ISSN:2348-2079

Volume-5 Issue-2

International Journal of Intellectual Advancements and Research in Engineering Computations

Robust execution of packet flow in routers to prevent ddos attack using trace back

1. Ms. N. Zahira Jahan MCA.,M.Phil., Associate Professor
2. Mr. Bhagesh B.T. Final MCA
Department of MCA, Nandha Engineering College, (Autonomous), Erode-52.
zahirajahan1977@gmail.com, bhageshbhasi@gmail.com

Abstract — Distributed Denial-of-Service (DDoS) attacks are a critical threat to the Internet. However, the memoryless feature of the Internet routing mechanisms makes it extremely hard to trace back to the source of these attacks. As a result, there is no effective and efficient method to deal with this issue so far. In this paper, we propose a novel trace back method for DDoS attacks that is based on entropy variations between normal and DDoS attack traffic, which is fundamentally different from commonly used packet marking techniques. In comparison to the existing DDoS trace back methods, the proposed strategy possesses a number of advantages it is memory non-intensive, efficiently scalable, robust against packet pollution, and independent of attack traffic patterns. The results of extensive experimental and simulation studies are presented to demonstrate the effectiveness and efficiency of the proposed method. Our experiments show that accurate trace back is possible within 20 seconds (approximately) in a large-scale attack network with thousands of zombies.

Index Terms—DDoS, IP trace back, entropy variation, flow.

I. INTRODUCTION

It is an extraordinary challenge to trace back the source of Distributed Denial-of-Service (DDoS) attacks in the Internet. In DDoS attacks, attackers generate a huge amount of requests to victims through compromised computers (zombies), with the aim of denying normal service or degrading of the quality of services. It has been a major threat to the Internet since year 2000, and a recent survey [1] on the largest 70 Internet operators in the world demonstrated that DDoS attacks are increasing dramatically, and individual attacks are more strong and sophisticated. Furthermore, the survey also found that the peak of 40 gigabit DDoS attacks nearly doubled in 2008 compared with the previous year. The key reason behind this phenomena is that the network security community does not have effective and efficient trace back methods to locate attackers as it is easy for attackers to disguise themselves by taking advantages of the vulnerabilities of the World Wide Web, such as the dynamic, stateless, and anonymous nature of the Internet [2], [3]. IP trace back means the capability of identifying the actual source of any packet sent across the Internet. Because of the vulnerability of the original design of the Internet, we may not be able to find the actual hackers at present. In fact, IP trace back schemes are considered successful if they can identify the zombies from which the DDoS attack packets entered the Internet. Research on DDoS detection [4], [5], [6], [7], [8], [9], mitigation [10], [11], [12], and filtering [13], [14], [15], [16], [17], [18] has been conducted pervasively. However, the efforts on IP trace back are limited.

A number of IP trace back approaches have been suggested to identify attackers and there are two major methods for IP trace back, the probabilistic packet marking (PPM) and the deterministic packet marking. Both of these strategies require routers to inject marks into individual packets. Moreover, the PPM strategy can only operate in a local range of the Internet (ISP network), where the defender has the authority to manage. However, this kind of ISP networks is generally quite small, and we cannot trace back to the attack sources located out of the ISP network. The DPM strategy requires all the Internet routers to be updated for packet marking. However, with only 25 spare bits available in as IP packet, the scalability of DPM is a huge problem.

Moreover, the DPM mechanism poses an extraordinary challenge on storage for packet logging for routers. Therefore, it is infeasible in

practice at present. Further, both PPM and DPM are vulnerable to hacking is referred to as packet pollution. IP trace back methods should be independent of packet pollution and various attack patterns. In our previous work on DDoS attack detection, we compared the packet number distributions of packet flows, which are out of the control of attackers once the attack is launched, and we found that the similarity of attack flows is much higher than the similarity among legitimate flows, e.g., flash crowds. Entropy rate, the entropy growth rate as the length of a stochastic sequence increases was employed to find the similarity between two flows on the entropy growth pattern and relative entropy, an abstract distance between two probabilistic mass distributions was taken to measure the instant difference between two flows.

In this paper, we propose a novel mechanism for IP trace back using information theoretical parameters, and there is no packet marking in the proposed strategy; we, therefore, can avoid the inherited shortcomings of the packet marking mechanisms. We categorize packets that are passing through a router into flows, which are defined by the upstream router where a packet came from, and the destination address of the packet. During non-attack periods, routers are required to observe and record entropy variations of local flows. In this paper, we use flow entropy variation or entropy variation interchangeably. Once a DDoS attack has been identified, the victim initiates the following pushback process to identify the locations of zombies: the victim first identifies which of its upstream routers are in the attack tree based on the flow entropy variations it has accumulated, and then submits requests to the related immediate upstream routers. The upstream routers identify where the attack flows came from based on their local entropy variations that they have monitored. Once the immediate upstream routers have identified the attack flows, they will forward the requests to their immediate upstream routers, respectively, to identify the attacker sources further; this procedure is repeated in a parallel and distributed fashion until it reaches the attack source(s) or the discrimination limit between attack flows and legitimate flows is satisfied.

Our analysis, experiments, and simulations demonstrate that the proposed trace back mechanism is effective and efficient compared with the existing methods. In particular, it possesses the following advantages:

- The proposed strategy is fundamentally different from the existing PPM o trace back mechanisms, and it outperforms available PPM and DPM methods. Because of this essential change, the proposed strategy overcomes the inherited draw-backs of packet marking methods, such as limited scalability, huge demands on storage space, and vulnerability to packet pollutions.
- The implementation of the proposed method brings no modifications on current routing software. Both PPM and DPM require update on the existing routing software, which is extremely hard to achieve on the Internet. On the other hand, our proposed method can work independently as an additional module on routers for monitoring and recording flow information, and communicating with its upstream and downstream routers when the push-back procedure is carried out.

- The proposed method will be effective for future packet flooding DDoS attacks because it is independent of traffic patterns. Some previous works [13] depend heavily on traffic patterns to conduct their trace back. For example, they expected that traffic patterns obey Poisson distribution or Normal distribution. However, traffic patterns have no impact on the proposed scheme; therefore, we can deal with any complicated attack patterns, even legitimate traffic pattern mimicking attacks.

- The proposed method can archive real-time trace-back to attackers. Once the short-term flow information is in place at routers, and the victim notices that it is under attack, it will start the trace back procedure. The workload of trace back is distributed, and the overall trace back time mainly depends on the network delays between the victim and the attackers.

The rest of the paper is organized as follows: Section 2 describes the background of DDoS attacks and the related work which has been done so far on IP trace back. Our entropy variation-based IP trace back model is proposed in Section 3. Detailed analysis of the proposed scheme is conducted in Section 4. The related algorithms are designed in Section 5. Section 6 focuses on the performance analysis for every aspect of the proposed mechanism with Section 7 summarizing the paper and discussing future work.

2 BACKGROUND AND RELATED WORK

2.1 Background of DDoS Attacks

DDoS attacks are targeted at exhausting the victim's resources, such as network bandwidth, computing power, and operating system data structures. To launch a DDoS attack, the attacker(s) first establishes a network of computers that will be used to generate the huge volume of traffic needed to deny services to legitimate users of the victim. To create this attack network, attackers discover vulnerable hosts on the network. Vulnerable hosts are those that are either running no antivirus or out-of-date antivirus soft-ware, or those that have not been properly patched. These are exploited by the attackers who use the vulnerability to gain access to these hosts. The next step for the attacker is to install new programs (known as attack tools) on the compromised hosts of the attack network. The hosts running these attack tools are known as zombies, and they can be used to carry out any attack under the control of the attacker. Numerous zombies together form an army or botnet.

There are two categories of DDoS attacks, typical DDoS attacks and Distributed Reflection Denial-of-Service (DRDoS) attacks. In a typical DDoS attack, the master computer orders the zombies to run the attack tools to send huge volume of packets to the victim, to exhaust the victim's resources. Unlike the typical DDoS attacks, the army of a DRDoS attack consists of master zombies, slave zombies, and reflectors. The difference in this type of attack is that slave zombies are led by master zombies to send a stream of packets with the victim's IP address as the source IP address to other uninfected machines (known as reflectors), exhorting these machines to connect with the victim.

2.2 Related Work of IP Trace back

It is obvious that hunting down the attackers (zombies), and further to the hackers, is essential in solving the DDoS attack challenge. The summary of the existing DDoS trace back methods can be found. In general, the trace back strategies are based on packet marking. Packet marking methods include the PPM and the DPM. The PPM mechanism tries to mark packets with the router's IP address information by probability on the local router, and the victim can reconstruct the paths that the attack packets went through. The PPM method is vulnerable to attackers, as pointed out in [10], as attackers can send spoofed marking information to the victim to mislead the victim. The accuracy of PPM is another problem because the marked messages by the routers who are closer to the leaves (which means far away from the victim) could be overwritten by the down-stream routers on the attack tree [11]. At the same time, most of the PPM algorithms suffer from the storage space problem to store large amount of marked packets for reconstructing the attack tree [2], [4]. Moreover, PPM requires all the Internet routers to be involved in marking.

Based on the PPM mechanism, Law et al. tried to trace back the attackers using traffic rates of packets, which were targeted on the victim [13]. The model bears a very strong assumption: the traffic pattern has to obey the Poisson distribution, which is not always true in the Internet. Moreover, it inherits the disadvantages of the PPM mechanism: large amount of marked packets are expected to reconstruct the attack diagram, centralized processing on the victim, and it is easy be fooled by attackers using packet pollution.

The deterministic packet marking mechanism tries to mark the spare space of a packet with the packet's initial router's information, e.g., IP address. Therefore, the receiver can identify the source location of the packets once it has sufficient information of the marks. The major problem of DPM is that it involves modifications of the current routing software, and it may require very large amount of marks for packet reconstruction. Moreover, similar to PPM, the DPM mechanism cannot avoid pollution from attackers.

Savage et al. [16] first introduced the probability-based packet marking method, node appending, which appends each node's address to the end of the packet as it travels from the attack source to the victim. Obviously, it is infeasible when the path is long or there is insufficient unused space in the original packet. The authors proposed the node sampling algorithm, which records the router address to the packet with probability, p, on the routers of the attack path. Then, the probability of a packet marked by a router d that hops away from the victim is pð1 pÞ$^{d\ 1}$. Based on the number of marked packets, we can reconstruct the attack path. However, it requires large number of packets to improve the accuracy of the attack path reconstruction. Therefore, an edge sampling algorithm was proposed to mark the start router address and end router address of an attack link and the distance between the two ends. The edge sampling algorithm fixed the problems of the node sampling algorithm to some extent.

Based on the PPM mechanism, in [18], the traffic that targeted the victim was measured to construct the attack and then identified where the attackers were located. They focused on the traffic flows, which end at the victim, and therefore, there was a tree which was rooted at the victim. For a router on the attack tree, the outgoing flow included two parts: the locally generated flows and the transit flows from the upstream router(s) of the attack tree. The victim will collect all the marked packets from the routers and reconstruct the attack tree based on the traffic rates of the different routers. This trace back method heavily depends on the queuing model, and it requires the traffic flows to obey specific patterns, e.g., the Poisson distribution.

In [12], the randomize-and-link approach to implement IP trace back based on the probabilistic packet marking mechanism was proposed. The algorithm targets two aspects: to reconstruct the marks from the marker efficiently and to make the PPM more secure against hackers' pollution. The idea is to have every router X to fragment its unique message M_x (e.g., IP address) into several pieces, M_0; M_1; . . . M_l. At the same time, the router calculates the checksum C ¼ CðM$_x$Þ, named as cord. The router assembles the mark as b_i, and injects b_i randomly into the unused IPv4 packet header (say, N bits, which is 25 bits in the paper: 16 bits of fragmentation ID, 1 bit of the fragmentation index, and 8 bits of service type, all of them are used rarely in a common IPv4 packet). b_i includes three parts: an index of the pieces ($\log_2$ l bits), a large checksum cord C ¼ CðM$_x$Þ(N $\log_2$ l jM$_i$jbits), and a piece of M_i; i ¼ 0; 1; . . . ; l (jM$_i$j bits). The cord is quite large, for example, 14 out of 25 bits, therefore, we can treat the cord as a random number, which is hard for hackers to predict. The victim can reconstruct the message efficiently by checking the cord and the index sequence.

Yaar et al. [10] studied the marking technique to improve the PPM mechanism. They broke the 16-bits marking space into three parts: 1 bit for distance, 2 bits for fragmentation index, and a hash fragmentation of 13 bits. By this modification, the proposed FIT algorithm can trace back the attack paths with high probability after receiving only tens of packets. The FIT algorithm also performed well even in the presence of legacy routers and it is a scalable algorithm for thousands of attack sources.

Snoeren et al. proposed a method by logging packets or digests of packets at routers [15], [12]. The packets are digested using bloom filter at all the routers. Based on these logged information, the victim can trace back the leaves on an attack tree. The methods can even trace back a single packet. However, it also places a significant strain on the storage capability of intermediate routers.

In [11], two hybrid schemes that combine the packet marking and packet logging method to trace back the attack sources are proposed Distributed Link List Trace back (DLLT) and the Probabilistic Pipelined Packet Marking (PPPM). The first one preserves the marking information at intermediate routers in a specific way so that it can be collected using a link-list-based approach. The second algorithm targets propagating the IP addresses of the routers that were involved in marking certain packets by loading them into packets going to the same destination,

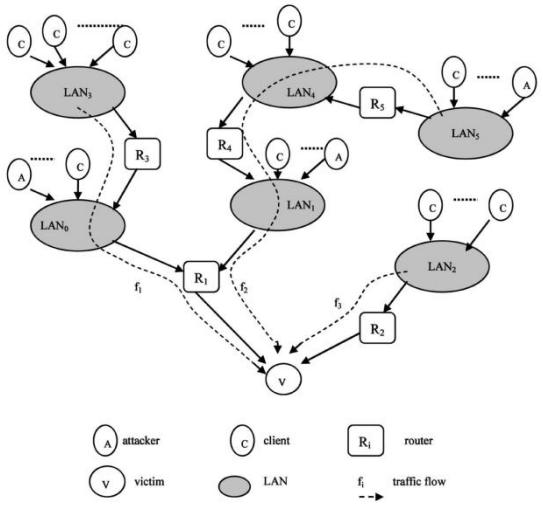

Fig. 1. A sample network with DDoS attacks

Different from PPM, Dean et al proposed a deterministic packet marking strategy for IP trace back. Every ingress router writes its own IP address into the outgoing IP packet header, and there is no more marking for the packet. They used an algebraic approach, originally developed for coding theory and learning theory, for encoding trace back information. Their idea is that for any polynomial fðxÞ of degree d in the prime field GF ðpÞ, fðxÞ can be recovered given fðxÞ evaluated at d þ 1 unique points.

PPM mechanism can only solve large flooding attacks, and it is not applicable for attacks consisted of a small number of packets. Moreover, PPM is vulnerable if hackers inject marked packets into the network. Therefore, the paper proposed a deterministic packet marking method for IP trace back. The basic idea is that at the initial router for an information source, the router embeds its IP address into the packet by chopping the router's IP into two segments with 17 bits each (16 bits for

half of the IP address and 1 bit works as index). As a result, the victim can trace which router the packets came from.

ID coding of the deterministic packet marking scheme using redundant decomposition of the initial router IP address. For an IP address, they divided them into three redundant segments, 0-13 bits, 9-22 bits, and 18-31 bits, and then five different hash functions are applied on the three segments to create five results. The resulting eight segments are recorded in the outgoing packets randomly. The victim can reassemble the source router IP using the packets it has received.

3. SYSTEM MODELING FOR IP TRACEBACK ON ENTROPY VARIATIONS

3.1 A Sample Network with DDoS Attacks

In order to clearly describe our trace back mechanism, we use Fig. 1 as a sample network with DDoS attacks to demonstrate our trace back strategy.

In a DDoS attack scenario, as shown in Fig. 1, the flows with destination as the victim include legitimate flows, such as f_3, and a combination of attack flows and legitimate flows, such as f_1 and f_2. Compared with non-attack cases, the volumes of some flows increase significantly in a very short time period in DDoS attack cases. Observers at routers R_1, R_4, R_5, and V will notice the dramatic changes; however, the routers who are not in the attack paths, such as R_2 and R_3, will not be able to sense the variations. Therefore, once the victim realizes an ongoing attack, it can push back to the LANs, which caused the changes based on the information of flow entropy variations, and therefore, we can identify the locations of attackers.

The trace back can be done in a parallel and distributed fashion in our proposed scheme. In Fig. 1, based on its knowledge of entropy variations, the victim knows that attackers are somewhere behind router R_1, and no attackers are behind router R_2. Then the trace back request is delivered to router R_1. Similar to the victim, router R_1 knows that there are two groups of attackers, one group is behind the link to LAN_0 and another group is behind the link to LAN_1. Then the trace back requests are further delivered to the edge routers of LAN_0 and LAN_1, respectively. there are attackers behind router R_4. The trace back request is then further passed to the upstream routers, until we locate the attackers in LAN_5.

3.2 System Modeling

In this paper, we categorize the packets that are passing through a router into flows. A flow is defined by a pair—the upstream router where the packet came from and the destination address of the packet. Entropy is an information-theoretic concept, which is a measure of randomness. We employ entropy variation in this paper to measure changes of randomness of flows at a router for a given time interval. We notice that entropy variation is only one of the possible metrics. Chen and Hwang used a statistical feature, change-point of flows, to identify the abnormality of DDoS attacks [6]; however, attackers could cheat this feature by increasing attack strength slowly. We can also employ other statistic metrics to measure the randomness, such as standard variation or high-order moments of flows. In order to differentiate from the original definition of entropy, we call H(F) as entropy variation in this paper, which measures the variations of randomness of flows on a given local router. we use Fig. 1 as a sample network with DDoS attacks to demonstrate our trace back strategy.

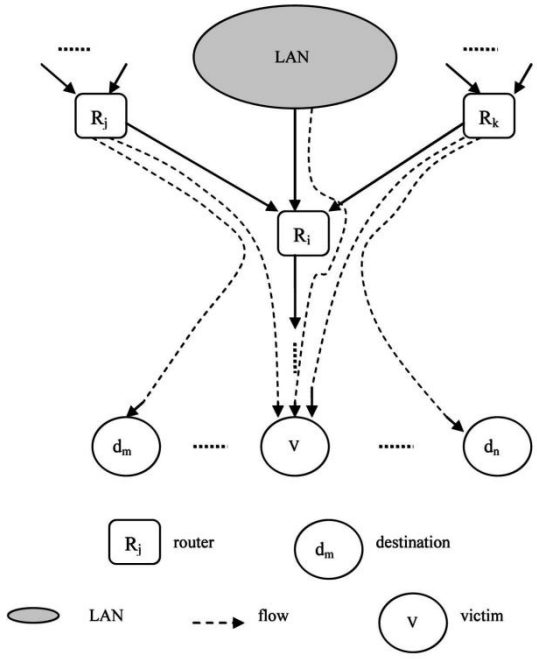

Fig. 2. Traffic flows at a router on an attack path.

4. TRACEBACK MODEL ANALYSIS

In this section, we first compare the proposed model with the existing proposals in order to show the advantages of the proposed mechanism. We then analyze the proposed entropy variation-based trace back model in detail. The features of a stand-alone router are analyzed first, followed by the investigation on the properties of the whole attack tree of a DDoS attack.

4.1 Comparison of Trace back Models

In order to show the advantages of the proposed mechanism, we compare our proposed method with the representatives of DPM and PPM [17] algorithms. The settings and network environment for the proposed algorithm are the same as that of DPM and PPM [11], respectively, in the comparisons. The DPM mechanism because it is a typical research instance for that category. It chooses one source (attacker) and one destination randomly from a tier-one ISP made up of roughly 70 backbone routers with links ranging from T1 to OC-3. The routers between the source and the destination perform packet digests using a bloom filter, and the average packet size is 400 bytes as indicated in the paper. Routers process more than 20 Mpkts/sec (roughly 2 OC-192 links, or 8 OC-48s), and there are around 1,000 flows at a router. We use [12] as an instance for the PPM strategy, which is treated as the most scalable PPM algorithm, and we calculate the related storage space and trace back time as the parameters provided by the paper, such as sampling probability p ¼ 0:05. The comparisons are listed in Table 1, and it shows clearly that the proposed mechanism outperforms the other two mechanisms in terms of scalability (the size of attack network that we can handle), storage (the storage space that we need on routers or victims to conduct IP trace back), trace back time (the overall time we need from the start time until the end of tracing process), and the operation workload (the operations on possible routers or victims).

4.2 Analysis of Entropy-Variation-Based Trace back Model

We present our assumptions below in order to make our analysis simple and clear. We assume the following:

1. There is no extraordinary change of network traffic in a very short time interval (e.g., at the level of seconds) for non-DDoS attack cases. It is true that the network traffic for a router may dynamically change a lot from peak to off-peak service times. However, this kind of change lasts for a relatively long time interval, e.g., at least at the level of minutes. If we break down these changes into seconds, the change of traffic is quite smooth in our context.

2. The number of attack packets is at least an order of magnitude higher than that of normal flows. During a DDoS flooding attack, the number of attack packets increases dramatically, and the attack packets are generated by thousands of zombies or bots [3]. Consequently, the number of attack packets is much higher than that of legitimate flows. Therefore, this assumption is reasonable. Of course, for the non-flooding attacks, this may not hold, and in this paper, we focus on the majority of the attack tools—flooding attacks. Furthermore, this is the lower bound that we can discriminate attack flows from the legitimate flows.

3. Only one DDoS attack is ongoing at a given time. It could be true that a number of attacks are ongoing concurrently in the Internet, the attack paths may overlap as well, but we only consider the one attack scenario to make it simple and clear.

4. The number of flows for a given router is stable at both the attack cases and non-attack cases.

	DPM	PPM	Entropy Variation
Scalability	**High** 2^{17}-2^{25} computers for single packet marking, more or multiple packet marking	**Low** 100 routers range of attack tree	**Very High** Unlimited under condition that every zombie generates obvious traffic
Storage	**Very High** 3.3G-44G/minute at each involved router	**High** Around 900M at the victim for one attack	**Very low** Around 240k/minute at involved routers
Traceback Time	**Low** Network delay	**Medium** Network delay plus calculation time	**Low** Network delay
Operation Workload	**Very High** Digesting packets with probability *P* (about 1M packets/second)	**Very High** Marking packets with probability *P* (about 1M packets/second)	**Very Low** Counting packet numbers for each flow

Table1.the comparision of the entrophy variation mechanisms against DPM and PPM

The local flow monitoring algorithm

1. initialize the local threshold parameter, C,δ, and sampling interval ΔT;
2. identify flows, $f_1, f_2, ..., f_n$, and set count number of each flow to zero, $x_1 = x_2 = ... = x_n = 0$;
3. when ΔT is over, calculate the probability distribution and the entropy variation as follows. $p_i = x_i \cdot \left(\sum_{i=1}^{n} x \right)^{-1}$, $H(F) = -\sum_{i=1}^{n} p_i \log p_i$;
4. save $x_1, x_2, ..., x_n$ and $H(F)$;
5. if there is no dramatic change of the entropy variation $H(F)$, namely, $|H(F) - C| \le \delta$, progress the mean $C[t] = \sum_{i=1}^{n} \alpha_i \cdot C[t-i]$, $\sum_{i=1}^{n} \alpha_i = 1$, and the standard variation $\delta[t] = \sum_{i=1}^{n} \beta_i \cdot \delta[t-i]$, $\sum_{i=1}^{n} \beta_i = 1$
6. go to step 2.

Fig. 5. The algorithm for local flow traffic monitoring

The IP traceback algorithm

1. initialize a set $A=\phi$, and obtain the local parameter of $C$and δ;
2. Let $U=\{u_i\}, i\in I$ be a set of the upstream routers, $D=\{d_i\}, i\in I$ be a set of the destinations of the packets, and v be the victim.
3. define attack flows, $f_i=<u_j, v>, i=1,2,...,n, u_j \in U$, and sort the attack flows in the descent order, and we have $f_1^{'}, f_2^{'},...,f_n^{'}$,
4. for *i*=1 to *n*
   {
   calculate $H(F\setminus f_i^{'})$
   if $(|H(F)-C|>\delta)$ then append the responding upstream router of $f_i^{'}$ to set A
   else break;
   end if;
   end for;
5. submit traceback requests to the routers in set A respectively, and deliver the confirmed zombies information, set A, to the victim.

Fig. 6. The IP traceback algorithm on a router

5. ALGORITHMS FOR THE IP TRACEBACK MODEL

In this section, we design the related algorithms according to our previous modeling and analysis. There are two algorithms in the proposed trace back suite, the local flow monitoring algorithm and the IP trace back algorithm.

The local flow monitoring algorithm is running at the non-attack period, accumulating information from normal network flows, and progressing the mean and the standard variation of flows. The progressing suspends when a DDoS attack is ongoing. The local flow monitoring algorithm is shown as Fig. 5. Once a DDoS attack has been confirmed by any of the existing DDoS detection algorithms, then the victim starts the IP trace back algorithm, which is shown as Fig. 6.

The IP trace back algorithm is installed at routers. It is initiated by the victim, and at the upstream routers, it is triggered by the IP trace back requests from the victim or the downstream routers which are on the attack path.

Even harder to find suitable data sets for our algorithms consequently, in order to evaluate our scheme, we have carefully conducted extensive simulations and real case observations. The simulation settings are arranged according to Fig. 1. We set the attack tree as a binary tree or three- First, we observe the stability of entropy variations at a local router during non-attack periods. We examine two kinds of flows, the Poisson distribution flows and the Normal distribution trace back algorithm is installed at routers. The IP trace back algorithm is installed at routers.

The proposed algorithms are independent from the current routing software, they can work as independent modules at routers. As a result, we do not need to change the current routing software.

The proposed algorithms are independent from the current routing software, they can work as independent. modules at routers. As a result, we do not need to change the current routing software. The IP trace back algorithm is installed at routers. It is initiated by the victim, and at the upstream routers, it is triggered by the IP trace back requests from the victim routers which are on the attack path and its The IP As

6. PERFORMANCE EVALUATIONS

In this section, we evaluate the effectiveness and efficiency of the proposed entropy variation based on IP trace back mechanism. Our first task is to show that the flow entropy variation is stable for non-attack cases, and find out the fluctuations for normal situations; the second task is to demonstrate the relationship between the drop of flow entropy Monitoring variation and the increase of attack strength, so that we can identify the threshold for identifying attack sources; we further simulate the whole attack tree for trace back, and evaluate the total trace back time.

Branch tree, respectively, and zombies are distributed in the attack tree uniformly. We note that, our entropy variation trace back mechanism is independent from the topology of attack network and it is also independent from the network topology of victims. We use the essential DDoS attack parameters as presented in [18] in our simulations, such as, 5-10 minutes attack duration, 10,000 packets per second of attack flows. The performance evaluation included two parts the first one focussed on the entropy variation monitoring at a local router; and the second part was to demonstrate the effectiveness of DDoS attacker trace back and the overall trace back time. As mentioned in the network security community lacks suitable data sets of real large-scale DDoS attacks, and it is even harder to find suitable data sets for our algorithms.

In order to confirm this in reality, we conducted a real case study by collecting network flow information from a gateway server of our campus

over one week, when there is no DDoS attack. There were thousands of flows per day from the collected data set. We sorted the flows in different ways: by traffic volume per connection and by the number of connections for a given time interval, and considered the top 1,000 flows as input for the experiments. The results are listed in Figs. 8 and 9, respectively.

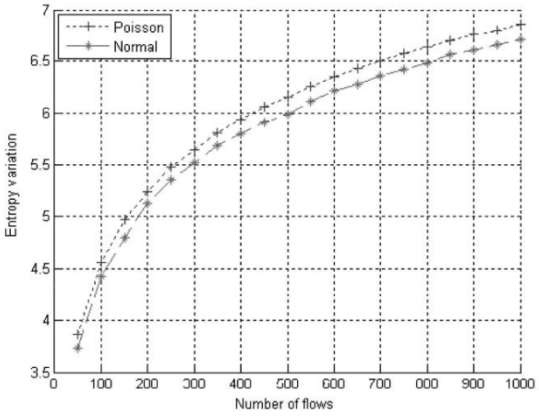

Fig. 7. The entropy variation against number of flows for Poisson and normal distributions

In Fig. 8, we increase the number of flows and check the variation of flow entropy variation on the gateway server. In Fig. 9, we sort the flows by the number of connections for different clients. We notice that the entropy variation is steady and smooth as we found in the simulation.

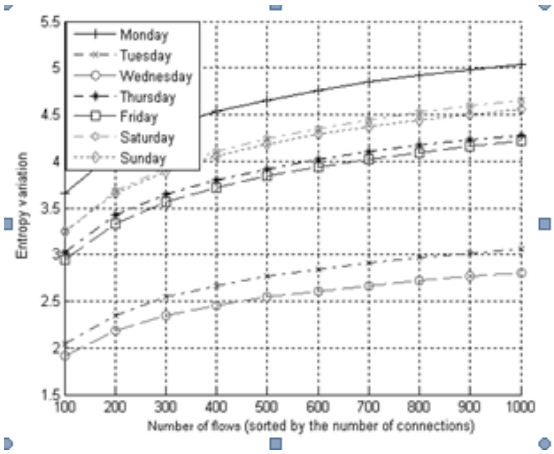

Fig. 9. Entropy variation against number of flows (sorted by the number of connections).

Consequently, in order to evaluate our scheme, we have carefully conducted extensive simulations and real case observations. The simulation settings are arranged according to Fig. 1. We set the attack tree as a binary tree or three.

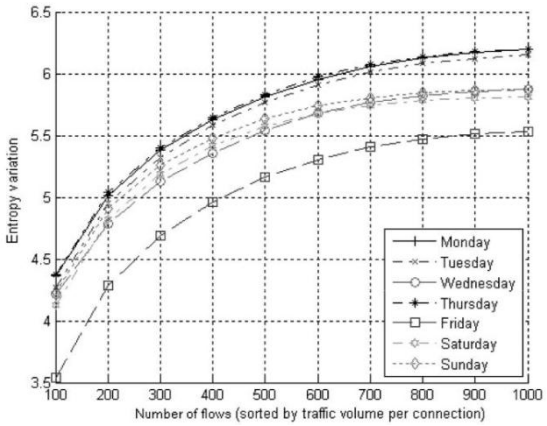

Fig. 8. Entropy variation against number of flows (sorted by traffic volume per connection).

indicates clearly that the entropy variation drops almost linearly with the increase of attack strength. flows for a local router as 1,000, and among them, there is one attack flow. We keep the packet rates of the non-attack flows at the same level, and increase the packet rate of the attack flow, from 1 to 600 times of the non-attack flow packet rate. The results are shown in Fig. 11.

Fig. 11 indicates clearly that the entropy variation drops almost linearly with the increase of attack strength. Furthermore, in order to have a direct presentation about the relationship between the decrease of entropy variation and the increase of attack strength, we transformed the results of Fig. 11 into Fig. 12.

In Fig. 10, we have learned that the standard variation of entropy variation of non-attack flows is about 0.015, and Fig. 12 indicates that the decrease of entropy variation is 0.02 when the attack strength is seven times of the normal flow, in other words, we can only discriminate DDoS attack flows when its attack strength is about seven times of the normal to relationship between the decrease of entropy variation and the increase of attack strength attack strength is not strong, say, less than seven times of the legitimate flows. The attach strength is the critical element for our trace back mechanism, and our strategy is effective once the attack strength is obvious from legitimate flows, for example, at least seven times stronger. Therefore, the proposed trace back method can deal with the majority of DDoS attacks.

Another accuracy issue for our method is the false positive, for example, flash crowds will create false positive if we start the trace back procedure at the victim site. Assume that there are 1,024 zombies in the following simulations, and they are distributed uniformly in terms of hops from the victim. In these simulations, we ignore the hops with no zombies, and the most far away zombies

are 10 hops away from the victim, namely, each hop has around 100 zombies.

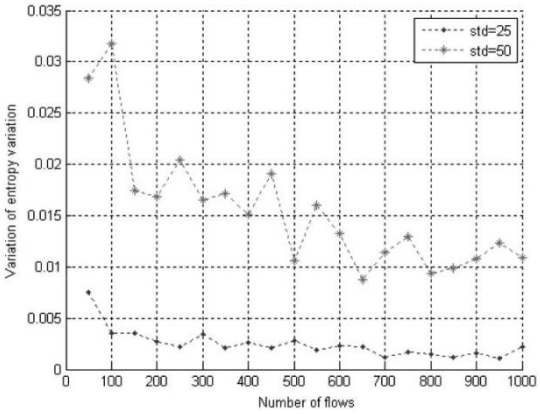

Fig. 10. The standard variation of entropy variation against number of flows with different standard variations.

In order to estimate the overall trace back time, we assume the same number of zombies (N ¼ 1;024) and aforementioned parameters. The zombies are evenly distributed in 10 groups in terms of hops away from the victim. Each of the groups can be anywhere from the victim: from 1 hop away to 31 hops away. In the worst case, the zombies are located evenly at the far end on the attack tree, in other words, the 10 groups of zombies are located from 21 to 30 hops away from the victim. We simulated each case for the binary attack tree and the three-branch attack tree, respectively, and the results are shown in Fig. 15.

Fig. 15 shows that the total trace back time is about 25 seconds in the worst case (the most far away zombies are 30 hops away from the victim), and it is less than 20 seconds if the most far away zombies are 23 hops away from the victim. In [23], it was reported that their trace back time is about 20 seconds for a single attack source with maximum 23 hops away from the victim.

In the worst case, the zombies are located evenly at the far end on the attack tree, in other words, the 10 groups of zombies are located from 21 to 30 hops away from the victim. Fig. 15 shows that the total trace back time is about 25 seconds in the worst case (the most far away zombies are 30 hops away from the victim), and it is less than 20 seconds if the most far away zombies are 23 hops away from the victim. In , it was reported that their trace back time is about 20 seconds for a single attack source with maximum 23 hops away from the victim. Based on if the number of hops to between two Internet ends is 15, then the general trace back time is around 10

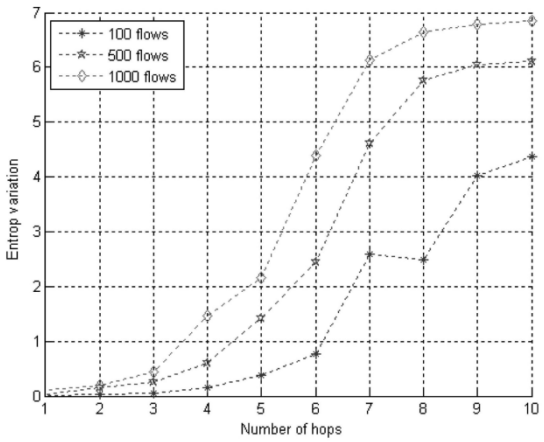

Fig. 13. The trace back time for DDoS attacks

seconds for the binary attack tree, and less than 7 seconds for three-branch attack tree for our trace back method. This simulation demonstrates that our method is better than the previous trace back method in terms of overall trace back time. Trace back to the most far away zombie effectively before it disappears from the attacking scene.

REFERENCES

[1] "IP Flow-Based Technology," ArborNetworks, http://www. arbornetworks.com, 2010.

[2] C. Patrikakis, M. Masikos, and O. Zouraraki, "Distributed Denial of Service Attacks," The Internet Protocol J., vol. 7, no. 4, pp. 13-35, 2004.

[3] T. Peng Leckie, and K. Ramamohanarao, "Survey of Network-Based Defense Mechanisms Countering DoS and DDoS Problems," ACM Computing Surveys, vol. 39, no. 1, p. 3, 2007.

[4] Y. Kim et al., "PacketScore: A Statistics-Based Packet Filtering Scheme against Distributed Denial-of-Service Attacks," IEEE Trans. Dependable and Secure Computing, vol. 3, no. 2, pp. 141-155, Apr.-June 2006.

[5] H. Wang, C. Jin, and K.G. Shin, "Defense against Spoofed IP Traffic Using Hop-Count Filtering," IEEE/ACM Trans. Networking, vol. 15, no. 1, pp. 40-53, Feb. 2007.

[7] Y. Chen and K. Hwang, "Collaborative Detection and Filtering of A. Belenky and N. Ansari, "IP Traceback with Deterministic Packet Marking," IEEE Comm. Letters, vol. 7, no. 4, pp. 162-164, Apr. 2003.

[8] A. Belenky and N. Ansari, "IP Traceback with Deterministic Packet Marking," IEEE Comm. Letters, vol. 7, no. 4, pp. 162-164, Apr. 2003.

[9] K. Lu et al., "Robust and Efficient Detection of DDoS Attacks for Large-Scale Internet," Computer Networks, vol. 51, no. 9, pp. 5036-5056, 2007.

[10] R.R. Kompella, S. Singh, and G. Varghese, "On Scalable Attack Detection in the Network," IEEE/ACM Trans. Networking, vol. 15, no. 1, pp. 14-25, Feb. 2007.

[11] P.E. Ayres et al., “ALPi: A DDoS Defense System for High-Speed Networks,” IEEE J. Selected Areas Comm., vol. 24, no. 10, pp. 1864-1876, Oct. 2006.

[12] R. Chen, J. Park, and R. Marchany, “A Divide-and-Conquer Strategy for Thwarting Distributed Denial-of-Service Attacks,” IEEE Trans. Parallel and Distributed Systems, vol. 18, no. 5, pp. 577-588, May 2007.

[13] A. Yaar, A. Perrig, and D. Song, “StackPi: New Packet Marking and Filtering Mechanisms for DDoS and IP .

[14] A. Bremler-Bar and H. Levy, “Spoofing Prevention Method,” Proc. IEEE INFOCOM, pp. 536- 547, 2005.

[15] J. Xu and W. Lee, “Sustaining Availability of Web Services under Distributed Denial of Services Attacks,” IEEE Trans. Computers, vol. 52, no. 2, pp. 195-208, Feb. 2003.

[16] W. Feng, E. Kaiser, and A. Luu, “Design and Implementation of Network Puzzles,” Proc. IEEE INFOCOM, pp. 2372-2382, 2005.

[17] X. Yang, D. Wetherall, and T. Anderson, “A DoS-Limiting Network Architecture,” Proc. ACM SIGCOMM, pp. 241-252, 2005.

[18] Z. Duan, X. Yuan, and J. Chandrashekar, “Controlling IP Spoofing through Interdomain Packet Filters,” IEEE Trans. Dependable and Secure Computing, vol. 5, no. 1, pp. 22-36, Jan.-Mar. 2007.

[19] F. Soldo, A. Markopoulou, and K. Argyraki, “Optimal Filtering of Source Address Prefixes: Models and Algorithms,” Proc. IEEE INFOCOM, 2009.