An Intelligent Approach of Packet Marking at Edge Router for IP Traceback*

Dae Sun Kim[1], Choong Seon Hong[2,**], and Yu Xiang[3]

School of Electronics and Information, Kyung Hee Univerity, 449-701, Korea
{dskim,cshong}@khu.ac.kr, skipperyu@hotmail.com

Abstract. With the help of real source identity in packets, network security system can intelligently protect and counteract the attacks. Packet marking is an important method of source identification, and there are some issues on it. For large amount of packets, analysis time and complicated computation are necessary while detect marking information. This paper focuses on this direction, and proposes a simple and efficient method to mark all packets belonging to upstream traffic with a deterministic, plain form identity. With this approach, we just need low processing power on some specific edge routers as well as a little extra network traffic to settle it. Furthermore, distilling mark from packets is easy since the mark is in plain text format.

1 Introduction

Attackers routinely disguised their location using incorrectly, or spoofed IP source address. A great amount of effort has been made upon traceback to get the source information. Ideally, full-path traceback is a good way. But according to [1], packets may be delivered along different path to the destination (load balancing or unwanted isolation of the network routing) sometimes, only the ingress interface on the router closest to the source is almost same. The authors of [5] divides traceback technologies into two categories according to tracing clues: Traceback across stepping-stones and IP traceback. The first type is mainly for connection trace. For against DoS or DDoS, IP traceback is useful. It focuses on packet trace: Logging, ICMP Trace, probabilistic packet marking (PPM), Algebraic approach, Tunnel technologies, etc. Actually there are still some limitations in these approaches: large amounts of packets and complicated computation are necessary. Convergence procedure is relatively slow and background noise and spoofing marked packets may affect their performance.

2 Related Work

Source Path Isolation Engine (SPIE) [8,9] uses hash-based technique to gain every packet's information for IP traceback that generates and stores audit trails

* This work was supported by University ITRC Project of MIC

** Dr. C.S. Hong is the corresponding author

R. Khosla et al. (Eds.): KES 2005, LNAI 3683, pp. 303–309, 2005.

for inquery, and can trace the origin of a single IP packet delivered by the network in the recent past. About 0.5 percent of the link capacity per unit time in storage is needed. For realization all routers need to be controlled by specific manager. Besides of the corporation among all ISPs, wide deployment in whole Internet also is a big challenge. In Pi [6] is also a per-packet deterministic mechanism that allows the victim side to filter out packets matching the attacker's identifier embedded in each packet. It uses the digest of path and routers IP address as the identity. Each packet traveling along the same path carries the same identifier. It uses 16 bits of Identity field of IP header to store mark. Since the space is not enough for mark, tradeoff has to be adopted upon some hands, such as efficency and reliability. Enough quantity routers along all possilbe paths are needed. The authors had given a experience to show that the scheme is available in case of half of all related routers along attacking path. IP Traceback-based Intelligent Packet Filtering [7] is an integrated infrasture assembled with some approaches. And it is based on focused on filterring out the majority of DDoS traffic to improve the overall throughput of the legitimate traffic. With the help of PPM, the victim side can find the attacking paths and then filters out these "infected" path traffic in some degree. But this scheme needs rigorous conditions. Its EPM function modle must be deployed almost on all source side routers as well as all routers on the defence line for victim network are with PPF function. This prefigures the cost is very high. Deterministic Packet Marking (DPM) [1] is a novel packet-marking algorithm with all packets marked at edge routers. Like some approachs, the 16-bit ID field and the reserved 1-bit Flag in the IP header are used to store mark. The biggest difference is that incoming interface's IP address (32 bits) will be stored in two packets. The coding in the ID Field assumes that there are almost no IP fragments in the Internet supported by empirical traffic analysis less than 0.5 percent of all packets in Internet are fragmented in [4]. But SPIE sends mark by extra network bandwidth overhead for every traffic packet. Pi marks every packet with path information for reconstructing attack path map without extra network overhead. [7] inserts path identity into packet as mark in probabilistic. DPM marks every packet at source side with plain text of IP address. We found that the scanty space for storing mark in packet is a key reason among almost all approaches. Our scheme is inspired from the above works and is absorbed in finding more available space for storing mark.

3 Overview of Proposed Packet Marking Scheme

After the literature survey of popular IP traceback techniques, we claim that complex and more restriction traceback will suffer from scalability and deployment problem. An optimal and simple scheme should be introduced with lower processing and network traffic overhead. Our proposal draws inspirations from DPM scheme and also marks all upstream packets at one of source side routers belonging to edge router of subnet or domain as shown in Figure 1. If we allocate the subnet a global identity (GID) number as mark directly and store the mark in all output traffic packets, then destination receiving these packets can detect the source easily. However, storing a global identity number need more

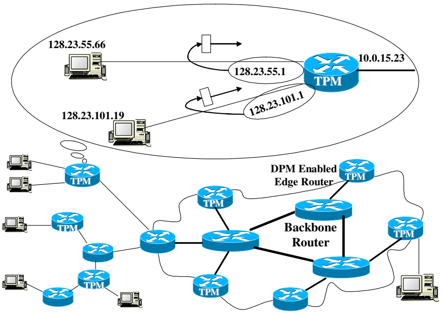
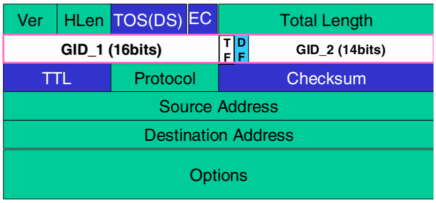


Fig. 1. TPM marks subnet identification at incoming interface of router which is connected to gateway of this subnet into all upstream traffic packets

space than 16 bits, it seem impossible that satisfy this asking from IP header fixed part directly. This is the key reason why most of current approaches use compression pattern for mark. In general viewpoint, there is no more space in IP header enough to store a global identity. Fragmentation related fields may be the last hope. Identity field is just only used for the fragmented packet, and Offset field is the same here with a little difference. If it is certain that a packet is not a fragment or DF is set, this field can be definitely used. In this case we can put the global identity of subnet in 30 bits space as in Figure 2. In general direct uses of IP address as global identity is the best and ideal choice but as shown in figure 2, there are only 31 bits that we can use. We can ignore the last one bit even more bits because IP addresses have only difference in last a few bits must locate in the same subnet or domain. The underlying reason we can perform the ignore action is that all we need to distinguish subnets instead of hosts. Total 30 bits resource is big enough to identify all subnets in Internet. Final receiver should know that if the packet is not fragmented. Thus we keep the DF (do not fragmentation) as it old value. The one bit at left side of DF is used for marking flag. For fragmented packet, instead, Option field is used. In this scheme we define a new sub-section in Option field with a new flag for IP traceback as shown in figure 3. Total overhead for the new sub-section is six bytes. The marks for both un-fragmented case and fragmented case are in plain



Ver	HLen	TOS(DS)	EC	Total Length		
GID_1 (16bits)				T F	D F	GID_2 (14bits)
TTL	Protocol	Checksum				
Source Address						
Destination Address						
Options						

Fig. 2. Storing Mark of Un-fragmented Packet in IP header fixed part

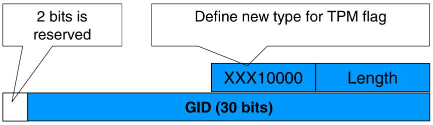
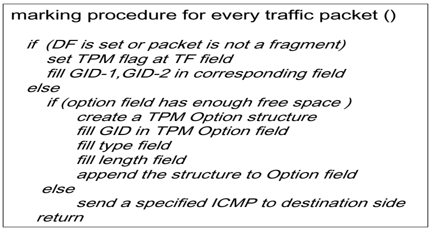


Fig. 3. Fragmented Packet Mark in Option Field

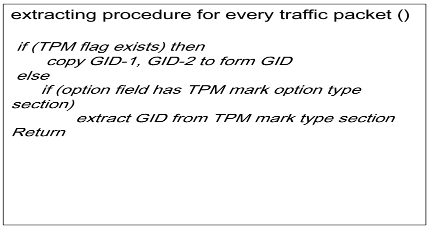
text form. Very easily works are needed both on marking at source side and extracting mark at destination side. The mark function is very easy to implement on routers. One of the possible algorithms of marking and extracting function looks as figure 4 and figure5.



```
marking procedure for every traffic packet ()

  if  (DF is set or packet is not a fragment)
      set TPM flag at TF field
      fill GID-1,GID-2 in corresponding field
  else
      if (option field has enough free space )
            create a TPM Option structure
            fill GID in TPM Option field
            fill type field
            fill length field
            append the structure to Option field
     else
            send a specified ICMP to destination side
    return
```

Fig. 4. Make Inserting Algorithm



```
extracting procedure for every traffic packet ()

 if (TPM flag exists) then
       copy GID-1, GID-2 to form GID
 else
      if (option field has TPM mark option type
section)
            extract GID from TPM mark type section
Return
```

Fig. 5. Make Extracting Algorithm

4 Discussion and Analysis

As well-known, the usage of fragmentation is decided by MTU size [3]. Fragmentation will degrade the efficiency and performance of Internet. In order to improve the performance of whole network, RFC1191 specifies a path MTU discovery protocol. At present this protocol is widely using in Internet, so majority of traffic do not experience fragmentation and normally DF is set. But there still has exception. A few IP protocol stack did not support this protocol. For instance, an un-fragmented packet (its MTU size is X, also it was not compliant PMTU) travels through the edge ingress router into backbone network and then pass through the downstream router. At this time if a more low speed data link media exists at the side of destination and its MTU is smaller than X, then fragmentation appears. It will destroy mark even the traffic has not been disrupted. Since the protected servers are valuable and important servers which access pattern is impossible lower than Ethernet link, thus the paths from it to the edge routers are not low bandwidth transmission media predicatively. According to the traffic analysis of Internet, nearly all packets size is not bigger 1500 bytes. In a word, in the paths from protected servers to marking point, the fragmentation case does not exist even in very special situation. As shown in figure 1, in TPM scheme it is unnecessary to add TPM function to all source side routers. Instead, just ensures all traffic packets to be marked one times and just one times. At the same time, instead of marking at edge router of backbone network, chooses TPM marking point close to hosts as possible. This rule is easy to be satisfied obviously. So that the granularity of source information can support the traffic flow restriction while against DoS or DDoS attacks.

5 Simulation Results and Evaluation

In this section we will demonstrate the simulation of our scheme and the comparison with DPM scheme. Based on OPNET 8.0.c remote server version under

MS Windows2000 professional version and Microsoft Visual C++ 6.0 compiling environment, The first simulation tests the impact on introducing the TPM function to existing routers. In our simulation, Database Access, email, http, ftp traffic are mixed and router's CPU background utilization is set to 30 percent for simulating the Internet traffic as possible. As figure 6 shown in TPM case is a very little higher than result points of without TPM case in Y-axis. Although smaller scale has been used in Y-axis, the difference between the two cases is too small to distinguish them easily. This means that the CPU utilization of edge router with TPM function embedded is very little higher than the CPU utilization of edge router without TPM function. The result in figure 7 shows nearly the same as figure 6. In second phase we compare our scheme with DPM scheme. Since DPM scheme must consider attacks might alternate the source IP address, we include this factor by adding a little part (0.8 percent)of source IP address changed. From figure 8 shown, TPM approach gains the average detection speed around 13 packets per seconds. We use two independent subnets as traffic source. Each subnet continually generates specific packets in uniform distribution pattern with interval time 0.1 to 0.2. That is to say TPM approach has almost detected all marks of received packets. On the contrary, the result line of DPM approach is around 6 marks per seconds in mark detection rate.

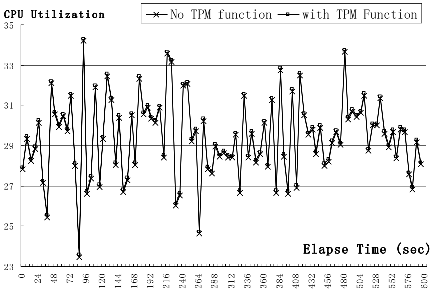


Fig. 6. CPU Utilization Comparison

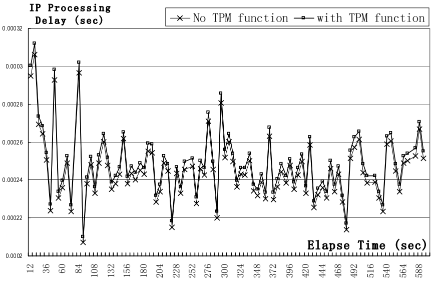


Fig. 7. IP Processing Delay Comparisons

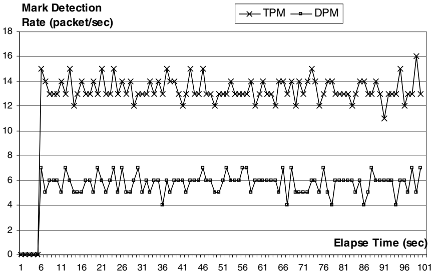


Fig. 8. Mark Detection Speed Comparisons

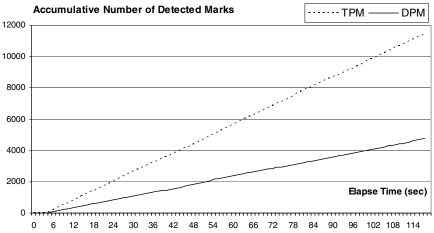


Fig. 9. Accumulative Number of Detected Packets

This is lower than half of TPM mark detection rate. There are several reasons for this difference. At first, DPM mark detection rate at most equal the half level as TPM approach even under ideal pre-arranged packets arriving sequence. Because DPM approach at least need two different mark parts to assemble a complete mark. Next is that the packets arrived in random. If the two or more packets carry with the same half of a complete packet continually, the detection rate must be decreased certainly. Another reason is that some packets with frequent change source IP address do not contribute mark detection as we discussed in previous part of this section.

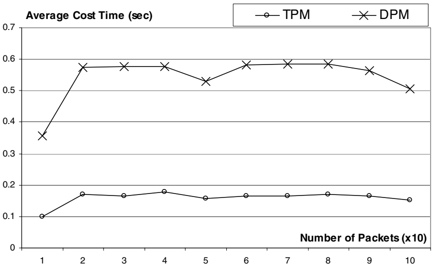


Fig. 10. Average Time Cost for Extracting Mark vs. Arriving Packets Speed around 5 to 10 pps

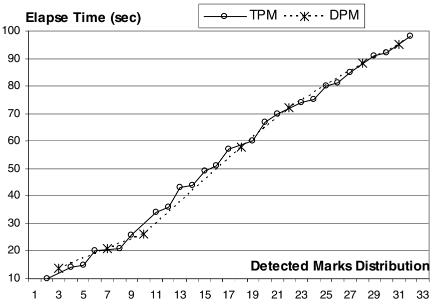


Fig. 11. Time Distribution of Detected Marks Comparison vs. Arriving Packets Speed around 0.5 to 1 pps

Therefore in figure 8 parts of marked packets could not contribute to assemble right mark. Furthermore instable source IP address will embarrass the mark's availability. It leads invalid mark packets even more than mark detected rate. figure 9 is about shows intuitionistic comparison between DPM scheme and our scheme upon number of detection rate by simulation procedure. Figure 10 is about the spending or cost of time for gaining the right mark. The last simulation item intends to compare the number of extracted marks and extracted time. The arriving packets number is reduced in order to convenient observing. Look at the result in figure 11, Some packets with mark were not detected in DPM scheme. Also the number of detected mark is smaller than half of TPM scheme detection. This confirms that during the same time TPM scheme distilled more marks than DPM scheme obviously.

6 Conclusion and Future works

In this paper we have introduced a new approach to IP traceback. This approach effectively addresses shortcoming of existing techniques in some degree. The important contributions of TPM lie in plain text form and a complete mark existing in every packet. It is not necessary computation to generate a different mark in real time for each wanted marking packet as current most approaches. This is big difference comparing with many packet marking approaches which generate digest information as mark using Message Authentication Code algorithm before each marking action. Marking all traffic packets can obtain the fastest speed of detection source information at destination sides. This approach is also efficient against various types of attacks, not only for DoS or DDoS attacks since all packets have trustworthy source side information, furthermore mark in plain text can be directly read and processed by security systems easily. We plan to do more deep investigation and experiments related to Internet traffic analysis, such as MTU size distribution, Option field using situation in Internet roundly to gain more supporting data for TPM scheme.

References

1. A. Belenky and N. Ansari, "IP Traceback with Deterministic Packet Marking," IEEE Communications Letters, Vol. 7, No. 4, pp. 162-164, April 2003
2. S. Savage, D. Wetherall, A. Karlin, and T. Anderson. "Network support for IP traceback" IEEE/ACM Trans. Networking, vol. 9, pp. 226-237, June 2001.
3. Behrouz A.Forouzan, Sohia Chung Fegan, "TCP/IP Protocol Suite" Chapter 7, pp 152-153
4. Stefan Savage, David Wetherall, Anna Karlin and Tom Anderson, "Practical Network Support for IP Traceback", Proc. of the ACM SIGCOMM conference, August 2000, Stockholm, Sweden, Computer Communication Review Vol. 30, No 4, October2000
5. "A Little Background on Trace Back".
   URL:discovery.csc.ncsu.edu/ pning/Courses/ csc774/on-trace-back.pdf
6. Abraham Yaar, Adrian Perrig and Dawn Song, "Pi: A Path Identification Mechanism to Defend against DDoS Attacks", In Proceedings of the 2003 Security and Privacy Symposium, May. 2003
7. Sung, M. and Xu, J. "IP Traceback-based Intelligent Packet Filtering: A Novel Technique for Defending Against Internet DDoS Attacks", IEEE Transactions on Parallel and Distributed Systems, vol 14, no 9, pp. 861–872, September 2003
8. Alex C. Snoeren et al., "Hash-Based IP Traceback", Proc. of the ACM SIGCOMM conference 2001, San Diego, CA, Computer Communication Review Vol. 31, No 4, October 2001, pp. 3-14
9. A. Snoeren, C. Partridge, L. Sanchez, C. Jones, F. Tchakountio, B. Schwartz, S. Kent, and W. Strayer. Single-Packet IP Traceback. In ACM/IEEE Transactions on Networking, vol. 10, no. 6, December 2002.