A Feasible IP Traceback Framework through Dynamic Deterministic Packet Marking

Shui Yu, *Senior Member, IEEE*, Wanlei Zhou, *Senior Member, IEEE*,
Song Guo, *Senior Member, IEEE*, and Minyi Guo, *Senior Member, IEEE*

Abstract—DDoS attack source traceback is an open and challenging problem. Deterministic packet marking (DPM) is a simple and effective traceback mechanism, but the current DPM based traceback schemes are not practical due to their scalability constraint. We noticed a factor that only a limited number of computers and routers are involved in an attack session. Therefore, we only need to mark these involved nodes for traceback purpose, rather than marking every node of the Internet as the existing schemes doing. Based on this finding, we propose a novel marking on demand (MOD) traceback scheme based on the DPM mechanism. In order to traceback to involved attack source, what we need to do is to mark these involved ingress routers using the traditional DPM strategy. Similar to existing schemes, we require participated routers to install a traffic monitor. When a monitor notices a surge of suspicious network flows, it will request a unique mark from a globally shared MOD server, and mark the suspicious flows with the unique marks. At the same time, the MOD server records the information of the marks and their related requesting IP addresses. Once a DDoS attack is confirmed, the victim can obtain the attack sources by requesting the MOD server with the marks extracted from attack packets. Moreover, we use the marking space in a round-robin style, which essentially addresses the scalability problem of the existing DPM based traceback schemes. We establish a mathematical model for the proposed traceback scheme, and thoroughly analyze the system. Theoretical analysis and extensive real-world data experiments demonstrate that the proposed traceback method is feasible and effective.

Index Terms—Cybersecurity, IP traceback, packet marking, scalability

1 Introduction

Distributed Denial of Service (DDoS) attack remains an open problem. The research in this field is usually categorized into detection [1], [2], mitigation [3], [4], and traceback [5], [6], [7]. There are various detection methods in place, such as detection against mimicking attacks [2], [8] and information theory based detection [9]. Based on detection, we are able to perform attack source traceback, and traceback is a critical step to eliminate cyber attacks. An overview of this research field could be found at a bookbrief [10]. Due to the fact that most of cyber attacks are conducted through botnets [2], [11], [12], we therefore use bots (compromised computers) and attack sources exchangeably in this paper.

For simplicity, we first clarify the terminology in DDoS attack and defence. As shown in Fig. 1, researchers usually treat a DDoS attack diagram as a tree T, which roots at the victim, V. The attack computers (bots) locate in LANs behind a router or a gateway, which is denoted as L (L stands for leaf). From L to V, it forms an *attack path*, including the intermediate routers $R_1, R_2, \ldots$. The current definition of DDoS attack source traceback (also known as IP traceback or traceback) is identifying a node on an attack path that is the closest one to L (ideally L). In other words, IP traceback means to find the most far away routers on attack pathes from the victim. Specific features of a given DDoS attack are helpful to design detection and traceback strategies. However, a general traceback method is highly expected regardless of any attack characteristics.

The current dominant traceback mechanism is packet marking, which includes two categories: Probabilistic packet marking (PPM) [13], [14], [15], [16] and deterministic packet marking (DPM) [6], [17], [18]. The basic idea is to inject marks into the unused space of IPv4 head to trace the source of the packet. Compared with the PPM mechanism, DPM is a better mechanism for traceback as it is simple, accurate, low demand on storage space and computing power. Besides packet marking, there are also some other traceback methods, e.g., watermark based traceback method [19], [20].

However, the current available DPM schemes suffer a critical disadvantage, scalability, which hinders its effective application in practice. Therefore, the maximum number of traceable sources is a major metric for various DPM schemes. As described in [16], there are at least two million routers on the Internet, and the current DPM schemes cannot cover all the possible routers. Defenders can only trace 2,048 sources in the original DPM scheme [17]. To date, the best result in this aspect is 262,144 traceable sources from the flexible DPM (FDPM) scheme [6]. This means we can only trace back around 10 percent of the total possible attack sources (in terms of routers) using the best available DPM scheme. This desperate situation motivates us to tackle the problem. A preliminary version of this work has been presented in [21].

- *S. Yu and W. Zhou are with the School of Information Technology, Deakin University, Victoria 3125, Australia. E-mail: {syu, wanlei}@deakin.edu.au.*
- *S. Guo is with the School of Computer Science and Engineering, The University of Aizu, Aizuwakamatsu, Japan. E-mail: sguo@u-aizu.ac.jp.*
- *M. Guo is with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, China. E-mail: guo-my@cs.sjtu.edu.cn.*

Manuscript received 4 July 2014; revised 17 Mar. 2015; accepted 8 May 2015. Date of publication 31 May 2015; date of current version 13 Apr. 2016. Recommended for acceptance by G. Min.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TC.2015.2439287

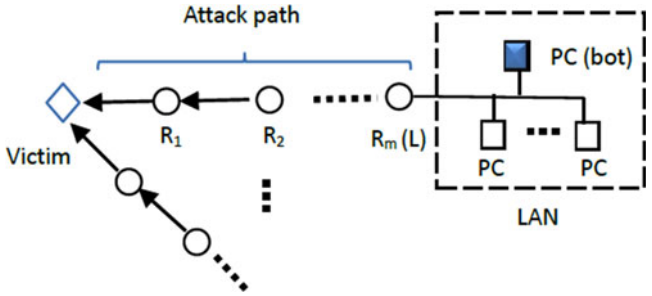


Fig. 1. An attack diagram and terms of distributed denial of service attack.

The scalability problem of the current DPM schemes roots in its static encoding mechanism. All the current DPM schemes are designed under an implicit assumption: all Internet routers are possibly involved in a DDoS attack. Therefore, they assign a unique and static ID for each router of the whole Internet. However, the available space in an IPv4 packet head is limited, and cannot serve the needs of encoding every Internet router a unique ID.

From our extensive study on DDoS attacks, we notice two characteristics of DDoS attacks: 1) In terms of space, most of the current DDoS attacks are organized by botnets [2], [11], [12], and for an attack session, the number of bots involved is at the hundreds or a few thousands level [22]. This means for an attack, there are only a small number of routers are involved, and it is not necessary and a waste to assign marks to the non-involved routers. 2) In terms of time, a DDoS attack session is usually short, and the attack frequency of a botnet is low [23]. Based on these two facts, we only need to assign unique marks for the attack related routers for a given attack session at a given time point. In other words, we can take advantage of different space and different time to significantly extend the scalability feature of the DPM mechanism.

As we know, DDoS attacks usually accompany with a surge of the number of packets addressed to victims. Due to detection sensitivity, we can detect DDoS attacks only when the increase of attack packets is sufficient. This phenomenon is generally easy to catch at the victim end, but hard to detect at the original LANs where bots seats. Especially, botnet writers are exhausting their skills to fly under the radar using available tools, such as stepping stones, reflectors, IP spoofing [1], [24], code obfuscation, memory encryption [25], and peer-to-peer implementation technology [24], [26], [27]. Therefore, a local DDoS detector may raise alarm when it notices suspicious network flows, which may be a legitimate traffic or a DDoS attack.

In this paper, we propose a marking on demand (MOD) scheme based on the DPM mechanism to dynamically assign marking IDs to DDoS attack related routers to perform the traceback task. In the proposed framework, we set up a global mark distribution server (MOD server). At every local router or gateway of participant Internet domains, we install a DDoS attack detector to monitor network flows. When there appears suspicious network flows, the detector requests unique IDs from the MOD server, and embeds the assigned unique IDs to mark the suspicious flows. At the same time, the MOD server deposits the IP address of the request router and the assigned marks into its MOD database, respectively. Once a DDoS attack is confirmed, the unique marks can be extracted from the attack packets. We can search the MOD database to identify the IP addresses of the attack sources using the marks.

We establish a mathematical model to represent the proposed traceback scheme, and analyze the effectiveness of the MOD traceback method. Compared with the existing DPM based traceback methods, the proposed one is featured a number of advantages, such as unlimited marking space, single packet traceback, low storage and computing demand. Our real world data set based experiments prove that the proposed method is effective and feasible in practice. Moreover, the proposed method can be employed for many other traceback applications, such as virus, spamming, and malware.

We note that any traceback is based on a successful detection. In this paper, we focus on traceback and assume detection methods are in place and effective.

The contributions from this paper are summarized as follows:

- We propose a feasible deterministic packet marking scheme for DDoS attack source traceback. Compared with the existing traceback methods, the proposed MOD scheme possesses a number of advantages. First, it addressed the scalability problem of the current DPM schemes, and can traceback to every possible attack source. Second, one packet traceback is feasible through the proposed scheme. Third, it offers defenders an opportunity to understand DDoS attacks in a global range, rather than a single attack.
- We proved the effectiveness of the proposed traceback method in both theoretical analysis and real world data set based experiments.
- The proposed framework makes it possible to traceback to individual compromised computers, rather than the current defined routers of attacking computers.

The remainder of this paper is organized as follows. We survey the related work in Section 2. In Section 3, we present the MOD scheme and the comparison between the proposed scheme and the existing ones. System analysis of the proposed traceback scheme is conducted in Section 4, followed by performance evaluation of the MOD scheme in Section 5. Further discussion on the MOD scheme is offered in Section 6. Finally, we summarize the paper and discuss future work in Section 7.

2 Related Work

The essential goal of DDoS attack is to deny the service of a victim through a large volume of requests, such as sending a large amount of ping requests to the victim, or massive request to the victim for downloading large files. Early DDoS attacks emerged around the year 2000, and well-known web sites, such as CNN, Amazon and Yahoo, have been the targets of hackers since then. The purpose of early attacks was mainly for fun and curiosity about the technique. However, recently we have witnessed an explosive increase in cyber attacks due to the huge financial or political rewards available to cyber attackers [1].

Currently, major DDoS attacks are carried out by Botnets. Hackers scan the whole Internet for vulnerable computers, and then compromise them as bots. As a result, an overlay network (botnet) of compromised computers is established, and controlled by botnet masters to commit malicious activities, such as DDoS attacks or information phishing [1], [28]. A DDoS attack can be carried out in various forms, such as flooding packets or synchronization attacks. A recent book by Yu overviews various aspects of DDoS attack and defence in cyberspace [10].

The current dominant traceback mechanism is packet marking, which includes two categories: PPM and DPM. In the IPv4 packet head, there are some unused bits, which are usually 16, 17, 19 or 24 bits for different underlay protocols [6]. Network operators can embed special marks or IDs in these available space for traceback purpose. Besides the packet marking mechanism, there are also other mechanisms, such as network traffic based traceback [7]. As they are not directly related to this paper, we do not discuss about them here.

The PPM strategy was firstly proposed in [13], and then further improved by researchers, such as in [16]. The basic idea of the PPM scheme is that at the network operator controlled domain, where the victim locates, special marks are injected into the available packet space for incoming packets with a probability at all routers of their domain. At the victim end, we can establish an attack tree based on the received marked packets, and identify the attack sources based on the attack tree. In order to establish a reliable attack tree, we have to accumulate a large number of marked packets, which causes a challenge on storage and computing power at the victim end. Moreover, the PPM scheme can only trace to the nodes within its domain, which are usually far away from the attacking bots.

Different from the PPM method, the DPM scheme deploys a deterministic method and tries to mark packets at routers that are the closest to attack sources (ideally, at the router of the LAN where bots stay). This scheme was firstly proposed by Belenky and Ansari [17]. They noticed that the PPM mechanism can only solve large flooding attacks, and it was not applicable for attacks consisted of a small number of packets. Therefore, they proposed a deterministic packet marking method for IP traceback. The basic idea was that at the initial router of an information source, the router embedded 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 worked as index). As a result, the victim can trace which router the packets came from.

Scalability is always a critical metric of the DPM schemes. Jin and Yang [29] improved the 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 were applied on the three segments to create five results. The resulting eight segments are recorded in the outgoing packets randomly. The victim could reassemble the source router IP using the packets it had received.

Furthermore, Xiang et al. [6] noticed the scalability disadvantage of the original DPM scheme, and proposed a FDPM method to traceback attack sources. They deployed a flexible mark length strategy to match different network environments, and the marking length varied from 16, 19 to 24 bits depending on the underneath network protocols. Moreover, they also designed a flexible flow-based marking scheme to adaptively change the marking rate according to the workload of a participating router in the scheme. The FDPM significantly improved the maximum number of traceable sources. For example, for the FDPM-19 and FDPM-24 schemes, they can trace to 8,192 and 262,144 sources, respectively. While the original DPM scheme can only trace to 2,048 sources.

Another interesting traceback method is watermark based strategy. Wang et al. [19] proposed to modulate watermarks into the time interval of a sequence of IP packets at the source side. On the other hand, the receiver can extract the watermark, and further identify the source of the packets. Jia et al. [20] proposed a simple single flow-based scheme to detect the existence of these kind of watermarks in the flow of anonymous communication systems.

Moore et al. [30] employed a network telescope technology to observe DDoS activities at a given part of the Internet, $\frac{1}{256}$ of the whole IPv4 address space. They collected DDoS attack data for a three year period. Based on their 22 data traces, they found that the average attack event frequency is 24.5/hour. If we extend the observation to the whole Internet, then the average attack event frequency is around 6,272 (24.5×256) per hour. At the same time, they found that the attack durations were relatively short: 60 percent of attacks were less than 10 minutes, and 80 percent were less than 30 minutes. Among all attacks, the highest probability of durations were five minutes (10.8 percent of attacks) and 10 minutes (9.7 percent of attacks). We will use these key statistics for our experimental part.

In order to estimate the attack power of a DDoS attack, we need to know the size of botnets. However, the research in this part is not that active as attack data is sensitive and hard to obtain from industry. Some reports that the footprint of the Torpig botnet is 182,800, and the median and average size of the Torpigs live population is 49,272 and 48,532, respectively [28]. Our recent collected data set of Conficker indicates that the size of a botnet could be as large as 2,201,183. As we know, bots are compromised computers, due to various reasons (e.g., system reinstallation, power off, anti-virus patching, and so on), the number of active bots of a given botnet is actually far less than their size or footprint. Rajab et al. [22] found that that the number of active bots for a given botnet is usually at the hundreds or a few thousands level. We will also use this information in our experimental part as well.

3 The Proposed Traceback Method

In this section, we firtly present the proposed marking on demand framework and the work flow of the proposed defence and traceback method, then we compare the MoD scheme against the existing DPM schemes.

3.1 The Marking On Demand Traceback Scheme

As aforementioned, we focus on traceback, rather than attack detection, and we assume detection methods are in place and effective.

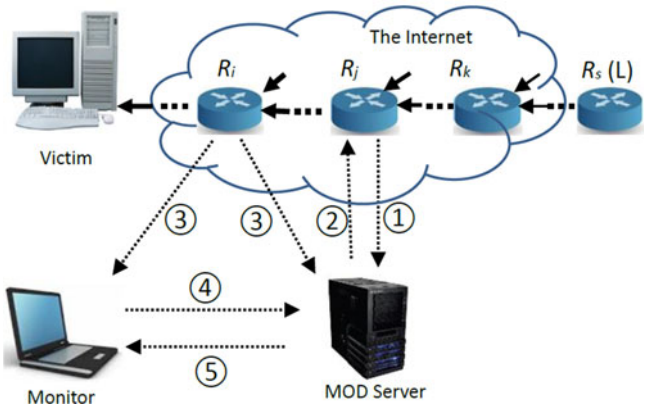


Fig. 2. The framework of the marking on demand scheme for DDoS attack traceback.

As network traffic is an effective metric for DDoS attack detection, we therefore use it in this paper for detection. Of course, we can use any detection method in practice, and it will not impact our traceback process.

For simplicity, for a given router, we define the network layer packets that share the same destination address as a *network flow* or a *flow*. We call a DDoS attack flow as an *attack flow*.

Based on this definition, if there are multiple bots within a domain or local area networks, then we treat all the packets to a victim as one network flow. Due to the aggregation nature of attack packets, the magnitude of an attack flow increases when we observe the attack flows closer to the victim.

The system diagram of the proposed marking on demand framework is shown in Fig. 2. There is an attack path from an attack source, router R_s, to intermediate routers R_k,R_j,R_i, and then the victim. It is possible that not every router on the attack path join the traceback scheme. We suppose R_k,R_j,R_i are participate routers, namely they have installed the related DDoS attack detector, possess the capability of packet marking, and the related communication capability that we proposed in the new scheme. The source router R_s may or may not participate in the traceback scheme. The goal of the proposed system is to traceback to attack source R_s as close as possible, and ideally, traceback to R_s.

In the proposed framework, we set up a global MOD server, which assigns unique marks responding to requests. Moreover, the MOD server also possesses a web based database, which stores the mark information for possible information retrieval.

Due to the detection sensibility or threshold, router R_k may not be able to notice the possible attack. With the aggregation of attack flows (they may come from different attack pathes), router R_j may be able to detect a surge of flows, but cannot confirm the attack, therefore, R_j treats it as suspicious flows, and sets off the alarm and starts the packet marking procedure. With the increasing magnitude of attack flows, finally, router R_i confirms the attack (in the worst case, the victim detects the attack).

Once an attack has been confirmed, the detecting router, e.g., R_i, will notify the MOD server with the marks that it extracted from the marked packets (they could be marked by R_s or R_j). The MOD server can then update its database to identify the earliest marking router on an attack path using the unique marks. This attack source information can be shared with public or specific users.

A detailed work flow of the proposed traceback method is presented as follows.

1) When there is a suspicious surge of network flows, the detector (e.g., R_j in Fig. 2) checks the marking space of the packets of the suspicious flows. If it is marked, then ignore it. If it is not marked, then submits a request to the MOD server for a unique mark (step 1 in the diagram).
2) The MOD server identifies a unique mark to serve the request, and deposits the related information (the mark, request source IP address, time stamp) into its database (step 2 in the diagram).
3) The detector (e.g., R_j in Fig. 2) uses the assigned mark to pad the suspicious passing flows at the available marking fields.
4) As the magnitude of attack flows gets sufficient, a downstream detector (e.g., R_i in Fig. 2) is able to identify the attack. The detector R_i will notify the MOD server about the attack with the unique marks. The MOD server will set this information in its database. Moreover, the detector will also notify the system monitor of the victim domain with the attack and its related unique marks. (step 3 in the diagram).
5) When the monitor performs the traceback task, it queries the IP addresses related to the unique marks that it received (step 4 in the diagram).
6) The MOD server checks its database about the marks, and responds the request with the related IP addresses. In this way, the monitor knows the attack sources and related action could be taken to counter the attack (step 5 in the diagram).

If a large number of domains participate in the traceback system, meaning installing the software of DDoS detection and packet marking functions, then we can traceback to the real source domain with one marked attack packet with a high probability.

3.2 Comparison on the Marking Schemes

In this section, we show the difference of the proposed marking scheme from the existing ones. We take two major DPM schemes from the literature, one is the refined version of the original DPM scheme [18], and another one is the typical and the most scalable FDPM scheme [6].

The available marking length in an IPv4 header is quite important for the performance of every DPM based schemes. In general, there are three possible units of an IPv4 packet: Fragment ID (16 bits), Reserved Flag (1 bit), and type of service (TOS in short, 8 bits). The original DPM scheme used 17 bits for marking (Fragment ID and Reserved Flag), and the FDPM scheme used 24 bits (Fragment ID and TOS) as a maximum length, and 16 bits (Fragment ID) as the least length. We refer reader to [17], [18], and [6] for the reasons why these space can be used for marking.

For the sake of comparison, we take the marking length as 24 bits as the maximum and 16 bits as the minimum for MOD scheme in our analysis and comparison. In general, we use variable l to represent the length of the available markable space.

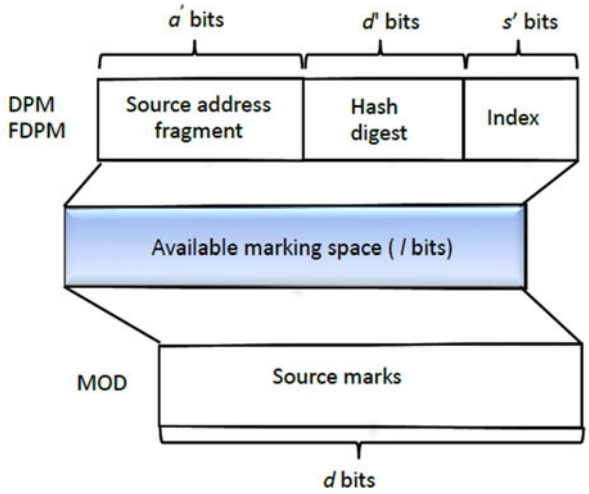


Fig. 3. The comparison between the proposed marking on demand scheme with the typical DPM schemes.

As show in Fig. 3, for a given available marking space l, in the DPM and the FDPM schemes, the available marking space is split into three parts: $d^{'}$ (d for ID) bits are employed to denote a unique ID for a ingress router, and $a^{'}$ (a for address) bits are used to carry a part of the IP address of the marking router, and $s^{'}$ (s for sequence) bits are deployed to indicate the sequence of the partial IP address. It is obvious that

$$l = a^{'} + d^{'} + s^{'}. \tag{1}$$

The constraints of equation (1) are

$$\begin{cases} a^{'} & \geq & 1 \\ d^{'} & \geq & 1 \\ s^{'} & \geq & 2 \\ 2^{s^{'}} & \geq & a^{'}. \end{cases} \tag{2}$$

The maximum traceable sources N_{tr} is decided by the variable $d^{'}$ as follows:

$$\max\{N_{tr}\} = 2^{d^{'}} = 2^{l-a^{'}-s^{'}}. \tag{3}$$

For the DPM methods, suppose $l = 17$, then $\max N_{tr} = 2,048$ for $d^{'} = 11$, $a^{'} = 1$, and $s^{'} = 5$ following the constraints in (2). It is obvious that it cannot cover all the possible routers on the Internet. At the same time, the minimum cost for this is that a victim has to collect at least 32 marked packets from the same source.

In the proposed MOD scheme, we target on single packet traceback, and all the available marking space is used for source marks. As shown in Fig. 3, we have

$$l = d. \tag{4}$$

The maximum traceable sources of the MOD scheme depends on variable d,

$$\max\{N_{tr}\} = 2^{l} = 2^{d}. \tag{5}$$

Combining equation (1), (4) and (2), the proposed MOD scheme is $2^{a^{'}+s^{'}}$ times better by its own in terms of marking efficiency.

A general qualitative comparison among the MOD scheme, the DPM scheme and the FDPM scheme are summarized as in Table 1.

4 System Analysis on the MOD Scheme

In this section, we establish a mathematical model for the proposed traceback scheme, and analyze the system with the solid knowledge of botnets and DDoS attack features. We mainly focus on the number of traceable sources and storage cost.

4.1 Traceable Sources

The maximum traceable sources is unlimited of the proposed traceback framework as we use the available marking space 2^{l} in a round robin fashion. However, we expect to know the related features of the MOD scheme, such as periodicity of mark usage. In order to achieve the goal, we firstly figure out how many routers to be traced for one attack session, and then how many routers to be traced in the environment of the whole Internet.

For a given time point t, we suppose that there are $N_b(t)$ bots involved in one attack session, and these bots come from $K(t)$ domains or networks, meaning there are $K(t)$ leaf nodes in the attack tree, and each leaf node represents a router. Let $B(t)$ be the number of bots of a network. Suppose $B_i(t) \geq B_j(t)$ if $i < j, 1 \leq i, j \leq k$. Here $N_b(t)$, $K(t)$, and $B(t)$ are all random variables.

Our goal is to traceback to the $K(t)$ routers. Research [22] have shown that the number of active bots of a botnet is at the a few thousands level. We therefore assume $N_b(t)$ is known, and expect to know how many routers ($K(t)$) are involved in a traceback process. In order to achieve this, we need to know the size distribution of botnets.

The size distribution of botnet is a long term open problem. The best result that researchers obtained in the past is that it is a non-uniform distribution [31]. Our recent research indicates that it follows power law in terms of networks [32]. We therefore use power law distribution in this paper for the size distribution of botnets.

Conficker is a well-known and one of the most recently widespread malware on the Internet. Shin et al. [33] collected a data set about 25 million Conficker victims from all over the world at different levels. We use this data set as an

TABLE 1
A General Comparison of the Proposed MOD Scheme with the Existing DPM and FDPM Schemes

	Scalability	N_{tr}	Working mode	Storage	False Positive
DPM	extremely limited	extremely limited	individual	high	inherent
FDPM	very limited	very limited	individual	high	inherent
MOD	unlimited	unlimited	global	low	non-inherent

TABLE 2
Statistics for Conficker Distribution in Terms of Domain Names at the Three Top Level Domains

	Number of botnets	Largest botnet	Smallest botnet
top level	462	2,201,183	1
level 1	20,104	1,718,306	1
level 2	96,756	1,714,283	1

TABLE 3
The Approximation of α and C in Three Top Level Domains

	Top domains	Level 1 domains	Level 2 domains
α	1.140	1.112	1.091
n	462	20104	96756
C	0.2164	0.1567	0.1320

example for the study of this paper. We count the number of bots in terms of domain names at three different domain levels: the top level, level 1, and level 2, respectively. Some statistics of the data set are listed in Table 2.

In this paper, we use the Zipf distribution as an instance of power law [34], [35]. Suppose all the bots are distributed in n domains, and the domains are sorted in terms of number of bots. Let x be the ranking of a domain, then we have the following probability.

$$Pr\{x = i\} = C \cdot i^{-\alpha}, \tag{6}$$

where α is a positive parameter, C is a constant. From the definition of power law, we know

$$\sum_{j=1}^{n} C \cdot j^{-\alpha} = 1. \tag{7}$$

In this distribution, α and n are the critical parameters. Based on our data sets, we extract these two parameters and calcualted the related parameter C, and present the results in Table 3.

As aforementioned, we use $B(t)$ to represent the size of a botnet, then $B_i(t)(1 \le i \le |K(t)|)$ follows Zipf distribution.

Currently, the common method of measuring a power law distribution is to draw a loglog diagram of probability against ranking. We further present the data sets in a loglog format in Figs. 4a, 4b and 4c, respectively. Moreover, we also present the theoretical output based on the parameters from Table 3.

From Fig. 4, we can see that the main body of the three scale measures are roughly straight lines. Especially, Fig. 4c fits the theory perfectly.

Furthermore, we suppose the smallest botnet has $M(t)$ bots, and let $|K(t)|$ be the maximum number of random variable $K(t)$ at time t. Following the definition again, we have

$$Pr\{X = |K(t)|\} = C \cdot |K(t)|^{-\alpha} = \frac{M(t)}{B(t)}. \tag{8}$$

From equation (8), we obtain

$$|K(t)| = e^{-\frac{1}{\alpha} \ln \frac{M(t)}{CB(t)}}. \tag{9}$$

From Table 2, we can see that $M(t) = 1$, then equation (9) can be simplified as

$$|K(t)| = e^{\frac{1}{\alpha} \ln CB(t)}. \tag{10}$$

In practice, if $|K(t)|$, α and $M(t)$ are known, We can estimate the total number of bots in an attack session as follows:

$$B(t) = \frac{M(t)}{|K(t)|^{\alpha}} \cdot \sum_{i=1}^{|K(t)|} \frac{1}{i^{\alpha}}. \tag{11}$$

We now extend our analysis to attack circumstance in the whole Internet. For a given time t, let random variable $A(t)$

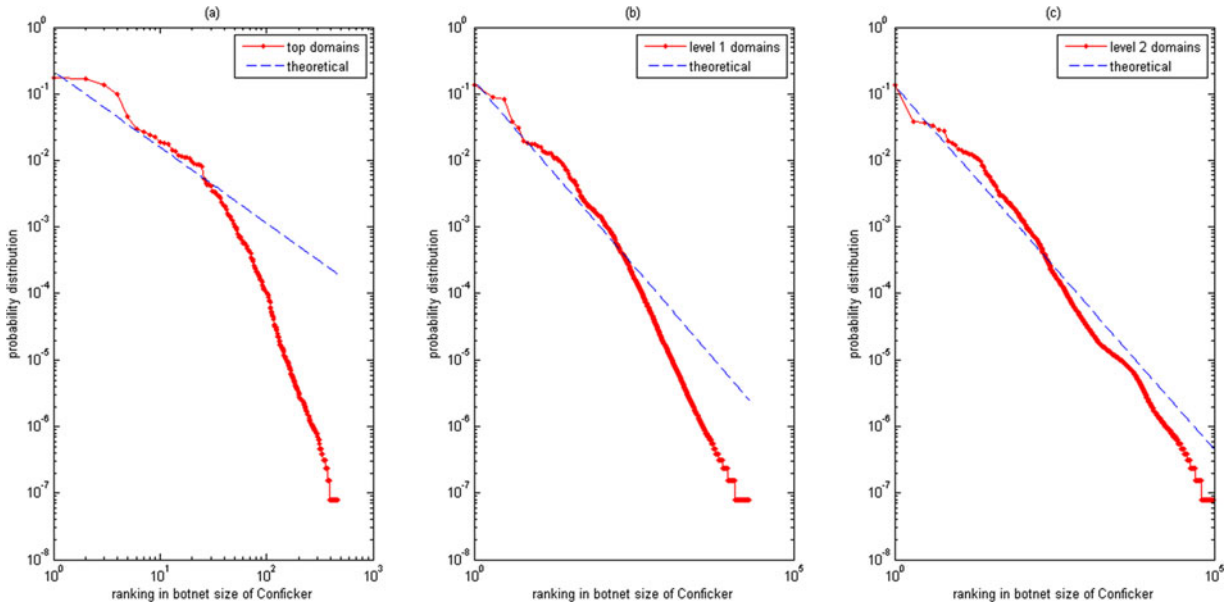


Fig. 4. Power law distribution of Conficker botnet in the top three levels of domain names with theoretical comparison.

be the number of ongoing attacks on the whole Internet, and random variable $D(t)$ for the attack duration for attacks. Then the number of routers to be traced at a given time point t is

$$N_{tr}(t) = A(t) \times K(t). \tag{12}$$

Then the time interval that we complete one round of the marking space is

$$T_I(t) = \frac{2^l}{A(t) \cdot K(t)} \cdot D(t) \tag{13}$$

Based on Wald Theorem, we have

$$E[T_I(t)] = \frac{2^l}{E[A(t)] \cdot E[K(t)]} \cdot E[D(t)], \tag{14}$$

where $E[\cdot]$ is the expectation.

As we use the marks in a repeated way, there is a possibility that two ongoing attack sessions share the same mark if one of them is a long time attack, meaning its duration is longer or equal to $T_I(t)$. In other words, suppose there are J different attack sessions within the time interval $T_I(t)$, there is no problem if the following holds

$$D_j(t) < T_I(t), \forall j \in \{1, 2, \ldots, J\}. \tag{15}$$

In the case that condition (15) does not hold, then we will have false positive (a router is not an attack source, but treated as an attack source) and false negative (a source is an attack source at this moment, but treated as not) in our traceback processing.

4.2 Storage Analysis

For all DPM based traceback schemes, each participating router needs to store the marked packets for extracting the unique marks of sources. As a result, the storage demand on routers is also an important metric for various DPM schemes. We note that we ignore the storage issue of the MOD server as it is not a problem in terms of databases.

This problem is usually modeled as the coupon collection problem, which is explained as follows. Suppose there are n $(n = 1, 2, \ldots)$ unique coupons to be collected. In order to collect all of the them, the total coupons that we have to collect is expressed as follows.

$$n\left(\frac{1}{n} + \frac{1}{n-1} + \cdots + \frac{1}{2} + 1\right). \tag{16}$$

In our case, $n = 2^{s'}$, the storage cost (denoted as N_s) for the existing DPM schemes is

$$N_s = 2^{s'}\left(\frac{1}{2^{s'}} + \frac{1}{2^{s'} - 1} + \cdots + \frac{1}{2} + 1\right). \tag{17}$$

For the proposed MOD attack source traceback scheme, we only need one marked attack packet to complete the traceback task, namely, $N_s = 1$. However, it takes far more than one storage space for existing DPM schemes.

TABLE 4
Quantity Comparison Among the Proposed MOD Scheme with the DPM Scheme and the FDPM Scheme

Scheme	Maximum traceable source	Marked packet required	Storage required
DPM-17	2^{11}	32	129.87
FDPM-16	2^{10}	32	129.87
FDPM-24	2^{18}	32	129.87
MOD-16	*unlimited*	1	1
MOD-17	*unlimited*	1	1
MOD-24	*unlimited*	1	1

5 Performance Evaluation

In this section, we conduct performance evaluation for the proposed MOD traceback scheme, mainly compared to the DPM and the FDPM schemes.

First of all, we expect to show the improvement of the proposed MOD scheme compared to its peers, the DPM and FDPM schemes.

Based on our previous theoretical analysis, we are able to compare the three DPM based traceback schemes. As the maximum number of traceable sources is a critical metric for deterministic packet marking schemes, we obtain the numerical results for the three DPM based schemes with different length of marking space, including the maximum number of traceable sources, number of marked packet needed to achieve maximum traceable source, and the storage cost to achieve the goal. The results are presented in Table 4.

From Table 4, we can see that in terms of maximum number of traceable sources, the MOD scheme is unlimited as we use the marks in a round robin fashion, which addressed the scalability problem of the DPM and the FDPM schemes.

In the DPM and the FDPM scheme, they need 32 marked attack packets to calculate the attack source in order to achieve their maximum number of traceable sources. However, in the proposed MOD scheme, we can traceback to the source with single packet.

In terms of storage cost, we can see that the proposed scheme only needs around $\frac{1}{129.87}$ of the storage cost of the DPM or the FDPM schemes.

We achieve unlimited traceback sources using the marks in a round robin style. In order to see the feasibility of the proposed method, we set our experiments in the global Internet scenario, which is represented by Fig. 2. Moreover, we need examine it with the characteristics of DDoS attacks in a global circumstance.

We summarize the key statistics of DDoS attacks in a global scenario from highly referred literature [22], [23], and present them in Table 5.

From the statistics, we know the average DDoS attack duration is about 5 minutes, then one critical question is in a global environment, how long will it last for us to use up the unique marks in one round, namely, what it will be for T_I in practice.

In order to answer the question, we firstly need to know how many attack sources (routers) are there in an attack session. From Table 2, it is reasonable to say the smallest botnet has only one bot. Moreover, from Table 5, we know that the

TABLE 5
Key Statistics on DDoS Attack Characteristics

Feature	Attack frequency [23]	Attack duration [23]	Attack rate [23]	Bots per attack session [22]
Value	105/minute	5 minutes	500 pkts/s	1000 - 2000

total number of bots in one attack session is around a few thousands level. Combining with the values of α that we extracted from the Conficker data set (shown in Table 2), we therefore can plot the relationship as in Fig. 5.

From Fig. 5, we can see that there is a nearly linear relationship between the number of maximum routers to be traced and the total number of attacking bots in an attack session. In particular, we can see that for the current scale of a botnet (with a few thousands bots), its bots are roughly distributed in around 200 domains. In other words, there are roughly 200 attack sources (routers) for a DDoS attack session.

Furthermore, base on our analysis on T_I (equation (14)) and the statistics of DDoS from Table 5, we can estimate the time interval that we complete one run of using all the marks. In this case, the length of marking space, could be 16, 17, or 24 bits, is a critical factor. We therefore represent them in Figs. 6a, 6b, and 6c, respectively.

First of all, we look at the most strict condition, Fig. 6a ($l = 16$), we can see that for botnets with around 1,500 active bots, we obtain $T_I \approx 20$ minutes, which is sufficiently greater than the average attack duration, 5 minutes. Even for the case that all the botnets possess 3,000 active bots respectively, we have $T_I > 10$ minutes. We still have a reasonable buffer from the 5 minutes threshold.

Similarly, from Fig. 6b ($l = 17$), and Fig. 6c ($l = 24$), we can see that $T_I \approx 40$ minutes, $5,000$ minutes (around three and a half days), respectively. Comparative, we have a better, or much better accuracy than $l = 16$ case.

Based on the available knowledge of DDoS attacks, long time attacks are rare as it will uncover botnets, and therefore threaten the sustainability of the botnets.

Therefore, we can conclude that the proposed traceback scheme is feasible and effective in practice.

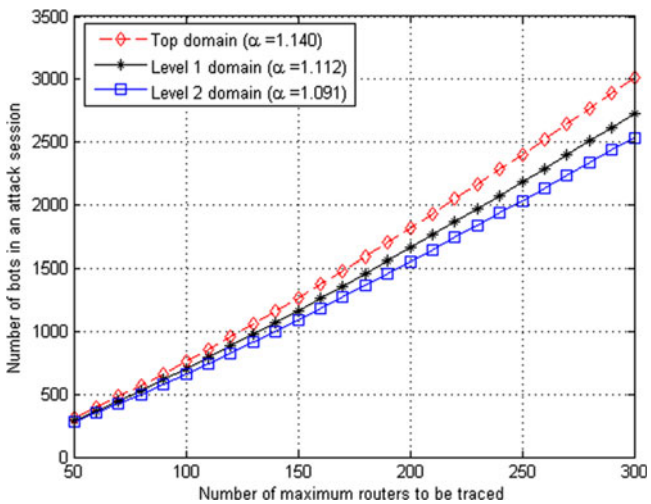


Fig. 5. The relationship between the number of total bots in an attack session and the number of routers to be traced.

6 Further Discussion

In this paper, we mainly focus on proposing a new traceback method to overcome the scalability problem in DPM based schemes, and demonstrating the effectiveness and feasibility of the MOD scheme. However, there are several other important aspects that we do not discuss in this paper. We list them here for interested readers.

- In the proposed MOD scheme, the MOD server is a critical component. Hackers may collaborate to disable or degrade the availability of the MOD server, e.g., using DDoS attacks. How to counter this kind of attack is an important issue for us. Preliminarily, we may use a distributed MOD system to counter attacks, or employ available DDoS mitigation methods to archive the same goal, such as mitigate DDoS attacks on MOD server by hosting the server in cloud, and take the advantage of cloud resource to beat DDoS attacks. Interested readers can refer to our recent work [4].
- Due to the huge amount of attack information in the MOD database, it is also a challenge on the performance of information retrieval from victims or other potential clients. Once again a distributed system of the MOD scheme can address this problem.
- As we have seen, one disadvantage of the proposed traceback method is the false positive and false negative issue cause by long time attacks. However, we can solve this problem by extending the length of marking space. For example, employing the existing mark coding techniques to divide a unique mark into multiple packet [6], [29]. Of course, this will increase the cost for traceback.
- We may be able to traceback to each and every possible bots in a global scenario by extending the proposed MOD scheme.

7 Summary and Future Work

In this paper, we propose a feasible DDoS attack source traceback scheme, the marking on demand scheme, based on the deterministic packet marking mechanism. In general, the proposed scheme fundamentally addresses the scalability problem of the existing DPM based traceback schemes. As a result, we can traceback every attack source (router) on the Internet, which is impossible for the previous traceback schemes. Our theoretical analysis and real world data set based experiments demonstrate that the proposed scheme is feasible.

In regards to future work, we first plan to extend the current work to improve the availability of the MOD server itself as it is a centralized system. Second, we expect to extend the proposed scheme to trace back to each every attack computer (but) by using multiple packets for

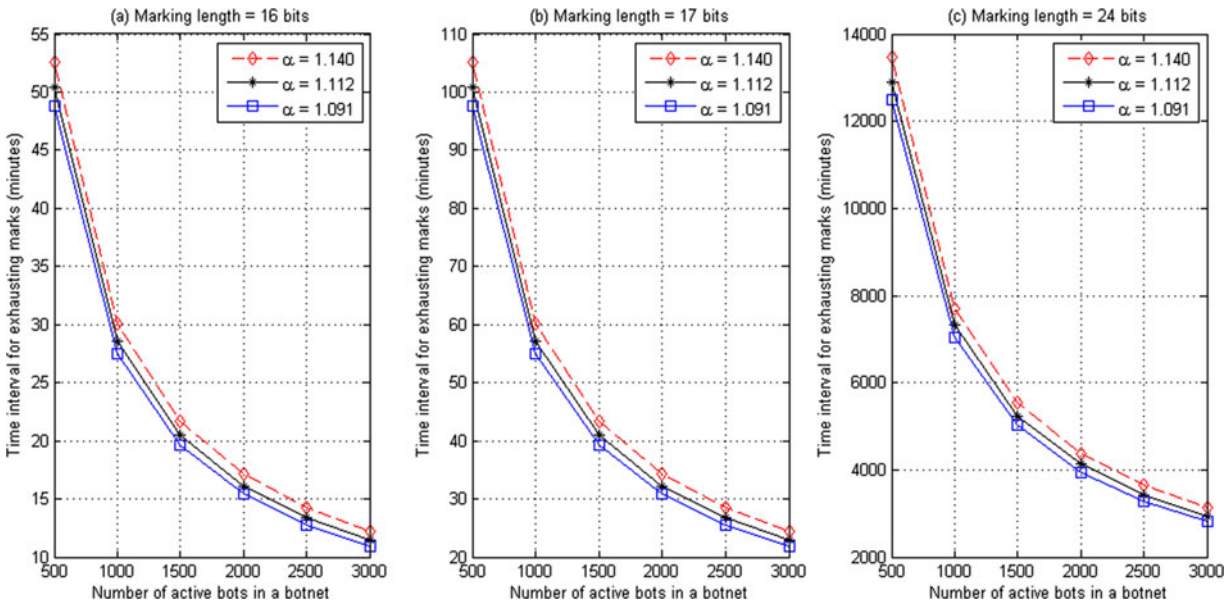


Fig. 6. The relationship among T_I and the number of concurrent attacking bots for different marking length.

marking coding. Thirdly, a thorough investigation on the MOD system is desired, such as the false positive rate and false negative rate of the MOD scheme. Finally, a real system prototype is planed to examine the efficiency of the proposed scheme in practice in the near future.

Acknowledgments

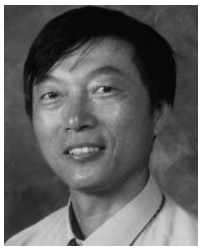
The authors would like to thank the EiC, Editor, and anonymous reviewers for their effort and insightful comments for the paper. The work of Dr. Yu was partially supported by the National Natural Science Foundation of China under Grant No 61379041.The work of Dr. Minyi Guo is partially supported by the National Natural Science Foundation of China under Grant Nos 61272099 and 61261160502. M. Guo is the corresponding author.

References

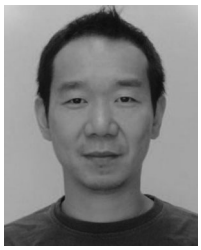
[1] T. Peng, C. Leckie, and K. Ramamohanarao, "Survey of network-based defense mechanisms countering the DoS and DDoS problems," *ACM Comput. Surv.*, vol. 39, no. 1, p. 3, 2007.
[2] S. Yu, W. Zhou, W. Jia, S. Guo, Y. Xiang, and F. Tang, "Discriminating DDoS attacks from flash crowds using flow correlation coefficient," *IEEE Trans. Parallel Distrib. Syst.*, vol. 23, no. 6, pp. 794–805, Jun. 2012.
[3] R. Chen, J.-M. Park, and R. Marchany, "A divide-and-conquer strategy for thwarting distributed denial-of-service attacks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 18, no. 5, pp. 577–588, May 2007.
[4] S. Yu, Y. Tian, S. Guo, and D. Wu, "Can we beat DDoS attacks in cloud?" *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 9, pp. 2245–2254, Sep. 2014.
[5] B. Al-Duwairi and G. Manimaran, "Novel hybrid schemes employing packet marking and logging for IP traceback," *IEEE Trans. Parallel Distrib. Syst.*, vol. 17, no. 5, pp. 403–418, May 2006.
[6] Y. Xiang, W. Zhou, and M. Guo, "Flexible deterministic packet marking: An IP traceback system to find the real source of attacks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 20, no. 4, pp. 567–580, Apr. 2009.
[7] S. Yu, W. Zhou, R. Doss, and W. Jia, "Traceback of DDoS attacks using entropy variations," *IEEE Trans. Parallel Distrib. Syst.*, vol. 22, no. 3, pp. 412–425, Mar. 2011.
[8] S. Yu, S. Guo, and I. Stojmenovic, "Fool me if you can: Mimicking attacks and anti-attacks in cyberspace," *IEEE Trans. Comput.*, vol. 64, no. 1, pp. 139–151, Jan. 2015.
[9] S. Yu, R. Doss, and W. Zhou, "Information theory based detection against network behavior mimicking DDoS attacks," *IEEE Commun. Lett.*, vol. 12, no. 4, pp. 319–321, Apr. 2008.
[10] S. Yu, *Distributed Denial of Service Attack and Defence*. New York, NY, USA: Springer, 2014.
[11] V. L. L. Thing, M. Sloman, and N. Dulay, "A survey of bots used for distributed denial of service attacks," in *Proc. IFIP Int. Inf. Security Privacy Conf.*, 2007, pp. 229–240.
[12] S. Yu, S. Guo, and I. Stojmenovic, "Can we beat legitimate cyber behavior mimicking attacks from botnets?" in *Proc. INFOCOM*, 2012, pp. 2851–2855.
[13] S. Savage, D. Wetherall, A. R. Karlin, and T. E. Anderson, "Practical network support for IP traceback," in *Proc. Conf. Appl., Technol., Archit., Protocols Comput. Commun.*, 2000, pp. 295–306.
[14] T. K. T. Law, J. C. S. Lui, and D. K. Y. Yau, "You can run, but you can't hide: An effective statistical methodology to trace back DDoS attackers," *IEEE Trans. Parallel Distrib. Syst.*, vol. 16, no. 9, pp. 799–813, Sep. 2005.
[15] A. Yaar, A. Perrig, and D. X. Song, "Fit: fast internet traceback," in *Proc. INFOCOM*, 2005, pp. 1395–1406.
[16] M. T. Goodrich, "Probabilistic packet marking for large-scale IP traceback," *IEEE/ACM Trans. Netw.*, vol. 16, no. 1, pp. 15–24, Feb. 2008.
[17] A. Belenky and N. Ansari, "IP traceback with deterministic packet marking," *IEEE Commun. Lett.*, vol. 7, no. 4, pp. 162–164, Apr. 2003.
[18] A. Belenky and N. Ansari, "On deterministic packet marking," *Comput. Netw.*, vol. 51, no. 10, pp. 2677–2700, 2007.
[19] X. Wang, S. Chen, and S. Jajodia, "Network flow watermarking attack on low-latency anonymous communication systems," in *Proc. IEEE Symp. Security Privacy*, Oakland, CA, USA, 2007, pp. 116–130.
[20] W. Jia, F. P. Tso, Z. Ling, X. Fu, D. Xuan, and W. Yu, "Blind detection of spread spectrum flow watermarks," in *Proc. INFOCOM*, 2009, pp. 2195–2203.
[21] S. Yu, W. Zhou, S. Guo, and M. Guo, "A dynamical deterministic packet marking scheme for DDoS traceback," in *Proc. IEEE Int. Conf. Global Commun.*, 2013, pp. 729–734.
[22] M. A. Rajab, J. Zarfoss, F. Monrose, and A. Terzis, "My botnet is bigger than yours (maybe, better than yours): Why size estimates remain challenging," in *Proc. 1st Conf. Hot Topics Understanding Botnets*, 2007, p. 5.
[23] D. Moore, C. Shannon, D. J. Brown, G. M. Voelker, and S. Savage, "Inferring internet denial-of-service activity," *ACM Trans. Comput. Syst.*, vol. 24, no. 2, pp. 115–139, 2006.
[24] V. L. L. Thing, M. Sloman, and N. Dulay, "A survey of bots used for distributed denial of service attacks," in *Proc. IFIP Int. Inf. Security Privacy Conf.*, 2007, pp. 229–240.

[25] A. Hackworth and N. Ianelli, "Botnets as a vehicle for online crime," *Int. J. Forensic Comput. Sci.*, vol. 2, no. 1, pp. 19–39, 2007.
[26] P. Wang, S. Sparks, and C. C. Zou, "An advanced hybrid peer-to-peer botnet," *IEEE Trans. Dependable Secure Comput.*, vol. 7, no. 2, pp. 113–127, Apr.–Jun. 2010.
[27] M. Bailey, E. Cooke, F. Jahanian, Y. Xu, and M. Karir, "A survey of botnet technology and defenses," in *Proc. Cybersecurity Appl. Technol. Conf. Homeland Security*, 2009, pp. 299–304.
[28] B. Stone-Gross, M. Cova, L. Cavallaro, B. Gilbert, M. Szydlowski, R. Kemmerer, C. Kruegel, and G. Vigna, "Your botnet is my botnet: Analysis of a botnet takeover," in *Proc. 2009 ACM Conf. Comput. Commun. Security*, 2009, pp. 635–647.
[29] G. Jin and J. Yang, "Deterministic packet marking based on redundant decomposition for IP traceback," *IEEE Commun. Lett.*, vol. 10, no. 3, pp. 204–206, Mar. 2006.
[30] D. Moore, C. Shannon, D. J. Brown, G. M. Voelker, and S. Savage, "Inferring internet denial-of-service activity," *ACM Trans. Comput. Syst.*, vol. 24, no. 2, pp. 115–139, 2006.
[31] Z. Chen and C. Ji, "An information-theoretic view of network-aware malware attacks," *IEEE Trans. Inf. Forensics Security*, vol. 4, no. 3, pp. 530–541, Sep. 2009.
[32] S. Yu, G. Gu, A. Barnawi, S. Guo, and I. Stojmenovic, "Malware propagation in large-scale networks," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 1, pp. 170–179, Jan. 2015.
[33] S. Shin, G. Gu, A. L. N. Reddy, and C. P. Lee, "A large-scale empirical study of conficker," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 2, pp. 676–690, Apr. 2012.
[34] M. Mitzenmacher, "A brief history of generative models for power law and lognormal distributions," *Internet Math.*, vol. 1, pp. 226–251, 2004.
[35] M. E. J. Newman, "Power laws, Pareto distributions and Zipf's law," *Contemporary Phys.*, vol. 46, pp. 323–351, 2005.

Shui Yu (M'05-SM'12) is currently a senior lecturer in the School of Information Technology, Deakin University. His research interest includes networking theory, cyber security, mathematical modelling, and big data. He has published two monographs and edited one book, more than 150 technical papers, including top journals and top conferences, such as *IEEE TPDS*, *IEEE TC*, *IEEE TIFS*, *IEEE TMC*, *IEEE TKDE*, *IEEE TETC*, and IEEE INFOCOM. He initiated the research field of networking for big data in 2014. His h-index is 18. He actively serves his research communities in various roles. He is currently serving the editorial boards of *IEEE Transactions on Parallel and Distributed Systems*, *IEEE Communications Surveys and Tutorials*, *IEEE Access*, and a number of other international journals. He has served more than 50 international conferences as a member of organizing committee, such as publication chair for IEEE Globecom 2015 and IEEE INFOCOM 2016, TPC co-chair for IEEE BigDataService 2015, IEEE ATNAC 2014 and 2015. He is a member of Deakin University Academic Board (2015-2016), a senior member of the IEEE, and a member of AAAS, the vice chair of Technical Subcommittee on Big Data Processing, Analytics, and Networking of IEEE Communication Society.



Wanlei Zhou (SM'09) received the PhD degree from The Australian National University, Canberra, Australia, in October 1991. He also received the DSc degree from Deakin University, Victoria, Australia in 2002. He is currently the chair professor and head of the School of Information Technology, Deakin University, Melbourne, Australia. His research interests include distributed and parallel systems, network security, mobile computing, bioinformatics and e-learning. He is a senior member of the IEEE.



Song Guo (M'02-SM'11) received the PhD degree in computer science from the University of Ottawa, Canada in 2005. He is currently a senior associate professor at the School of Computer Science and Engineering, University of Aizu, Japan. His research interests are mainly in the areas of protocol design and performance analysis for reliable, energy-efficient, and cost effective communications in wireless networks. He is an associate editor of the *IEEE Transactions on Parallel and Distributed Systems* and an editor of *Wireless Communications and Mobile Computing*. He is a senior member of the IEEE and the ACM.

Minyi Guo received the PhD degree in computer science from the University of Tsukuba, Japan. He is currently a chair professor and a head of the Department of Computer Science and Engineering, Shanghai Jiao Tong University, China. He received the national science fund for distinguished young scholars from NSFC in 2007. His research interests include parallel and distributed computing, compiler optimizations, embedded systems, pervasive computing, and bioinformatics. He has more than 300 publications in major journals and international conferences in these areas. He is on the editorial board of the journals *IEEE Transactions on Parallel and Distributed Systems* and *IEEE Transactions on Computers*. He is a senior member of the IEEE, a member of the ACM, IEICE IPSJ, and CCF.