Journal of Information Processing Vol.22 No.4 609–616 (Oct. 2014)

[DOI: 10.2197/ipsjjip.22.609]

Regular Paper

Transparent Probabilistic Packet Marking

Masayuki Okada[1,a) Nasato Goto[2,b) Akira Kanaoka[3,c) Eiji Okamoto[2,d)

Received: November 29, 2013, Accepted: June 17, 2014

Abstract: Probabilistic Packet Marking (PPM) is known to be one of the better defense methods against Denial of Service (DoS) attacks. However, most of the routers on the Internet are not yet ready for PPM. Before a new router that has the PPM function can be deployed, several challenges such as cost, operation, and availability must first be resolved. In this paper, we propose a device for transparent PPM that makes the target router PPM-capable. The device does not change the existing configuration of the router nor do existing routers have to be replaced. We implemented and evaluated our proposed device on Linux with excellent results.

Keywords: IP traceback, Denial of Service attack

1. Introduction

In Internet Protocol (IP) networks, communication is achieved by sending packets from a source host to a destination host. Most of these communicating hosts belong to different networks and so they cannot communicate directly. Therefore, routers are used to relay their communications.

A router decides how to carry out its routing activities according to the IP addresses of the source and destination hosts. Although the header of an IP packet has the IP addresses of the source and destination hosts, there is no information about the routers that relay the packet. As a result, the receiver of the packet cannot distinguish which routers were used to relay the packet from the source host.

There are several scenarios in which knowing the route a packet has taken can be beneficial. One such scenario is the case where the routes between a target host and Distributed Denial of Service (DDoS) attackers need to be determined. In this scenario, if information about the routes is known, DDoS attack packets can be dropped before arriving at the target host. Detection of communication bottlenecks on multiple routes is another scenario.

IP traceback is a mechanism that gives route information about packets. There are many studies dealing with IP traceback. The widespread areas of IP traceback studies can be seen in the taxonomical work on IP traceback done by Takahashi et al. [1].

Probabilistic Packet Marking (PPM) is an IP traceback technique that overwrites router information onto packets. Since each relaying router writes its own information on the relayed packets probabilistically, the receiver of the packets can gather information about the routers on the routes taken by the packets. This enables the receiver to reconstruct information about the routes from the senders to a receiver.

PPM was originally proposed by Savage et al. in 2000 [2]. The seminal work by Savage et al. spawned many followers of PPM [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18]. In recent times, obtaining optimum probability for marking is achieved by considering the marking load and the network topology [11]. PPM studies are currently advancing from the fundamental research stage to practical application.

However, several challenges need to be overcome to facilitate the deployment and operation of PPM technologies. One of these challenges is the question of how to apply PPM to routers that are currently in operation. To apply PPM to routers that are currently in operation, replacing or upgrading its operating system or firmware, or replacing the router itself to a PPM-ready router may be required. However, for core routers, this is complicated. In addition, it incurs higher costs.

In this paper, a transparent probabilistic packet marking device that can be utilized without replacing or upgrading routers that are currently in operation is proposed. Our proposed device can facilitate PPM at layer 1 (Network Interface Layer) as a repeater or at layer 2 (Internet Layer) as a bridge/switch in TCP/IP in a transparent manner. In addition, a tolerant system for a transparent probabilistic packet marking device, and a method of automatic configuration of IP addresses for marking, are proposed. In this paper, we present a prototype of our proposed device and evaluate and compare its performance in several configurations for non-PPM routers.

The remainder of this paper is organized as follows: In Section 2, IP traceback and PPM techniques are described. Existing challenges to PPM in terms of operational and deployment aspects are outlined in Section 3. Our proposed transparent PPM device is presented in Section 4, and its implementation and performance evaluation discussed in Section 5. Finally, Section 6 concludes this paper.

[1] Japan Network Information Center, Chiyoda, Tokyo 101–0047, Japan
[2] University of Tsukuba, Tsukuba, Ibaraki 305–8573, Japan
[3] Toho University, Funabashi, Chiba 274–8510, Japan
[a) okadams@nic.ad.jp
[b) goto@cipher.risk.tsukuba.ac.jp
[c) akira.kanaoka@is.sci.toho-u.ac.jp
[d) okamoto@risk.tsukuba.ac.jp

2. IP Traceback and Probabilistic Packet Marking

2.1 Classification of IP Traceback Techniques

Takahashi et al. created taxonomy of IP traceback techniques. They divided the techniques into two broad categories: Intra-AS traceback and Inter-AS traceback. They further divided Intra-AS traceback into Traffic Monitoring and Packet Monitoring subcategories. In their paper, over 40 studies were grouped into a total of eight categories.

Packet Monitoring deals with the determination of the route taken by a packet. This category is divided into Packet Marking, Messaging, Packet Logging, and Hybrid and Modified Routing sub-categories. Probabilistic Packet Marking, which marks information onto packets probabilistically, belongs to the Packet Marking subcategory.

2.2 Probabilistic Packet Marking

Packet Marking methods write information that enables route re-construction by receivers using specific fields in the IP header. Victims of DDoS attacks or routers on the routes used to carry out the attacks can reconstruct route information by extracting partial information from the packets received or relayed. In early packet marking methods, only information such as the IP addresses of routers and other related information was marked on packets. Packet marking methods have subsequently become more generalized and now deal with various types of information, not only IP addresses. Packet marking methods can also be divided into two types: Deterministic Packet Marking (DPM) [19], [20] and Probabilistic Packet Marking (PPM). DPM marks information onto packets based on a predefined timing metric; for example, the rate of traffic or the number of packets. PPM marks information onto packets probabilistically.

In contrast to PPM, DPM is regarded as easy to implement because DPM does not need the overhead required by probabilistic operations. On the other hand, thanks to its probabilistic property, in PPM methods, packets that have already been marked at an earlier router on a path can be sent to destination hosts without overwriting. It has been conjectured that if DPM is used, an attacker can avoid the marks by estimating the marking timing.

Advantages of packet marking methods include the fact that there is no need for a centralized management system, no additional traffic, and the possibility of automation of traceback.

2.3 PPM Method by Savage et al.

In the original PPM method proposed by Savage et al., the routers along the route of attacks marked their own information onto packets probabilistically. Theirs was the first method to mark information onto packets probabilistically.

A router *A* that has some static probability setting marks its own IP address onto the Identification field of IP version 4 (IPv4) headers. Because the Identification field is only 16 bits, despite the 32 bit IP address in IPv4, 64 bit data, which comprise bit interleaving with 32 bit IP address and the 32 bit hash value of the IP address, are divided into eight fragments of 8 bits each. Eight bits are used as fragmented data, 3 bits as the offset of fragments, and 5 bits as the distance metric (number of hops) from the marked router, in the 16 bit identification field. Those routers that do not mark the packets increment the distance value on the packets, and then relay them.

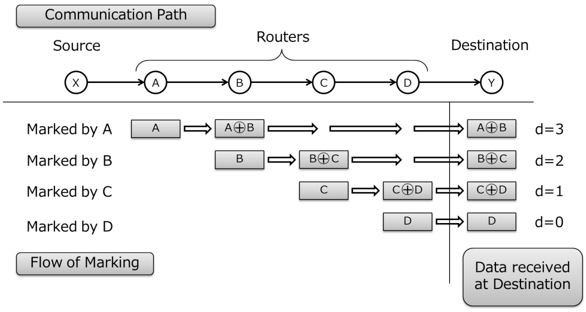

Fig. 1 Receiving a marked packet via Savage et al.'s method.

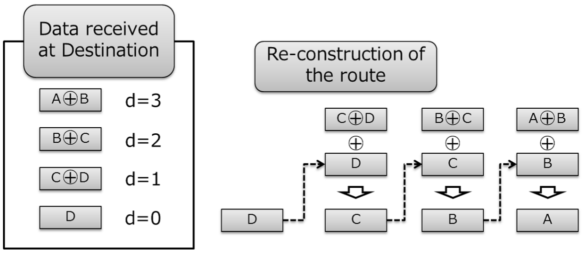

Fig. 2 Rebuilding path information using Savage et al.'s method.

The characteristics of Savage et al.'s original method not only include marking of single router information, but also making exclusive OR with the next-hop router information. This characteristic allows the receiver to recognize the edges between two routers that were used along the route of the packets. They called this "Edge Sampling" (**Fig. 1**).

To reconstruct the routes, the receiving host or router gathers the marked packets and reconstructs them based on its distance information d. First, it focuses on the packets of $d = 0$. Since the exclusive OR operation was not carried out on these packets, the information written in the fragmented data is the data about the nearest router. Next, based on the IP address obtained for the nearest router, it focuses on the packets of $d = 1$. Fragmented data is gathered and the exclusive OR operation performed, after which the IP address of the next router is obtained. After $d = 2$, all IP addresses along the route can be obtained using this method recursively (**Fig. 2**).

2.4 PPM Implementation and Optimum Marking Probability

In Savage et al.'s original work [2], the discussion about the required number of packets and other evaluations were made based on the marking probability $p = 1/25$. The same probability is used in several other studies [4], [5], [6], [8]. Although these studies mentioned the burden incurred by packet marking on the routers, optimization of marking probability is not discussed to any great extent. Okada et al. [11] implemented the PPM function in the Linux kernel and evaluated its performance. Their results indicated that there was no significant packet marking burden. We can therefore discuss optimization by solely focusing on minimization of the number of packets required to reconstruct routes.

3. Obstacles to the Application of Probabilistic Packet Marking

Several studies done on PPM purport to make PPM more efficient by reducing the number of packets required to reconstruct routes. Optimization and implementation are also discussed in some studies. Although research on PPM itself has advanced, there are many deployment and operational obstacles that remain to be overcome. For example, the following must be considered:

- PPM-unsupported routers in a network
  - Handling of unmarked packets at receiving (destination) hosts
  - Adaptiveness of joining or leaving PPM-supported routers
  - Actions after reconstruction of routes
- Application of PPM to routers that are currently in operation
- Uniformity of probabilities in every router
- Interoperability among the several PPM methods
- Legal aspects

In this paper, our focus is on the "Application of PPM to routers that are currently in operation." At present, to apply PPM to routers that are currently in operation, their OS or firmware have to first be updated or replaced or, even more, the router itself may have to be replaced by one that has a PPM-ready OS or firmware. However, operators or decision makers of Internet Service Providers (ISPs) may wish to avoid these measures due to the cost involved in the upgrading and replacing process. Therefore, we propose a transparent PPM device that can be implemented without the need to upgrade or replace the routers that are currently in operation.

4. A Device for Transparent Probabilistic Packet Marking

Routers mark their own IP address information onto packets or increment the distance information contained in packets. A transparent PPM (or TPPM, in the following figures) device is connected between routers, and marks and increments as a proxy for the PPM-unsupported router with which it is associated. This transparent PPM device does not route packets and cannot be seen in the IP layer.

The basic way in which a transparent PPM device is deployed is depicted in **Fig. 3**. The transparent PPM device is deployed and connected between two routers. It then carries out marking or incrementing on the behalf of one or more of the routers to which it is connected.

There are two ways in which such a transparent PPM device can be tangibly realized: 1) As a repeater that relays Ethernet frames without any changing of the Ethernet frame header and its data except for changes to the Identification field of the IPv4 header; or 2) as a bridge or a switch that exchanges Ethernet frames.

The transparent PPM device can mark and increment in either of two configurations: 1) As an agent for a single router, or 2) as an agent for multiple routers. In the former case, marking and incrementation is simple; the device can simply apply PPM to packets coming from an input port. However, in the latter case, the device has to focus on output ports after relaying and decide whether to mark or increment based on each input port and output port pair. Therefore, the technical difficulties associated with the latter type of transparent PPM device is much higher.

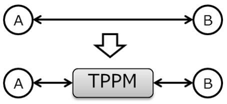

Fig. 3 Deployment of transparent PPM device.

Thus, since a router has at least two ports (interfaces), if we want a router to be fully PPM-supported using a transparent PPM device, we have to deploy multiple transparent PPM devices depending on the number of interfaces. This still remains a problem in the effort to support PPM without replacing routers that are currently in operation. Because there are several deployment patterns for transparent PPM devices and some deployment patterns result in only partial PPM support, the actual probabilities of a packet being marked vary according to the type of deployment. (These probabilities are described in detail below.)

In the normal case using Savage et al.'s method if implementation is achieved literally, marking of a packet occurs twice: Marking is first carried out by the first router A, an exclusive OR operation ($A \oplus B$) is then carried out by a second router B that is next to the first router A. However, if we use the transparent PPM device with Savage et al.'s method and the device carries out marking and incrementation on both sides (router A and B) of the device, the exclusive ORed data $A \oplus B$ can be marked onto packets in one action. This decrease in the number of actions needed for marking is one advantage of our proposed device.

If a router which have n interfaces to be full PPM ready by proposed devices, we have to deploy n devices around the router. It may causes high deployment costs. In such a case, replacing the router to a new PPM ready router is a better solution. If PPM function is required to specific interfaces or links on the router, proposed device can provide good solution from an aspect of deployment cost.

4.1 Transparent PPM Repeater

A transparent PPM repeater directly overwrites the Identification field in the IPv4 header without exchanging or changing the Ethernet frames. The repeater can have several ports and can be connected to several routers or other devices. The frames that come to the repeater on one port are relayed to all other ports.

4.2 Transparent PPM Bridge/Switch

A transparent PPM bridge or switch relays upper layer (IP layer) packets by exchanging Ethernet frames. A transparent PPM bridge has two Ethernet ports. A transparent PPM switch has more than two Ethernet ports and relays packets to a specific port based on the destination MAC address of the Ethernet frame.

4.3 Points to Note in Transparent PPM Device Utilization

If a router supports PPM without using the proposed transparent PPM devices, packets from every interface can be marked appropriately. However, when the proposed transparent PPM devices are used but the support by the devices is not full, for example, only two devices are connected to a router that has three interfaces, the frequency of appearance of packets marked by the router will differ according to packet direction. This asymmetric property affects the number of packets required to be reconstructed.

That is, if we consider a device such as Savage et al.'s "Edge Sampling" device and the device just works at the "Edge," it is very natural in Savage et al.'s method from the aspects of sampling edges (not router information).

4.4 Marking Probability on Transparent PPM Devices

In this subsection, probabilities and marked information are considered in several configurations in which the transparent PPM devices are used. For simplification, we assume that the transparent devices have only two interfaces and are connected to two routers by devices that we call X and Y. Thus, the marking probability of the interface i of the router X is expressed as $P_X^{(i)}$, and the number of packets sent from interface i of the router X is expressed as $S_X^{(i)}$. Then, the marking probability P_X of router X is expressed as follows:

$$P_X = \frac{\sum_i P_X^{(i)} S_X^{(i)}}{\sum_i S_X^{(i)}}$$

4.4.1 Configuration in Which Device Acts as a PPM for a Router X

The probability of the device marking is $P_X^{(i)}$, where the value of the mark is X and the value of the distance is d that is zero.

The probability of the device not marking is $1 - P_X^{(i)}$, then it increments d.

4.4.2 Configuration in Which Device Acts as a PPM for an Edge $X \oplus Y$

If the device marks, the probability of packets being sent from X is

$$P_X^{(i)} \frac{\sum_{w \neq i} \left(1 - P_Y^{(w)}\right) S_Y^{(w)}}{\sum_{w \neq i} S_Y^{(w)}},$$

the probability of packets being sent from Y is

$$P_Y^{(j)} \frac{\sum_{w \neq j} \left(1 - P_X^{(w)}\right) S_X^{(w)}}{\sum_{w \neq i} S_X^{(w)}}.$$

The value of the marking is $X \oplus Y$, and the value of the distance is $d = 1$.

4.4.3 Configuration in Which Device Acts as a PPM for Routers X and Y, and an Edge $X \oplus Y$

There are two ways to mark packets sent from X; marking as X, or as Y. If the device marks as X, the probability is

$$P_X^{(i)} \frac{\sum_{w \neq j} \left(1 - P_Y^{(w)}\right) S_Y^{(w)}}{\sum_{w \neq j} S_Y^{(w)}},$$

where X and Y are connected using interface i of X between internet j of Y. The value of the marking is $X \oplus Y$, and the value of the distance is $d = 1$.

If the device marks as Y, the probability is

$$\frac{\sum_{w \neq i} P_Y^{(w)} S_Y^{(w)}}{\sum_{w \neq i} S_Y^{(w)}}.$$

The value of the marking is Y, and the value of the distance is $d = 0$.

Marking packets sent from Y is similar to the case in which they are sent from X; if the device marks as Y, the probability is

$$P_Y^{(j)} \frac{\sum_{w \neq j} \left(1 - P_X^{(w)}\right) S_X^{(w)}}{\sum_{w \neq i} S_X^{(w)}}.$$

The value of the marking is $X \oplus Y$, and the value of the distance is $d = 1$.

If the device marks as X, the probability is

$$\frac{\sum_{w \neq i} P_X^{(w)} S_X^{(w)}}{\sum_{w \neq i} S_X^{(w)}}.$$

The value of the marking is X, and the value of the distance is $d = 0$.

4.5 Tolerant Transparent PPM System

4.5.1 Problem on Transparent PPM Device

Although deployment of Transparent PPM devices can bring benefits, Transparent PPM devices may increase the failure rate of a network. The failure rate of a system consisting of 2 routers connected by a cable is given by each failure rate of routers and a cable. If we add a Transparent PPM device between these two routers, the Transparent PPM device is going to be a single point of failure in the system.

For example, let us consider the failure rate of a system on Fig. 3, where the failure rate of router A, router B, the cables used in the system and the Transparent PPM device are α_A, α_B, α_{Ca} and α_T, respectively. γ, the failure rate of the system which does not include the Transparent PPM device, is $\gamma = 1-(1-\alpha_A)(1-\alpha_B)(1-\alpha_{Ca})$. The failure rate of the system including the Transparent PPM device is, $1-(1-\gamma)(1-\alpha_{Ca})(1-\alpha_T)$. This shows that the failure rate of the Transparent PPM device α_T has a direct impact on the failure rate of the system.

4.5.2 Applying Rapid Spanning Tree Protocol to Transparent PPM Devices

Rapid Spanning Tree Protocol (RSTP) is a one of protocols of Spanning Tree Protocol (STP) used for tolerance of failure and avoiding loops in the communication network. Although the time required for switching is around 50 seconds in usual STP, RSTP can bring the switching in seconds.

Using switches which support RSTP and Virtual LAN (VLAN), tolerance of Transparent PPM device can be achieved. **Figures 4** and **5** show the construction of the tolerant Transparent PPM system. In these figures, Sw means switches which sup-

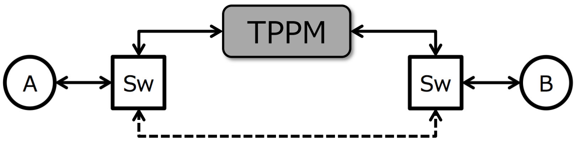

Fig. 4 Tolerant transparent PPM system: Half side TPPM.

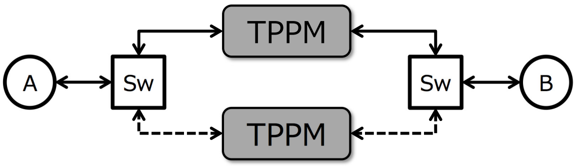

Fig. 5 Tolerant transparent PPM system: Both side TPPM.

ports RSTP and VLAN. In normal phase, the communication is achieved using the path including Transparent PPM device (or the preferred path). If the device goes to failure, a switch of the sender side can detect that it is not working, then change the path to another one.

4.5.3 Failure Rate using Tolerant Transparent TPPM System

The failure rate of the system in Fig. 4 is,

$$1-(1-\alpha_A)(1-\alpha_B)(1-\alpha_{Ca})^2(1-\alpha_{Sw})^2 \\ \{1-(1-(1-\alpha_{Ca})^2(1-\alpha_T))\alpha_{Ca}\},$$

where α_{Sw} is the failure rate of a switch. Also, the failure rate of the system in Fig. 5 is,

$$1-(1-\alpha_A)(1-\alpha_B)(1-\alpha_{Ca})^2(1-\alpha_{Sw})^2 \\ \{1-(1-(1-\alpha_{Ca})^2(1-\alpha_T))^2\}.$$

5. Development of a Prototype Transparent PPM Device

5.1 Implementation of a Transparent PPM Repeater

We actualized our proposed transparent PPM device as a transparent PPM repeater. The PPM function was added to the Linux kernel. Therefore, there are several techniques used to achieve high speed PPM without reducing relay speed.

5.1.1 Re-calculation of IP Header Checksum

After marking of PPM is overwritten or distance data is incremented, the checksum of the IP header has to be re-calculated. Processing of Ethernet frames and IP datagrams are done separately in the Linux kernel. The PPM primarily operates during the processing of Ethernet frames; calling the IP datagram function there incurs a high cost. Therefore, the FastCsum function in the processing part of the IP datagram was transplanted into the processing part of the Ethernet frames. This transplantation results in high-speed processing of IP header checksum re-calculation.

5.1.2 Flexibility for Marking Probability Setting

Instead of embedding marking probability configuration into the kernel, our implementation uses a flexible marking probability setting; that is, marking probability can be set from user-level. The marking probability can be written to a file using sysfs, a virtual file system provided after Linux kernel 2.6. User-level settings can expand not only flexible marking probability, but also result in flexible PPM methods. We can insert several PPM methods into the kernel and can choose PPM methods from user-level.

5.1.3 Support for Automatic IP Address Configuration

Although automatic IP address configuration was not achieved in this implementation, the related kernel parts have already been developed and are ready to be added to this automatic scheme using a switching operation.

5.2 Evaluation of the Transparent PPM Repeater Developed

In this part, the performance of PPM function is evaluated. The purpose of the evaluation is to measure whether PPM incurs overhead or not. An evaluation environment was built to evaluate the prototype transparent PPM repeater developed. Our evaluation focused on the throughput of the device. A packet sender, a packet receiver, and the transparent PPM repeater were connected in series (**Fig. 6**). The OS of the packet sender was Linux (CentOS), and equipped with a 1 Gbps NIC. The OS of the packet receiver was Mac OS X, and equipped with a 1 Gbps NIC. The OS of the transparent PPM repeater was Linux (Debian GNU/Linux 6.0.1) on IBM x306m (Intel Pemtium 4 531 (3.0 GHz), 2GM RAM, Broadcom 5721 NIC (1 Gbps)).

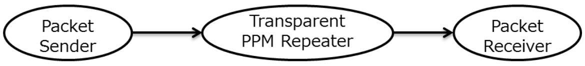

Fig. 6 Evaluation environment for the transparent PPM repeater.

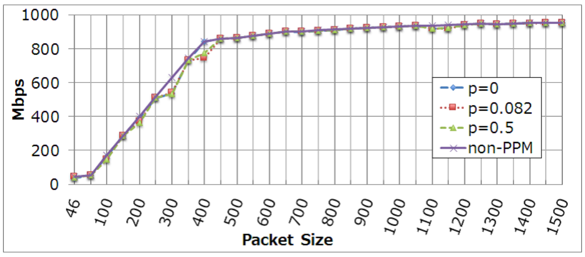

Fig. 7 Throughput in each parameter.

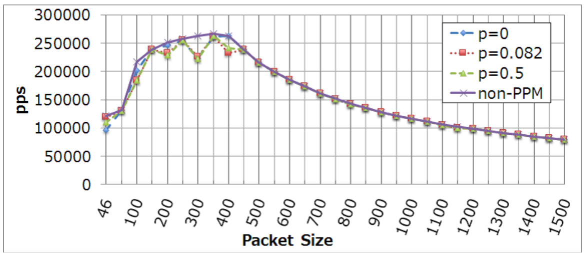

Fig. 8 Packet per seconds in each parameter.

The following four configurations were measured for comparison:

- non-PPM (normal kernel)
- PPM with $p = 0$ (no marking, only incrementing d)
- PPM with $p = 0.082$
- PPM with $p = 0.5$

$p = 0.082$ is the probability that is claimed as the optimum probability in the Internet topology by Okada et al. [11].

The throughput and pps (packet per second) of each of the four configurations was measured for several packet sizes ranging from 46 to 1,500 (based on the largest MTU in Ethernet). Each throughput and pps were measured 10 times and averages were calculated. **Figures 7** and **8** graphically depict our results.

There were no large differences among the results for the four configurations. This indicates that the action by PPM does not incur any significant overhead. Thus, the performance of the prototype configurations was good even though PPM was in operation.

Further, the resulting graphs for the $p = 0$ and non-PPM configurations had virtually the same shape, and those for the $p = 0.082$ and $p = 0.5$ configurations had virtually the same shape. The small difference between these two shapes represents the overhead incurred during PPM marking.

The minimum throughput was 35.189 Mbps, for $p = 0$ and a packet size of 46 bytes. The maximum throughput was 951.825 Mbps, for $p = 0$ and a packet size of 1,500 bytes. The

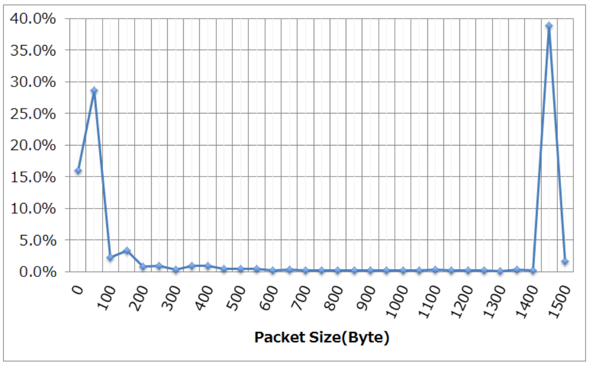

Fig. 9 Distribution of packet sizes in our laboratory.

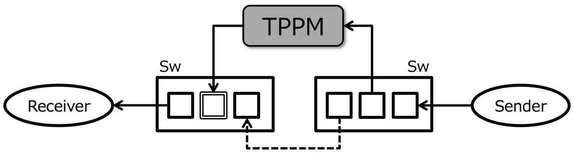

Fig. 10 Evaluation environment for tolerant transparent PPM system.

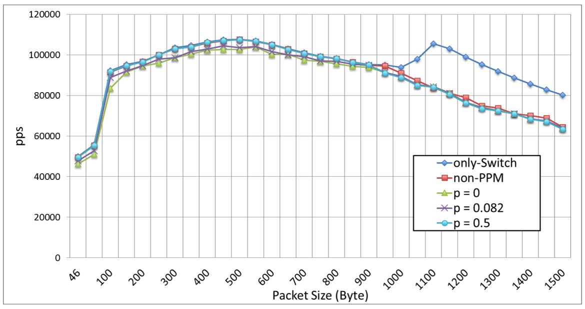

Fig. 11 Packet per seconds in each parameter in tolerant transparent PPM system evaluation.

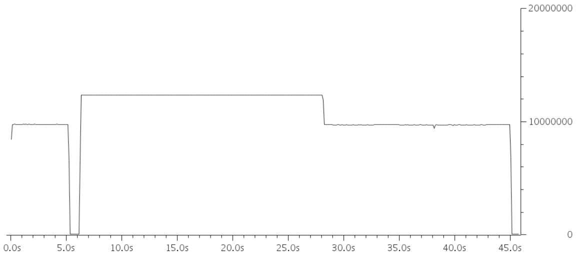

Fig. 12 Network traffic of receiver during the tolerant transparent TPPM system evaluation.

minimum pps was 79,265.3 for $p = 0.082$ and a packet size of 1,500 bytes. The maximum pps was 266,246.8 for non-PPM and a packet size of 350 bytes. It can be seen that every throughput and pps for the same size was quite near the same value from Figs. 7 and 8. Thus, even though the hardware specification used for the prototype transparent PPM bridge was not high, the throughput and pps for the larger packet size were quite high.

In the real world, the distribution of packet sizes is roughly divided into two types: very small size and very large size. **Figure 9** shows the distribution of packet sizes in our laboratory captured over a two-week period.

On the basis of its distribution, we can estimate the performance of the prototype device to be in the range 492–494 Mbps in a real-world environment.

5.3 Evaluation of the Tolerant Transparent PPM System Developed

The proposed tolerant Transparent PPM system is also evaluated. An evaluation environment was similarly built to evaluate the prototype transparent PPM repeater developed. This evaluation focused on the throughputs and the required time for switching of the device. A packet sender, a packet receiver, and the transparent PPM repeater were connected in series (**Fig. 10**). The OS of the packet sender was Mac OS X, and equipped with a 1 Gbps NIC. The OS of the packet receiver was Linux (Cent OS), and equipped with a 1 Gbps NIC. The OS of the transparent PPM repeater was Linux (Debian GNU/Linux 6.0.1) on FUJITSU ESPRIMO D750/A (Intel Core i5 (3.20 GHz), 4 GM RAM, Intel PRO/1000 NIC (1 Gbps) and BUFFALO LGY-PCI-GT NIC (1 Gbps)). Switches were NETGEAR GS108T (8port, 1 Gpps/port).

A number of packets per seconds is also evaluated like Section 5.2. Each parameter and way of obtaining scores are also same in the evaluation in Section 5.2, excepting adding "only-switch" which is the environment only including one switch between the sender and the receiver, to evaluate the overhead between the proposed tolerant transparent PPM system and one switch. In this comparison, the proposed tolerant transparent PPM system is considered as a just one network equipment even it consists of several devices. **Figure 11** graphically depict our results.

Also, the time required to switch to an unfailing path are observed. From the sender, UDP packets are sent using netperf. During the sender sending the data, an Ethernet cable on the switch of receiving TPPM output (double lined in Fig. 10) is disconnected as a simulation of failing of TPPM to observe switching to an unfailing path. **Figure 12** shows the one of network traffic of receiver during the evaluation. Marking probability $p = 0.082$ is used in the evaluation.

Average time of switching to an unfailing path is 0.91 seconds with 10 times evaluation. We can see its fastness to switch. Also, we can observe each event in Fig. 12: Occurring failure, switching to unfailing path, and switching back to TPPM side after recovering TPPM side failure. Around 5 sec, we can see the failure has occurred. Also switching to unfailing path in around 6 sec, switching back to TPPM side after recovering in around 28 sec. The difference between the traffic shows ones for TPPM side (around 10,000,000 bytes) and ones for unfailing side (around 120,000,000 bytes).

6. Future Works

6.1 Automatic IP Address Configuration by Observing ARP Packets

When the transparent PPM devices are being deployed, configuration of marking values and probabilities is required. We can automate this configuration by observing ARP packets.

In TCP/IP communication, after IP packet data are prepared to be sent to routers or hosts, an Ethernet frame is constructed with source and destination MAC addresses. If the router or host does not know the destination MAC address, ARP is used to obtain the destination MAC address using a query such as "Who has IP address ***.***.***.***," which gets a reply such as "The MAC address of IP address ***.***.***.*** is ##:##:##:##:##:##." An ARP query is sent to broadcast and every device that is directly connected in the Ethernet receive the query. After receiving a reply, the router or the host keeps the information in its cache for a specified period of time.

If we observe the ARP packets at a repeater, bridge, or switch, we can obtain the IP addresses to MAC addresses mapping table. In particular, if the device is connected between only two routers, the IP addresses coming from the routers on both sides can be obtained. Because we can assume that there are very little IP packets that have their source or destination as the routers, if several replies that state the same MAC address are observed, the MAC address might be that of the router. Therefore, we can see a bias according to the frequency of a pair of IP address and MAC address. Basically, since a router rarely sends packets by itself, a low frequency pair of IP address and MAC address can be assumed to be the correct IP address and MAC address of the router.

Therefore, after observing ARP packets over a specified period of time, the device can extract the IP addresses used as marking information and then automatically configure marking. For example, since Cisco routers initially keep values in their cache for four hours (14,400 s), we can automatically finish configuration after a four-hour period of observation.

However, this method is not secure. To achieve automatic configuration, we have to consider authentication of the device and routers.

6.2 Measuring Delay

Evaluation in Section 5 are mainly focused on performance of proposed device and system. One more aspect of evaluation is the delay caused by the proposed device and system. Actually, measuring delay on TCP/IP network in a rigorous manner is hard. The most significant factor is synchronizing time between a packet sender and a receiver. In Section 5 evaluation, the performance of the developed device and the other ones have similar shapes of graphs. If we have significant delays by the developed device, the shape of graphs might be different. From this point of view, it can be estimated that the developed device cause no delay or small delay that is negligible on average.

However, delay by the proposed device and system also has to be evaluated.

6.3 Cost of Operation

The discussion of deployment costs is partially discussed before. We also have to consider the cost of operation for the proposed device or the proposed system. For example, new operation is required to the device or the system adding to currently working devices.

7. Conclusion

Probabilistic Packet Marking (PPM), an IP traceback method, is rapidly progressing to the deployment and operational stages from the fundamental research stage. To overcome one of the challenges involved with its deployment and operation, in this paper, we proposed a transparent PPM device that can be implemented without the need to upgrade or replace routers that are currently in operation. The transparent PPM device is used as a proxy to carry out PPM functions on behalf of the operating routers. Since PPM actions are processed at the IP layer of the TCP/IP protocol stack, the proposed device is transparent to the IP layer.

We developed and implemented a prototype of our proposed device in the Linux kernel. The results of evaluations of its bridging performance indicated a high throughput of over 900 Mbps for larger packet sizes. Also we proposed and developed the tolerant transparent PPM system. The results of the evaluation show the high performance and low impact of failure of Transparent PPM device. These results will serve to enhance the deployment and operational stages of PPM studies.

References

[1] Takahashi, T., Hazeyama, H., Miyamoto, D. and Kadobayashi, Y.: Taxonomical Approach to the Deployment of Traceback Mechanisms, *Proc. 2011 BCFIC Riga* (2011).

[2] Savage, S., Wetherall, D., Karlin, A.R. and Anderson, T.: Practical Network Support for IP Traceback, *Proc. ACM SIGCOMM*, pp.205–306 (2000).

[3] Peng, T., Leckie, C. and Ramamohanarao, K.: Survey of Network-based Defense Mechanisms Countering the DoS and DDoS Problems, *ACM Computing Surveys*, Vol.39, No.1 (2007).

[4] Song, D. and Perrig, A.: Advanced and Authenticated Marking Schemes for IP Traceback, *Proc. IEEE INFOCOM*, pp.876–886 (2001).

[5] Dean, D., Franklin, M. and Stubblefield, A.: An Algebraic Approach to IP Traceback, *Proc. Network and Distributed System Security Symp.* (*NDSS*), pp.3–12 (2001).

[6] Okazaki, N., Kawamura, S. and Park, M.: A Study of an Efficient Method for Re-construction of Path in the DoS Attacks, *IPSJ Journal*, Vol.44, No.12, pp.3197–3201 (2003).

[7] Law, T.K.T., Yau, D.K.Y. and Lui, J.C.S.: An Effective Statistical Methodology to Traceback DDoS attackers, *IEEE Trans. Parallel Distrib. Syst.*, Vol.16, No.9, pp.799–813 (2005).

[8] Goodrich, M.T.: Probabilistic Packet Marking for Large-scale IP Traceback, *IEEE/ACM Trans. Networking*, Vol.16, No.1, pp.15–24 (2008).

[9] Durresi, A., Paruchnri, V., Barolli, L., Kannan, R. and Lyengar, S.S.: Efficient and Secure Autonomous System Based Traceback, *Journal of Interconnection Networks*, Vol.5, No.2, pp.151–164 (2004).

[10] Paruchnri, V., Durresi, A. and Barolli, L.: FAST: Fast Autonomous System Traceback, *Proc. 21st International Conference on Advanced Networking and Applications* (*AINA '07*), pp.498–505 (2007).

[11] Okada, M., Kanaoka, A., Katsuno, M. and Okamoto, E.: Probability Estimation for Probabilistic Packet Marking, *IPSJ Journal*, Vol.52, No.9 (2011).

[12] Kanaoka, A., Okada, M., Katsuno, M. and Okamoto, E.: Probabilistic Packet Marking Method Considering Topology Property for Efficiently Re-building DoS Attack Paths, *IPSJ Journal*, Vol.52, No.3 (2011).

[13] Okada, M., Kanaoka, A., Katsuno, Y. and Okamoto, E.: 32-bit AS Number Based IP Traceback, *Proc. 5th International Workshop on Advances in Information Security* (*WAIS-2011*) (2011).

[14] Liu, J., Lee, Z.-J. and Chung, Y.-C.: Dynamic Probabilistic Packet Marking for Efficient IP Traceback, *The International Journal of Computer and Telecommunications Networking*, Vol.51, No.3 (2007).

[15] Tian, H., Bi, J., Jiang, X. and Zhang, W.: A Probabilistic Marking Scheme for Fast Traceback, *Proc. 2010 2nd International Conference on Evolving Internet* (2010).

[16] Yen, W. and Sung, J.-S.: Dynamic Probabilistic Packet Marking with Partial Non-Preemption, *Proc. 5th International Conference on Ubiquitous Intelligence and Computing* (*UIC '08*) (2008).

[17] Lu, L., Chan, M.-C. and Chang, E.-C.: A General Model of Probabilistic Packet Marking for IP Traceback, *Proc. 2008 ACM Symposium on Information, Computer and Communications Security* (*ASIACCS '08*) (2008).

[18] Yan, Q., He, X. and Ning, T.: An Improved Dynamic Probabilistic

Packet Marking for IP Traceback, *Proc. I.J.Computer Network and Information Security* (2010).

[19] Belenky, A. and Ansari, N.: On deterministic packet marking, *Journal Compute Networks: The International Journal of Computer and Telecommunications Networking Archive*, Vol.51, No.10 (July 2007).

[20] Andrew, L.L.H., Hanly, S.V., Chan, S. and Cui, T.: Adaptive Deterministic Packet Marking, *IEEE Communications Letters*, Vol.10, No.11, pp.790–792 (2006).

Masayuki Okada works in the Engineering Department at JPNIC. He experienced a BGP operation of an academic network since 2000. Mr. Okada joined JPNIC in 2004 and is responsible for the development and operation of the IP resource management system related to routing and JPIRR research, as well as the use of IRR. In recent years, he has focused his efforts on outreach about RPKI technology and its operational deployment. He finished his Ph.D. in 2012 in Computer Science.

Nasato Goto is a master student at University of Tsukuba. He received his B.E. degree from University of Tsukuba in 2013. His research interests is network security.

Akira Kanaoka received his Ph.D. degree in engineering from University of Tsukuba, Japan in 2004. He worked at SECOM Co., Ltd. from 2004 to 2007, and at University of Tsukuba from 2007 to 2013. He is currently an assistant professor of Department of Information Science, Faculty of Science, Toho University. His research interests include network security and cryptographic application.

Eiji Okamoto received his B.S., M.S. and Ph.D. degrees in electronics engineering from Tokyo Institute of Technology in 1973, 1975, 1978, respectively. He worked and studied communication theory and cryptography for NEC central research laboratories since 1978. In 1991 he became a professor at Japan Advanced Institute of Science and Technology, then at Toho University. Now he is a professor at Faculty of Engineering, Information and Systems, University of Tsukuba. His research interests are cryptography and information security. He is a coeditor-in-chief of International Journal of Information Security and a member of IEEE and ACM.