: worldwide SYN flooding attack detection algorithm
For every *one-way* flow record in period B_i: **Step1.** If it's the start of B_i, reset the sketch table. **Step2.** Read the current flow record, 1) If it's a single SYN packet flow, update the destination address's $X_{AS}(i)$ metric if it's address-abnormal and $X_S(i)$ otherwise; 2) If it's a address-normal single SYN+ACK packet flow, update the source address's $X_{SA}(i)$ metric. **Step 3.** If it's the end of B_i, travel through the sketch table and identify attacks according to Table III.

We use different thresholds $TH_1 \sim TH_6$ for distinct scenarios and discuss them in the section V-A.

V. PERFORMANCE EVALUATION

Our algorithm is deployed at 28 PoPs of CERNET with the help of a Netflow-based system named Network Behavior Observation System (NBOS) [16] developed by CERNET. The whole internal addresses are up to 200 /16 blocks. The server used by every PoP is a 64-bit Linux 2.6.32 machine with two Intel(R) Xeon(R) E5-2609 CPUs (quad core), 8GB(some only 4GB) main memory, 500GB disk, and a Gigabit NIC.

A. Parameter Configuration

According to NSFOCUS's mid-year DDoS threat report in 2013, most attacks are short and small, and 70% of them had a packet rate smaller than 0.2Mpps [17]. Thus we set the packet rate threshold as $TH = 0.1Mpps$. Subsection III-B shows that only part of the attack traffic passes through the border of the live network. Assume $\rho_1 (0 \leq \rho_1 \leq 1)$ of the attackers are inside N, and every attacker spoofs an internal address under a probability of ρ_2. If spoofed source addresses are used, then only $(1-\rho_1) \times \rho_2$ of the attack traffic can be observed in scenario S_1. For scenario S_2, the proportion is $(1-\rho_1) \times (1-\rho_2)$. The proportions for scenarios S_5 and S_6 are $\rho_1 \times \rho_2$ and $\rho_1 \times (1-\rho_2)$. Consequently, the thresholds in the algorithm are set as: $TH_1 = (1-\rho_1) \times \rho_2 \times TH, TH_2 = (1-\rho_1) \times (1-\rho_2) \times TH, TH_5 = \rho_1 \times \rho_2 \times TH, TH_6 = \rho_1 \times (1-\rho_2) \times TH$. When address-abnormal traffic is observed, an anomaly is detected. Hence, set $TH_3 = TH_4 = 0$. For scenarios S_2 and S_5, the source addresses may be real, the ratios are $1-\rho_1$ and ρ_1 in this case. We set $TH_2 = (1-\rho_1) \times TH, TH_5 = \rho_1 \times TH$ to avoid possible misjudged attacks. In practice, we use $\rho_1 = \rho_2 = 0.01$ and $TH_1 = TH_6 = 990pps, TH_2 = 99Kpps, TH_3 = TH_4 = 0pps, TH_5 = 1Kpps$.

B. Time Cost

The proposed algorithm WSAND only needs to process one-way single SYN and SYN+ACK packet flows. Assume that the number of these flows in a detection period is n, the number of the related IP addresses is m, and the count of the unexpired attack is k. The key is to design good hash functions in the algorithm in order to reduce collision. In the worst case, the hash tables become linked lists and the complexity of our algorithm is $O(nm+mk)$. In the best case, there is no collision and the time complexity is $O(n+m)$. The time cost in one of the busiest PoPs–Wuhan PoP, which has an average of 10 Gbps occupied bandwidth and a maximum of 17Gbps, is introduced below. Among 288 detection periods (300 seconds each) on 2014-6-20, WSAND's average and maximum time costs in a single detection period are 1.18 and 2 seconds respectively. The performance on the other days is similar and this indicates that WSAND is quasi real-time.

C. Empirical Results

Under these configurations, our algorithm has been running stably since December 2013. Attacks detected in March 2014 are discussed in detail in this subsection.

We term an attack observed by a PoP *"incident"* and 207622 incidents were observed in total. Fig. 2 is the ratio of the number of the observed incidents to the count of the internal addresses at every PoP. We can see that *location* is an important influence factor of the detection ability. This indicates that distributed deployment is important and we should attract other participants to join our system. Table V shows the detected numbers of the eight incident types. The majorities are T_5 and T_4 and there is no type T_{12} observed.

TABLE V
INCIDENT TYPES

Type	T_1	T_2	T_3	T_4	T_5	T_6	T_8	T_{12}
Count	633	34	1	5376	199928	389	1261	0

Incidents observed by different PoPs with an identical victim and overlapping durations are merged into a single *attack*. There were 97318 attacks observed in total during

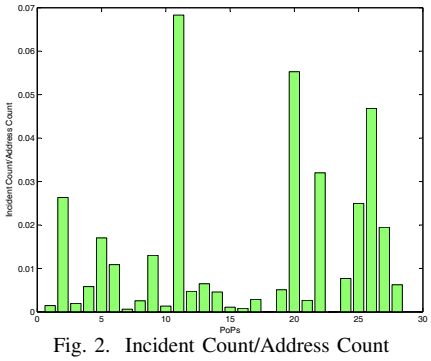

Fig. 2. Incident Count/Address Count

March 2014. Hence, an attack was observed by 2.13 PoPs on average. The distribution of the number of involved PoPs is shown in Table VI, where 48.19% of the attacks are observed by one PoP, and 15.88% are observed by more than 3 PoPs. Attacks observed by multiple points indicate that our method is effective. There was a maximum of 14 PoPs involved in one attack. This attack was against 74.x.x.2, one of the Google's server addresses. It was observed on March 19 and the corresponding incidents were all type T_1. This indicates that the attackers spoofed addresses inside CERNET to attack this victim and the backscatter was observed by our system.

TABLE VI
DISTRIBUTION OF THE INVOLVED POP COUNT

#PoPs	1	2	3	> 3
%	48.19	23.54	12.39	15.88

We divide the attack duration into 5 intervals and the distribution of the attack duration on 31 days is shown in Fig. 3. About 3139 attacks were observed every day on average. 67.25% of them last less than 10 minutes and only 8.31% last longer than 1 hour. Compared with [5], we can see that attacks have become shorter over the past eight years because [5] shows that 60% of attacks last less than 10 minutes and 15% was longer than 1 hour in duration. The longest attack in March was observed on March 5, whose duration was 200329 seconds. The attack was against 72.x.x.227, which is a server of godaddy.com. This attack was observed by 10 PoPs and all the incidents were type T_1.

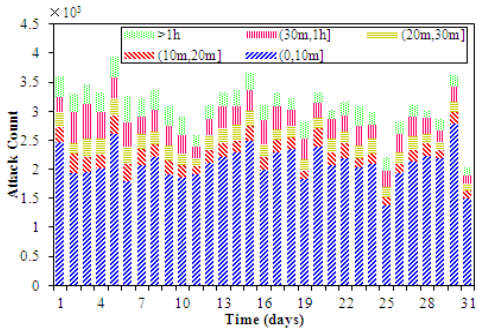

Fig. 3. Distribution of the Attack Duration

The average attack rate in every detection period is calculated and we take March $1 \sim 5$ for example. The average attack rates of the five days are shown in Fig. 4. Different from [5], only the observed attack rates instead of the estimated global attack rates are presented. Fig. 4 shows that attacks during the night are generally larger than ones during the day (UTC+8 time). The patterns of the rest 26 days were similar. Reference [5] reports that most attacks' observed attack rates were smaller than 39(=10000/256) pps. In comparison, attacks nowadays have grown much bigger. The observed largest attack was against 202.x.x.139 on 2014-3-21. This address belongs to Nanjing University of Chinese Medicine inside CERNET. 6 other PoPs participated in this attack: 5 of them observed type T_5 incidents and 1 PoP observed a type T_4 incident. This is a direct attack from hosts inside the six PoPs. The rates of the attack were 1.24Mpps and 2.14Gbps and it last about 33 minutes.

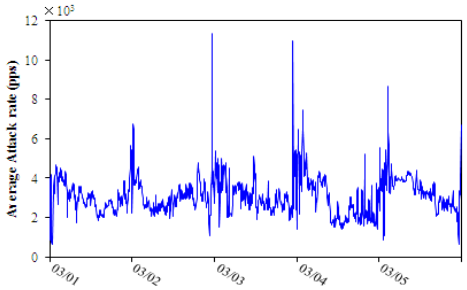

Fig. 4. Average Attack Rate (pps) in 5-minute Intervals on March $1 \sim 5$

In March 2014, 17345 victims were observed in total. Sort them in a descending order according to their attack packet rates. Fig. 5 shows the distributions of the average attack rate (pps, Kbps) and the average attack duration of the victims. There is little difference between victims' attack durations while 3.5K victims' average attack rates account for 60% of the total. These victims are further divided by their positions into four categories: inside CERNET, domestic but outside CERNET, abroad and unknown. The results are shown in Table VII. The majority of the victims are domestic but outside CERNET.

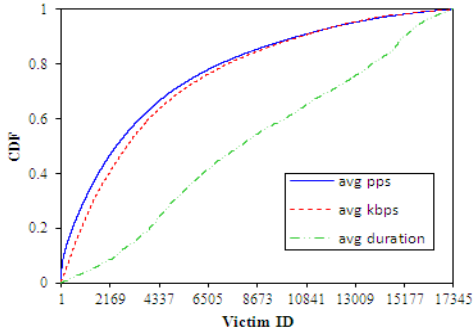

Fig. 5. CDF of Average Packet Rate, Bit Rate, and Attack Duration

TABLE VII
DISTRIBUTION OF THE VICTIM POSITION

Position	CERNET	domestic but $\notin$ CERNET	abroad	unknown
%	0.1	77.39	22.14	0.37

D. Internal Bot Detection

Table V shows that most of the incidents are type T_5, which means that internal hosts launched these attacks and the source IP addresses are also inside the network. These source addresses may be real or spoofed. According to our observation, many type T_5 attacks were launched only by 1 or 2 addresses and some of these addresses keep attacking different destinations. We speculate that some bots may use their

own addresses to sent attacks. For example, host 219.x.x.251 inside Nanjing PoP launched 53 type T_5 attacks over eight days. We captured its bidirectional packets and try to tell if it's infected. According to our observation, this address is indeed a bot. Part of the captured packets are shown in Fig. 6. Packet 8 is the control packet sent by the controller 183.x.x.215 and it contains the target's address 117.x.x.138. Once received this packet, the bot launched a type T_5 SYN flooding attack against the target. However, the source addresses of some other type T_5 attacks are spoofed. In future, we will make efforts to identify internal bots in an automatic way in order to prevent DDoS attacks from the source.

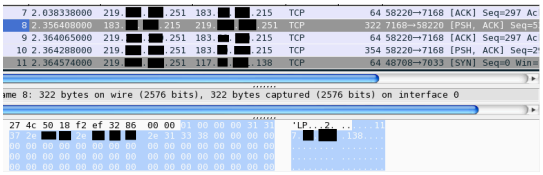

Fig. 6. A Screenshot of the Captured Packets

E. Validation

In order to validate the correctness of WSAND, we manually check if there are corresponding attacks in the packet level trace. We obtain the packet trace captured by the Jiangsu PoP on 2014-11-09 from [15]. Because the IP trace only monitors 1/4 of the inner address space, among 307 incidents detected by WSAND on that day, only 80 of them can be monitored. The others are all type T_5 and are initialed by 1or 2 inner addresses which are not monitored by the packet trace. For the 80 incidents that can be monitored, 74(92.5%) are correctly detected by WSAND. According to the packet trace, one outstanding feature is that most of them are attacks against port 80 with a packet length of 936 or 1047 bytes. For the other 6 incidents, only SYN+ACK packets are captured but WSAND indicates they are all type T_5 attacks. One possible explanation is that type T_5 actually happened but in the other 3/4 inner address space which are not monitored by the packet trace. The other is that WSAND misjudged normal SYN packets into attack ones. We plan to capture packet traces of the whole inner space for a more thorough validation in future.

VI. Conclusion and Future Work

According to the position of (the attacker, the victim, the attacking address) 8 scenarios and 14 types of the SYN flooding attacks are introduced. Based on every scenario's unique features, we propose a Netflow based detection algorithm named WSAND to detect Internet-wide SYN flooding attacks. To evaluate its performance, WSAND is deployed at 28 main PoPs of CERNET. A large quantity of worldwide SYN flooding attacks is detected in a real-time fashion. We can see that:

- Similar to darknets, backscatter can also be observed at live network borders and used to infer attacks.
- Attacks targeting inside hosts can be observed.
- With the help of the detected attacks, internal zombies who use real addresses to initial attacks can be discovered.
- Our observation indicates that:
  - 67.25% of the attacks last less than 10 minutes and only 8.31% last longer than 1 hour. Compared with [5], attacks nowadays become shorter.
  - Attacks during the night are generally larger than ones during the day (UTC+8 time). Compared with [5], attacks have become much bigger since 2006.
  - 77.39% of the victims are domestic but outside CERNET and 22.14% are abroad.

Based on the NBOS system, we also observed a large quantity of UDP flooding attacks, CHARGEN (port 19), NTP (port 123), and DNS (port 53) reflection attacks. In future, we will systematically analyze and describe these attacks. In addition, a differential threshold configuration mechanism will be established for popular and regular hosts.

References

[1] Defense.Net, "ddos attack timeline," https://www.defense.net/ddos-attack-timeline.html.

[2] F. C. Freiling, T. Holz, and G. Wicherski, *Botnet tracking: Exploring a root-cause methodology to prevent distributed denial-of-service attacks*. Springer, 2005.

[3] A. Networks, "The 9th annual worldwide infrastructure security report (wisr)," https://www.pinterest.com/arbornetworks/9th-annual-worldwide-infrastructure-security-repor/.

[4] Prolexic, "Prolexic quarterly global ddos attack report q4 2013," https://www.prolexic.com/knowledge-center-ddos-attack-report-2013-q4.html.

[5] D. Moore, C. Shannon, D. J. Brown, G. M. Voelker, and S. Savage, "Inferring internet denial-of-service activity," *ACM Transactions on Computer Systems (TOCS)*, vol. 24, no. 2, pp. 115–139, 2006.

[6] E. Wustrow, M. Karir, M. Bailey, F. Jahanian, and G. Huston, "Internet background radiation revisited," in *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement*. ACM, 2010, pp. 62–74.

[7] E. Glatz and X. Dimitropoulos, "Classifying internet one-way traffic," in *Proceedings of the 2012 ACM conference on Internet measurement conference*. ACM, 2012, pp. 37–50.

[8] C. S. David Moore, "Sco offline from denial-of-service attack," http://www.caida.org/research/security/sco-dos/.

[9] H. Wang, D. Zhang, and K. G. Shin, "Detecting syn flooding attacks," in *INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 3. IEEE, 2002, pp. 1530–1539.

[10] C. Sun, J. Fan, L. Shi, and B. Liu, "A novel router-based scheme to mitigate syn flooding ddos attacks," *IEEE INFOCOM (Student Poster)*, 2007.

[11] H. Wang, D. Zhang, and K. G. Shin, "Change-point monitoring for the detection of dos attacks," *Dependable and Secure Computing, IEEE Transactions on*, vol. 1, no. 4, pp. 193–208, 2004.

[12] C. Sun, C. Hu, Y. Zhou, X. Xiao, and B. Liu, "A more accurate scheme to detect syn flood attacks," in *INFOCOM Workshops 2009, IEEE*. IEEE, 2009, pp. 1–2.

[13] K. Xu, Z.-L. Zhang, and S. Bhattacharyya, "Profiling internet backbone traffic: behavior models and applications," in *ACM SIGCOMM Computer Communication Review*, vol. 35, no. 4. ACM, 2005, pp. 169–180.

[14] D. Eckhoff, T. Limmer, and F. Dressler, "Hash tables for efficient flow monitoring: Vulnerabilities and countermeasures," in *Local Computer Networks, 2009. LCN 2009. IEEE 34th Conference on*. IEEE, 2009, pp. 1087–1094.

[15] J. K. L. of Computer Networking Technology, "Ip trace distribution system," http://iptas.edu.cn/src/system.php.

[16] Z. Weiwei, G. Jian, G. Wenjie, and C. Shaomin, "Netflow-based network traffic monitoring," in *Network Operations and Management Symposium (APNOMS), 2011 13th Asia-Pacific*. IEEE, 2011, pp. 1–4.

[17] NSFOCUS, "Mid-year ddos threat report 2013 details ddos attack trends," http://en.nsfocus.com/2013/news_0912/144.html.