I.J.Computer Network and Information Security, **2010, 2, 47-53**
Published Online December 2010 in MECS (http://www.mecs-press.org/)

An Improved Dynamic Probabilistic Packet Marking for IP Traceback

Qiao Yan
Department of Computer and Software Shenzhen University, Shenzhen, China
yanq@szu.edu.cn

Xiaoming He and Tuwen Ning
Department of Information Engineering Shenzhen University, Shenzhen, China
{hfreeman2008}@126.com

***Abstract*—An improved dynamic probabilistic packet marking algorithm named IDPPM is presented, which not only can locate and attack a source rapidly and accurately, but also can reduce the marking overhead of routers near the attackers, which is its greatest contribution given by our technique. In contrast to previous work, the challenge of weakest node and weakest link is solved with the price of a little more numbers of packets to reconstruct the attack path. Theoretical analysis and NS2 simulation results in IPv4 and IPv6 testify that the approach is feasible and efficient respectively.**

***Index Terms*—Distributed Denial of Service (DDoS), IP traceback, Dynamic Probabilistic Packet Marking (DPPM), IPv6**

I. Introduction

Denial-of-service (DoS) attacks pose an increasing threat to a network systems in recent years as they are simple to implement, hard to prevent, and difficult to trace. In particular, Distributed-denial-of-service attacks (DDoS) become a major threat for the Internet because cohorts of malicious or compromised hosts coordinate to send a large volume of aggregate traffic to a victim. And the attackers often use spoofed IP address to disguise the true IP address because of the flaws of IP protocol. So IP source tracing on network is one of the most pressing tasks for network security researchers.

IP source tracing on network is a skill purposed to locate the attack source and specify the transmission path of the attack packets. The stateless nature of the Internet makes it very difficult to confirm the origin of IP packet. The reason is that the IP protocol provides no real means of authentication for packet origins and essentially operates entirely on trust when dealing with inter network traffic[1].But a variety of IP traceback techniques have been proposed and assessed for source tracing. The idea of encoding the address of the routers into attacking packets was first presented by Burch and Cheswick [2]. Savage, et al [3] proposed the famous traceback scheme, probabilistic packet marking (PPM), for practical IP traceback. PPM is a technique by marking individual packets with portion of the attack path with a constant marking probability (3%), based on the assumption that attackers send numerous packets. Song and Perrig have an advanced marking scheme (AMS) that copes with multiple attackers IP traceback problem by assuming some knowledge of Internet topology [4]. Because the victim possesses the map of its upstream routers, it becomes unnecessary to fragment edges. So the path Reconstruction speed is greatly improved and the false positive is much lower. Furthermore, the technique features low router overhead, supports incremental deployment, and provides efficient authentication of routers' markings. Tao Peng , et al [5] proposed an APPM scheme for routers to mark packets with adjusted probabilistic based on its position in the attack path. By implementing this scheme, the number of packets needed to reconstruct the attack path is substantially reduced compared with the optimal uniform marking probability in PPM. In 2006, a subtle approach, called dynamic probabilistic packet marking (DPPM), was presented to further improve effectiveness of PPM [6]. Instead of using a fixed marking probability, DPPM deduces the traveling distance of a packet and then choose a proper marking probability. And DPPM may completely remove uncertainty and enable victims to precisely traceback the attacking origin even under spoofed marking DoS attacks. Formal analysis indicates that DPPM outperforms PPM in most aspects. In 2009,the work of Feng bo, et al present a new packet marking algorithm to improve the effectiveness of PPM by using dynamic probability and fragment-reassembly[7], which significantly solves the problems of the lost of marking information and the difficulties to reconstruct the attack path. And Gao dapeng, et al proposed a new approach of composed packet marking method [8]. Compare with the DPPM algorithm, the marking probability of border router decreases from 1 to 0.5 in this new proposal. But the problem of excessive burden on the router near the attackers is not solved completely. In this paper, an improved dynamic probabilistic packet marking algorithm named IDPPM is presented to solve this problem.

The rest of this paper is organized as follows. We present a brief description about dynamic probabilistic packet marking in Section II. Section III introduces our improved dynamic probabilistic packet marking (IDPPM)

algorithm and shows the theoretical analysis about its performance. The concrete strategy of IDPPM algorithm applied in IPv4 and simulation results that demonstrate the effectiveness of our improved algorithm are provided in Section Ⅳ .In section Ⅴ ,we implement IDPPM algorithm scheme in the next generation Internet Protocol ,IPv6 .And the simulation results verify the applicability and efficiency of this approach .Finally we concluded in Section Ⅵ.

II. Dynamic Probabilistic Packet Marking

We assume that the attack path $\psi = a, r_1, r_2, ..., r_D, v$ is comprised of D routers, where a and v denote the attacker and the victim of a DoS occurrence, and r_i (i=1, 2,…, D) indicate D routers in the attack path. Let p_i represent the marking probability of router r_i. Define the leftover probability for router r_i, denoted by α_i, to be the probability that an attacking packet has lastly been marked at router r_i and nowhere further down the path [7].For victim v, α_i is the probability that allows v to learn that router r_i is on the attack path by examining this arriving packet [7].

It can be seen that:

$$\alpha_i = \begin{cases} p_i * \prod_{j=i+1}^{D} (1-p_j) & for \quad 1 \le i < D \\ p_D & for \quad i = D \end{cases} \qquad (1)$$

For a given attack path, let i($1 \leqslant i \leqslant D$) be the traveling distance of a packet from its source. According to dynamic probabilistic packet marking (DPPM), the marking probability of router r_i is chosen $p_i = 1/i$ to mark packet. And the value of traveling distance of a packet from its source can be deduced from Time-to-live (TTL) value in the IP header. The marking probability is the only difference between DPPM and PPM. But formal analysis indicates that DPPM outperforms PPM in most aspects.

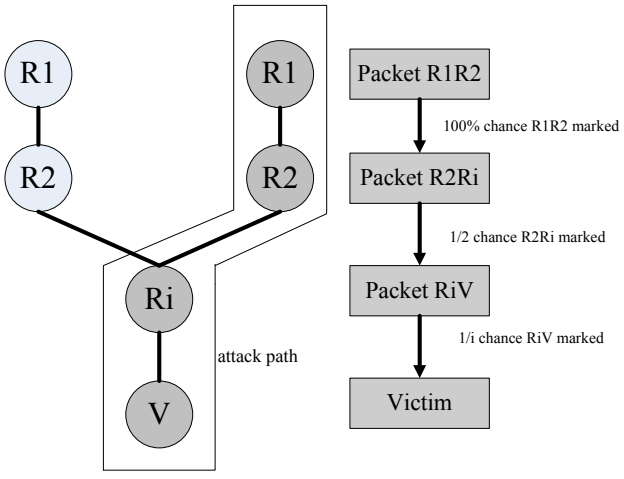

Figure 1. Dynamic probabilistic packet marking.

It can be shown that

$$\alpha_i = \frac{1}{D} \qquad for \quad 1 \le i \le D \qquad (2)$$

So each router along the attack path has the same leftover probability. By the theory of Coupon collector [9], we know that the victim needs the minimal number of packets to reconstruct the attack path successfully. In addition , the uncertainty introduced by spoofed marking may be removed completely because every packet is marked at least once along the attack path in IDPPM .Fig. 1 shows a basic overview of the DPPM procedure, with the marking probability $p_i = 1/i$ to mark packets for router r_i .

III. An Improved Dynamic Probabilistic Packet Marking Algorithm

A. The Basic Idea Of The Improved Dynamic Probabilistic Packet Marking

When the value of i is smaller, which means that the router is closer to the attacker, the marking probability of the router is greater. Especially, as the i is 1,2,3, the marking probability of r_1 , r_2 , r_3 is up to 1,1/2,1/3 respectively, which will result in an excessive burden on the router, and even the service is paralyzed. The above-mentioned is the biggest drawback of DPPM [7]. And from a purely sampling point-of-view, edge (a, r_1) is the "weakest link" and node a is the "weakest node" requiring the most samples for path reconstruction because the packet's marking information will be overwritten[10].

In this paper, we present a technique which make an improvement on DPPM by using 2bits field (F0F1) in packet header to solve these problems, which is named IDPPM. The following is basic idea of IDPPM. We initialize the value of F0F1 to (00). A router checks the value of i .If the value of i is equal to (1,2,3),it denotes that the distance between the router and the attacker is 1,2,3 respectively, the router will mark the packet with probability p1,p2,p3,and set the value of F0F1 to (01,10,11) separately. When a router marks packets, it must first check the value of F0F1.If F0F1= (00), the router marks the packet with the DPPM algorithm. If F0F1= (01, 10, 11), this means that this packet is marked before, the router does not mark this packet in order to avoid overwritten.

We choose the value of p1, p2, p3 to1/D. This can not only greatly reduce the marking probability of routers which are near the attackers, but also ensure that the leftover probability of the router (r_1 , r_2 , r_3) are the same as DPPM. Furthermore, IDPPM can resolve the "weakest link" and "weakest node" puzzle by using 2bits field to avoid overwritten during the attacker path reconstruction.

B. The Algorithm of IDPPM

```
Marking procedure at router R:
  let R'=BitIntereave(R,Hash(R))
  let k be the number of non-overlapping fragments R'
    for each packet w
      let x be a random number from[0,…,1)
      if F0F1==(00) then
        let o be a random integer from [0,…,k-1]
        let f be the fragment of R' at offset o
        if 3<i and x<1/i then
           Mark_packet()
        else if i==1 and x<1/D then
           write 01 into w.F0F1
           Mark_packet()
        else if i==2 and x<1/D then
           write 10 into w.F0F1
           Mark_packet()
        else if i==3 and x<1/D then
           write 11 into w.F0F1
           Mark_packet()
    else
        if w.distance=0 then
        let f be the fragment of R' at offset w.offset
            write f ⊕ w.frag into w.frag
           increment w.distance

Mark_packet():
  write 0 into w.distance
  write o into w.offset
  write f into w.frag

Path Reconstruction procedure at victim v
    let FragTbl be a table of tuples(frag,offset,distance )
    let G be a tree with root v
    let edges in G be tuples (start,end,distance )
    let maxd=0,last=0
    for each packet w from attacks
        FragTbl.Insert(w.frag,w.offset,w.distance)
        if w.distance>maxd then
            maxd=w.distance
    for d=0 to maxd
          for all ordered combinations of fragments at
          distance d construct edge z
              if d ≠0 then
                    z=z ⊕ last
              if Hash(EvenBits(z))=OddBits(z)then
                    insert edge(z,EvenBits(z),d) into G
                    last= EvenBits(z)
remove edge(x,y,d) with d≠distance from x to v in G
extract path(Ri…Rj)by enumerating acyclic paths in G
```

C. Performance Analysis

a) Overhead on Routers

Each marking poses some cost to a router. We now proceed to compare the overhead of DPPM and IDPPM. For simplicity, we use number of markings performed as our measurement for overhead on router [7]. Let us consider a DoS attack with N packets sent from α to ν. And let o_{dppm} and o_{idppm} denote individual overhead of

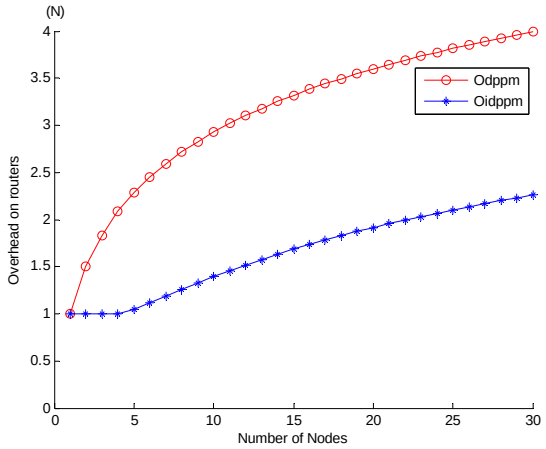

Figure 2. A comparison of total overhead by DPPM and IDPPM.

DPPM and IDPPM in a route along the attack path, respectively. Let O_{dppm} and O_{idppm} denote the total overhead summed over all D routers of DPPM and IDPPM, respectively.

$$o_{dppm} = N / i \tag{3}$$

$$o_{idppm} = \begin{Bmatrix} N * \frac{1}{D} & i \le 3 \\ N / i & i > 3 \end{Bmatrix} \tag{4}$$

$$O_{dppm} = N\left(\frac{1}{1}+\frac{1}{2}+\frac{1}{3}+\frac{1}{4}+...+\frac{1}{D}\right) \tag{5}$$

$$O_{idppm} = \begin{cases} N * \frac{i}{D} & i \le 3 \\ N\left(\frac{3}{D}+\frac{1}{4}+...+\frac{1}{i}\right) & i > 3 \end{cases} \tag{6}$$

Fig. 2 compares the total overhead of DPPM and IDPPM with different D. It is clear that O_{idppm} is less than O_{dppm} significantly on all routers. This means that the routers under IDPPM suffer a low total overhead. This is mainly due to the fact that IDPPM algorithm reduces the marking probability of routers (r_1 , r_2 , r_3) greatly by using the F0F1 field. So $o_{idppm} < o_{dppm}$, $O_{idppm} < O_{dppm}$.

b) False Positive

We would like to note that a path reconstruction mechanism will suffer from false positives. The main reason is that it is difficult to prove whether the path is reconstructed completely or partly [11].

As the IDPPM algorithm marks the edge router of r_1 with the value of F0F1 (01). If F0F1=01, it denotes that this packet is marked by the edge router (r_1), and the

TABLE I.
COMPARISON OF $E(N_{dppm})$ AND $E(N_{idppm})$

	D					
	5	10	15	20	25	30
$E(N_{dppm})$	9	24	41	60	81	103
$E(N_{idppm})$	16	32	50	70	91	113
$D^3/(D-1)^3$	1.95	1.37	1.23	1.17	1.13	1.11

path is reconstructed completely. So this can reduce the false positives obviously.

c) Expected Value Of Minimal Of Packets For Reconstruction

To satisfy the requirement of at-least-one-marking per router, a victim needs to collect a certain number of packets [7]. The expected value of minimal number of packets required for a successful traceback by both DPPM and IDPPM, denoted by $E(N_{dppm})$ and $E(N_{idppm})$, respectively, depends on the leftover probability. We learned from in (2) that the leftover probability of all routers on the attack path is 1/D.Therefore, we conclude that

$$E(N_{dppm}) = D * \ln D \tag{7}$$

For IDPPM algorithm,

$$\alpha_i = \begin{cases} \frac{1}{D} & i = 1,2,3 \\ \frac{1}{D} * (\frac{D-1}{D})^3 & i > 3 \end{cases} \tag{8}$$

Therefore, we can obtain the value of $E(N_{idppm})$:

$$E(N_{idppm}) = \begin{cases} D * \ln D & i = 1,2,3 \\ D * (\frac{D}{D-1})^3 * \ln D & i > 3 \end{cases} \tag{9}$$

It can be seen that IDPPM needs a little more numbers of packets for a successful traceback than DPPM from (7) and (9).Table 1 displays some numerical values of $E(N_{dppm})$ and $E(N_{idppm})$ .The table 1 clearly shows that the difference between DPPM and IDPPM decreases gradually with the value of D increasing. $E(N_{idppm})$ is 1.95 times as much as $E(N_{dppm})$ when D=5,but it is acceptable for that the amount of $E(N_{idppm})$ and $E(N_{dppm})$ is extremely small. And $E(N_{idppm})$ is in close proximity to $E(N_{dppm})$ with D =25, 30.

Ⅳ. IDPPM ALGORITHM IMPLEMENTATION IN IPV4

A. Marking Field Selection And Encoding Issues

ver	hlen	TOS	total length	
identification			flags	offset
TTL		protocol	header checksum	
source IP address				
destination IP address				

Figure 3. The IPv4 header(darkened areas represent underutilized bits).

offset	distance	edge fragment	F0F1
3-bit	5-bit	8-bit	2-bit

Figure 4. The marking field encoding format in IPv4.

According to [12], since less than 0.25% of all Internet data packets will use the “identification" (16-bit), we think that the path information is overloaded into this field is appropriate. The TOS field is an 8bits field in the IP header. And the field has been little used in the past. Reference [13] shows that setting this field arbitrarily makes no measurable difference in packet delivery. As shown in Fig. 3, we choose to use ID field (16-bit) and 2 bits out of the TOS field as marking field for IDPPM algorithm.

There are just only 18bits field available for use in each packet. So we use the Compressed Edge Fragment Sampling scheme to encode the edge fragments into the IP marking Field. The marking field encoding format is shown in Fig. 4:

F0F1: value set is (00, 01, 10, 11), mainly used to mark the router r_1 , r_2 , r_3 .

B. Simulation

Our simulations were run under Windows XP on a 2.7 GHz Pentium 4 with 768 MB of RAM.To test the performance of the IDPPM algorithm, we choose to use NS-2.33 to simulate.

And we need to expand the NS2 to evaluate the effectiveness of the PPM, DPPM and IDPPM algorithm. First, the offset (3-bit), the distance (5-bit) and the edge fragment (8-bit) are added into the IP header to be used as marking field. Second, we use the default address format (a 32-bit integer node-id) to identify the node itself. And the PPM, DPPM and IDPPM marking algorithm are injected into the “recv” function in the “trace.cc” file. Then the marking information of each packet is output into the trace file by modifying the “format” function in the “trace.cc” file so as to process all data at centralized locations. Calling the Tcl scripts generates the trace files. Finally, calling awk documents written with the attack path reconstruction algorithm processes the trace file to locate attack sources. The result of simulation is shown in Fig. 5.

It can be seen that the IDPPM, just as DPPM, requires obviously much less packets than PPM to reconstruct the attack path. Although the IDPPM algorithm needs a little more packets to traceback than DPPM algorithm, its individual overhead on the routers close to the attacker and the total overhead summed over all routers are less than DPPM algorithm dramatically. This means that the

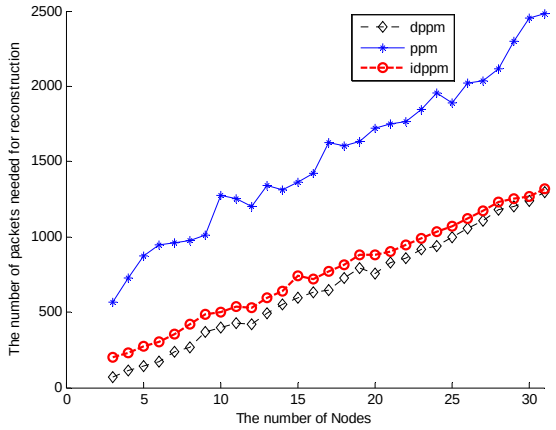

Figure 5. A comparison of numbers of packets required by PPM, DPPM and IDPPM in IPv4.

IDPPM algorithm features fewer packets to reconstruct the attack path compared with the PPM algorithm and lower overhead on router compared with the DPPM algorithm.

Ⅳ. IDPPM Algorithm Implementation In IPv6

A. *Ipv6 Header*

Implementation of IDPPM algorithm to IPv6 requires a thorough analysis of IPv6 header so we can efficiently mark malicious packets and reconstructs the attack path.IPv6 header is simpler but longer than IPv4 header as shown in Fig. 6.

It is clear that there is not an Identification field in IPv6 header, so it is necessary to find a field within the IPv6 header or extension header that can be used to put the marking information. According to [3], the Flow Label can be overloaded to implement the IDPPM algorithm, to avoid increasing packet size by overloading the extension header and to simplify processing for intermediary routers.

According to the IPv6 specifications [14], the Flow Label field in the IPv6 datagram header is a 20-bit field denoting a packet sequence flowing from one source address to a specific destination or destinations. Specific requirements for its usage are not yet finalized, but RFC 3697 has listed many of the details to its general functioning.

B. *Marking Field Encoding Issues*

As discussed above, implementation IDPPM algorithm in the context of IPv6 is based on overloading

ver	traffic class	flow label	
payload length		next header	hop limit
source IP address (128-bit)			
destination IP address (128-bit)			

Figure 6. Standard IPv6 header.

offset	distance	edge fragment	F0F1
5-bit	5-bit	8-bit	2-bit

Figure 7. The marking field encoding format in IPv6.

the Flow Label field. Compared with IPv4 addresses, IPv6 addresses are 128-bit instead of 32-bit.The total router address information are 256 bits, because the original IP address(128bits) is interleaved with its hash value(128bits)(the original address on odd bits, the hash value on even bits). But there are only 20 bits in Flow Label field, used to store the marking information. Just like the scheme used in IPv4, we chose to implement the IDPPM algorithm based on the Compressed Edge Fragment Sampling scheme in IPv6, to solve this challenge. In this scheme, each edge fragment is 8-bit too. However, each offset is 5-bit instead of 3-bit in IPv6. So the total router address information can be just expressed by combination of the edge fragment and offset ($8*2^5$=256).The encoding format of the Flow Label field is shown in Fig. 7.

distance: value set is (0~31),used to denote the distance from current router to victim. The network topology is about 20 hops, no more than 32 hops, so the distance with 5 bits is enough.

F0F1: value set is (00, 01, 10, 11), mainly used to mark the router r_1 , r_2 , r_3 .

edge fragment: value set is (edge fragment[0],……, edge fragment[7]).

offset: value set is (0~31).

C. *Simulation*

Just like implementation of IDPPM algorithm in IPv4, we choose to use NS-2.33 to verify the applicability and efficiency of this approach.

a) feasibility and effectiveness

The result of simulation is shown in Fig. 8. It is obvious that the IDPPM algorithm, just as DPPM, needs much less packets than PPM algorithm to reconstruct the attack path and locate the attacker. This means that, compared with PPM algorithm, one of the characteristic features of this technique is its quicker rate of converge.

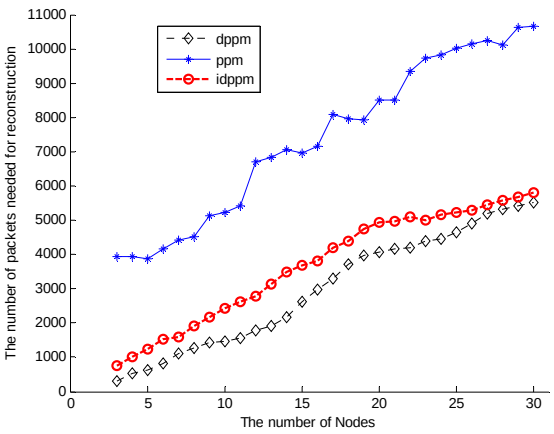

Figure 8. A comparison of numbers of packets required by PPM, DPPM and IDPPM in IPv6.

Although the IDPPM algorithm requires a little more packets to traceback the attacker than DPPM algorithm, it is acceptable for that the amount of difference is extremely little. Especially as the distance grows, the difference becomes less and less. This means that the IDPPM algorithm features fewer packets to reconstruct the attack path and locate the attacker compared with the PPM algorithm and lower overhead on routers and lower false positive compared with the DPPM algorithm, just like the simulation results in IPv4. In a word, the results demonstrate that the Flow Label can be overwritten to afford the IDPPM algorithm in IPv6.

b) challenge

Breaking IP address into small chunks and marking packets with the small portion of IP address leads to state explosion problem which is one of the major drawbacks of the Compressed Edge Fragment Sampling scheme[15].The router IP address(128-bit) in IPv6 is considerably much longer than that one(32-bit) in IPv4. But the number of bits of marking field in packet is not increased accordingly. This results in the number of offset increased rapidly in order to mark the router IP address into the marking field by Compressed Edge Fragment Sampling scheme. So state explosion problem is more serious in IPv6 . Fig. 9, Fig. 10 and Fig. 11 display a comparison of numbers of packets required by PPM, DPPM and IDPPM algorithm in IPv4 and IPv6 respectively. It is obvious that the amount of packets needed to reconstruct the attack path in IPv6 far more than that ones in IPv4.This is the state explosion problem performance. That said, the convergence of algorithm is needed to improve and the computational complexity to reconstruct the attack path is needed to reduce in IPv6.

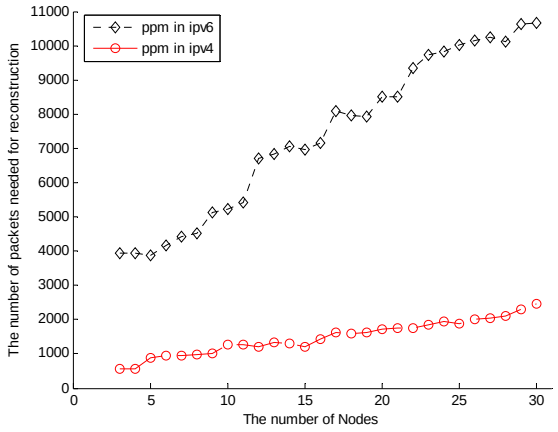

Figure 9. A comparison of numbers of packets required by PPM in IPv4 and IPv6.

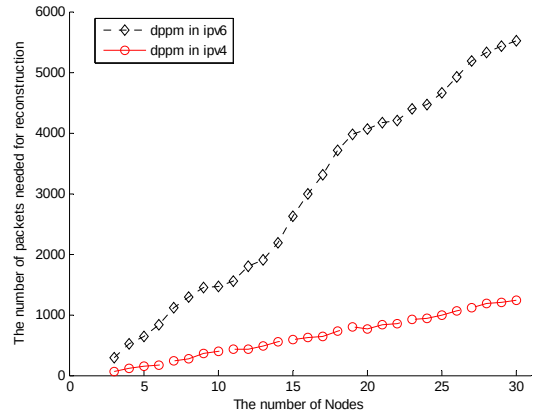

Figure 10. A comparison of numbers of packets required by DPPM in IPv4 and IPv6.

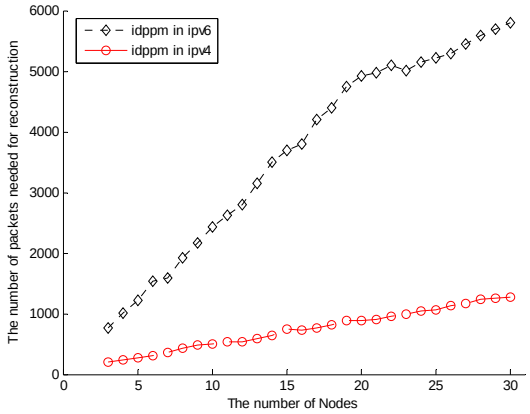

Figure 11. A comparison of numbers of packets required by IDPPM in IPv4 and IPv6.

Ⅴ. Conclusions And Future Work

In this paper, we introduce the IDPPM algorithm to locate the packet flooding the attacks source in the Internet. The IDPPM algorithm not only can locate and attack the source rapidly and accurately, but also can reduce the marking overhead of routers near the attackers, which is greatest contribution gave by our technique. In contrast to previous work, the challenge of weakest node and weakest link is solved with the price of a little more numbers of packets to reconstruct the attack path. The rate of false positive is reduced obviously with the value of F0F1 (01).The results of NS2 testify that the approach is feasible and efficient.

The further research direction is the implementation of IDPPM algorithm in IPv6 to avoid the state explosion problem. There are two reasons leading to the state explosion problem. First, the router IP address is 128-bit in IPv6 and the total router address information needed to mark is 256-bit.But there are not enough marking field in packet header used to store the marking information. The 2nd reason comes from the Compressed Edge Fragment Sampling scheme. This scheme can solve the difficult problem of encoding the total router address information into marking field of packet header, but it give rise to the high complexity and heavy loads on the victim to reconstruct the attack path in a certain extent. So the next step we would like to take is the improvement of the Compressed Edge Fragment Sampling scheme, in order to reduce complexity of the algorithm and computation overhead on the victim to reconstruct the attack path.

ACKNOWLEDGMENT

This paper is benefited greatly from the help of many different people-far more than can be listed completely here. Still, we would like to thank to LIU Ling for her suggestion to the simulation.

REFERENCES

[1] R. Morris, A weakness in the 4.2 BSD Unix TCP/IP Software, AT&T Bell Labs, Technical Report Computer Science 117, 1981.

[2] H. Burch, and B. Cheswick, "Tracing Anonymous Packets to Their Approximate source," the 14th USENIX conference on System administration, USENIX Association Press, pp. 319-328,Jul 2000

[3] S. Savage, D. Wetherall, and A. Karlin, "Network Support for IP Traceback," Proc. IEEE/ACM Transactions on Networking, IEEE Press, pp. 226-237, June 2001

[4] D. Song,and A. Perrig, "Advanced and Authenticated Marking Schemes for IP Traceback," Proc. the IEEE INFOCOM, IEEE Press, pp. 878-886,2001

[5] T. Peng, C. Leckie, and K. Ramamohanarao, "Adjusted Probabilistic Packet Marking for IP Traceback," Proc. the Second International IFIP-TC6 Networking Conference on Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; and Mobile and Wireless Communications, pp. 697-708, 2002

[6] L. Jenshiuh,L. Zhi-Jian,and C. Yeh-Ching, "Dynamic probabilistic packet marking for efficient IP traceback," Proc. the International Journal of Computer and Telecommunications Networking, Elsevier North-Holland Press, Feb 2007,pp. 866-882, doi: 10.1016

[7] F. Bo,G. Fan,and Y. Min, "Dynamic Probabilistic Packet Marking Based On PPM," Proc. WMWA 09. Second Pacific-Asia Conference, pp. 289-292 ,June 2009

[8] G. Dapeng,Y. Shicai,and Y. Wenzhi, "Research on Composed Packet Marking for IP Traceback Algorithm," Computer Engineering, Vol . 35, pp. 115-117, May 2009(In Chinese).

[9] A. Boneh,and M. Hofri, *The Coupon Collector Problem Revisited Commun[J]*, Static Stochastic Models,1997,pp. 39-66

[10] K. Park,and H. Lee, "On the effectiveness of Probabilistic Packet Marking for IP Traceback under denial of service attack,"Proc. IEEE INFOCOM 2001, IEEE Press ,pp. 338-347,2001

[11] V. Kuznetsov, A. Simkin,and H. Sandstom, "An evaluation of different IP traceback approaches," Proc. the 4th International Conference on Information and Communications Security, pp. 37-48,2002

[12] L. Stoica,and H. Zhang, "Providing Guaranteed Services Without Per Flow Management," Proc. the conference on Applications, technologies, architectures, and protocols for computer communication, ACM Press, pp. 81-94,1999

[13] D. Drew,F. Franklin, S. Adam, "An Algebraic Approach to IP Traceback," Proc. the ACM Transactions on Information and System Security, ACM Press ,pp. 119-137, May 2002

[14] J. Rajahalme,A. Conta,B.Carpenter,and S.Deering, *IPv6 Flow Label Specification*, RFC 3697,2004.

[15] M. Waldvogel, "GOSSIB vs. IP Traceback Rumors," Proc. the Computer Security Applications Conference, 2002

Qiao Yan received the Ph.D degrees in information and communication engineer from Xidian University, China, in 2003. Now she is a professor in Shenzhen University. Her research interests include network security.

Xiaoming He received the BA degree from Hei Longjiang institute of science and technology in 2008. He is currently completing the Master's of Information Process Technologies degree at Shenzhen University's Information Engine Institute. His area of research interest is in the area of IP traceback.

TuWen Ning received the BA degree from Shenzhen University in 2009. He is currently completing the Master's of Information Process Technologies degree at Shenzhen University's Information Engine Institute also. His area of research interest is in the area of network security.

