

## 基于路由器接口的一致概率包标记算法

闫巧, 姚希彦\*

(深圳大学 计算机与软件学院, 广东 深圳 518060)

(\* 通信作者电子邮箱 yaoyang861214@163.com)

**摘要:** 概率包标记 (PPM) 算法是防御分布式拒绝服务攻击 (DDoS) 的重要方法, 针对 PPM 因为重复标记而存在最弱链和弱收敛性问题, 以及因为分片问题而导致重构路径时计算量大等问题, 提出一种基于路由器接口 (ID number) 的一致概率包标记算法——IDCPPM, 该算法使每个路由器的标记信息都能以一致的概率到达受害者, 且由于不用分片, 因而有效地减少了重构路径时所需要接受包的数量, 降低了算法的复杂度, 并且新方案能扩展到 IPv6 中。理论分析和实验仿真证明了该方法的有效性。

**关键词:** IP 追踪; 概率包标记; 最弱链; 弱收敛性; 分布式拒绝服务

**中图分类号:** TP393.08 **文献标志码:** A

### Packet marking algorithm with consistency probability based on router interface

YAN Qiao, YAO Xi-yan\*

(College of Computer Science and Software Engineering, Shenzhen University, Shenzhen Guangdong 518060, China)

**Abstract:** Probabilistic Packet Marking (PPM) algorithm is an important method to prevent the Distributed Denial of Service (DDoS) attacks. But it has the weakest chain and the weak convergence of issues because of the repeated marking, as well as large amount of computation because of the fragmentation problem when reconstructing the path. A new marking algorithm — IDCPPM was proposed which was based on router interface (ID number) with a consistency probability. The algorithm enabled the marking information to reach the victims with a consistency probability. For its non-fragmentation, it effectively reduced the number of packets needed to reconstruct the path and reduce the complexity of the algorithm. Also it can be applied to IPv6. The theoretical analysis and experimental results prove the effectiveness of this method.

**Key words:** IP traceback; Probabilistic Packet Marking (PPM); weakest link; weak convergence; Distributed Denial of Service (DDoS)

近年来, 拒绝服务攻击 (Distributed Denial of Service, DDoS) 由于其易实施、难防范、难追踪等特点, 成为当今重要的安全问题之一。拒绝服务攻击事件持续上升, 给整个网络带来了极大的危害。对于如何防御 DDOS 攻击, 目前已有许多方案<sup>[1]</sup>: 包标记、日志记录等。其中包标记方案由于其灵活性高、网络中路由负担小等优点, 而成为众学者研究的热点, Savage 等<sup>[2]</sup>提出的概率包标记 (Probabilistic Packet Marking, PPM) 方法为后续研究奠定了基础。其后针对 PPM 的不足, 有一些其他改进方案<sup>[3-13]</sup>, 其中李德全等<sup>[3]</sup>提出的使用自适应概率、标记 12 位 number 的自适应包标记算法, 在收敛时间、误报率、复杂度方面的效果都比 PPM 好。本文在此基础上提出一种基于路由器接口的一致概率包标记方案, 能更加有效地减少重构路径所需的数据包的数量, 降低重构路径的复杂度, 最终提高重构路径的效率。

### 1 概率包标记与自适应的包标记算法

概率包标记包括标记算法和重构算法, 标记算法让网络中的路由器  $R$  以一定的概率标记所经过的数据包, 即在数据包中写入路由器  $R$  的 IP 信息。重构算法是让受害者  $V$  收集到一定的数据包后, 提取并组合包中所携带的途经路由器  $R$  的 IP 信息片段, 得到攻击路径信息。

PPM 的标记算法为: 将路由器的 32 位 IP 地址及其校验码按位交叉, 得到 64 位的信息。再将 64 位信息分成 8 块 (每块 8 位), 以 0~7 对它们按序标号, 依概率标记到数据包中。

在实现路径重构的时候, 用 distance 域来表示标记路由器到受害者  $V$  之间的距离, 一般来说, 网络中的数据包所经过的路径长度一般小于 25 跳, 所以标记距离的 distance 域用 5 位来表示。

文献[3]提出自适应的概率标记方案: 为整个网络上的每个路由器提供一个 12 位的随机 number, 将该 12 位的 number 分成四片 0, 1, 2, 3, 在标记域用  $3 \times 3$  位表示 frag0, frag1, frag2, 用 2 位 offset 域来表示偏移位。需要标记时, 使用自适应的概率<sup>[14-15]</sup>, 每个数据包标记相连的三片 offset,  $(\text{offset} + 1) \% 4$ ,  $(\text{offset} + 2) \% 4$  写入 frag0, frag1, frag2, 共 9 位, distance = 0 时, 将路由器的第  $(\text{offset} + 1) \% 4$  片写入 frag1 中, distance = 1 时, 将路由器的第  $(\text{offset} + 2) \% 4$  片写入 frag2 中。同样用 5 位 distance 域表示标记路由到  $V$  的距离。

自适应标记算法比 PPM 方法效果好一些, 但是仍然没有很好地解决 PPM 中所存在的一些问题:

假设一条攻击路径为  $P, P = \{A, R_0, R_1, R_2, \dots, R_n, V\}$ 。其中  $A$  表示攻击者,  $V$  表示受害者,  $R_0, R_1, R_2, \dots, R_n$  表示数据包所经过的一组路由器, 其标记概率为:  $\{P_0, P_1, P_2, \dots, P_n\}$ 。

1) 存在最弱链问题。在 PPM 中, 路由器  $R_i (i = 0, 1, 2, \dots, n)$  对每个数据包以固定概率  $p = 0.04$  标记, 每个数据包到达受害者的概率为  $P_i = p(1-p)^{d-i}$ , 则有  $P_0 < P_1 < P_2 < \dots < P_n$ 。因此, 可知离攻击者越近的路由的标记信息越难得到,  $\{A, R_0\}$  的信息最难收集到。这样就要受害者收集更多的包, 才能收集到足够重构最弱链的信息。文献[3]中使用自适应

收稿日期: 2012-04-27; 修回日期: 2012-06-13。 基金项目: 国家自然科学基金资助项目 (60972011)。

作者简介: 闫巧 (1972-), 女, 广西资源人, 教授, 博士, CCF 会员, 主要研究方向: 网络安全; 姚希彦 (1986-), 女, 河南鹤壁人, 硕士研究生, 主要研究方向: 网络安全。

标记概率,  $P_{-1} = 1, P_0 = 1/2, P_1 = 1/6, P_2 = 1/10, P_3 = P_4 = \dots = 0.04$ , 增大了  $\{A, R_0\}$  收集到的概率, 但是存在次弱链问题, 即  $P_3 < P_4 < P_5 < \dots < P_n$ , 是一个局部最弱链问题。因此说自适应算法没能很好地解决最弱链问题。

2) 需要收集较多的数据包, 收敛时间长。在 PPM 中, 路由器  $R_i (i = 0, 1, 2, \dots, n)$  对每个数据包以一定的概率标记, 每个路由 IP 都分成 8 个分段, 就需用 8 个数据包来标记, 由标记概率  $P_0 < P_1 < P_2 < \dots < P_n$  知, 离攻击者越近的路由的标记信息越难得到, 而要得到 8 个这样的不重复的数据包来组合一个 IP, 受害者收到的数据包的总量就会增加到 8 倍以上。这样就会导致收敛时间长。在文献[3] 中 ID number 分成 4 个分片, 是 PPM 的一半, 理论上收到的数据包总量也是 PPM 的一半, 收敛时间也为 PPM 的一半。

3) 要组合的 IP 分片多, 重构算法复杂。在 PPM 中重构攻击路径至少需要 8 个有效数据包才能组合出一个完整的边信息, 在  $m$  条攻击路径的时候, 要组合  $m^8$  次。在文献[3] 中至少需要 4 个有效的数据包, 需要组合  $m^4$  次, 复杂度降低了, 但是仍然不能有效地降低重构算法的复杂度。

4) 误报率高。在一般的包标记算法中, 误报发生的次数与攻击者的数量成正比。假设有  $m$  个攻击源,  $k$  为分片数, 算法的误报率为  $1 - (1 - 1/2^k)^n, n = m^k$ , PPM 算法<sup>[2]</sup> 的误报率为  $1 - (1 - 1/2^{32})^n, n = m^8$ , 自适应包标记算法<sup>[3]</sup> 的误报率为  $1 - (1 - 1/2^{36})^n, n = m^4$ , 实验结果证明自适应算法的误报率有了很好的改进。

## 2 基于路由器接口的一致概率包标记

### 2.1 改进的一致概率包标记思想

针对 PPM 中的最弱链和收敛时间长等问题, 提出一种基于路由器接口的一致概率包标记算法——IDCPPM。

在 PPM 中, 标记地址用的是 32 位的真实 IP 地址, 这样重构的时候就需要组合比较多的次数, 算法复杂度高, 误报率也高; 而实际上, 要区分网络上相邻的路由地址, 可以不需要一个 32 位的 IP 地址, 文献[3, 8-9] 都提出可以用一个 12 位的 number 来标识。一般来讲, 与某一路由相邻的路由器一般不超过 128 个, 文献[5] 中使用 7 位来标示路由器接口信息, 而新方案给每一个路由器的相邻路由器分配一个 8 位的 ID number, 并且以相等的概率从  $\{0, 1, \dots, 2^8 - 1\}$  中选择。这样就保证每个路由器的相邻路由器的 ID number 都是唯一的。重构攻击路径同一条时, 不会出现重复 ID number, 保证重构能顺利进行。改进的方案还使用 4 位 hashnum 域表示 ID

number 的校验和, 这样就算有攻击篡改标记域中内容, 也可以利用验证这个 hashnum 来舍弃这些被修改过的错误信息, 以最大正确概率重构攻击路径。在使用拓扑图校验的同时, 同时使用 hashnum 校验, 就能最大限度的减少误报。

在 IDCPPM 中, 考虑到离攻击者较近的包很可能被后续标记覆盖, 本文提出一种非覆盖的一致概率来标记并引入判断标记域的方法。在实际网络中, 当很多分组同时到达路由时, 路由来不及同时处理, 就使用“队列”这个概念来处理这些分组, 以至于这些分组不会被随意丢弃。根据 Savage 等的研究, 当包标记概率  $P = 1/d$  时重构攻击路径所需要的包的数据量是最少的。因此, 为了使重构路径的数据包最少, 就要使每个路由的标记信息以相同的概率  $1/d$  到达受害者。类比网络中的“队列”思想, 本文提出一种方案: 在每一个路由器上使用一个变量 count 来记录要覆盖的上游路由标记信息的个数, 通过判断标记域是否为空来决定是否需要记录。对于每一个经过路由器的包, 先判断是否要标记: 1) 若要标记, 则判断标记域是否为 -1; 若为 -1, 表示没有被边界路由器之外的路由器标记, 则可以进行标记; 若标记域不为 -1, 则不能进行标记, count 加 1。2) 若不标记, 则判断标记域是否为 -1, 若为 -1, 表示没有被边界路由器之外的路由器标记, 则可以进行标记, count (count 大于 0) 减 1; 若标记域不为 -1, 则不能进行标记。即在每个路由器上的标记概率都保持在一个一致的概率  $P$ , 标记到包中的路由器信息不会被覆盖, 也不存在某个路由器因“运气不好”而导致出现漏标的情况。每个数据包被  $R_i (i = 0, 1, 2, \dots, n)$  标记后最后达受害者的概率为:  $P_i = p$ 。文中标记概率采取  $p = 0.04$ , 所以  $P_0 = P_1 = P_2 = \dots = P_n$ 。方案 IDCPPM 是需要网络的拓扑图的。拓扑图包含所有的路由器, 路由器的 IP 地址, 路由器的 ID number, 以及它们之间的连接状态。网络上的任何地方拓扑发生改变, 这些改变都将及时发送给含有拓扑图的服务器, 使拓扑图及时得到更新。

### 2.2 IDCPPM 算法描述

改进的包标记算法与基本包标记方案使用类似的存储标记位置。不同的是, 除了 IP 包头的 16 位 Identification 域外, 还征用了标志域(flags)中的第 1 位保留位。因为在 RFC-791 中要求该位必须为 0。但是把该位置 1 不会影响数据包的转发。其标记域选择如图 1 所示。

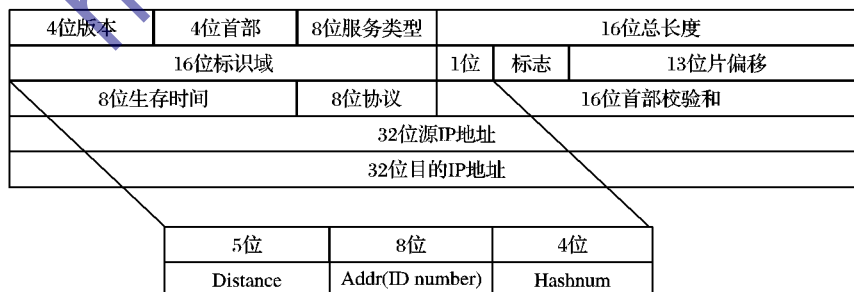


图1 标记域选择

有大量的包在整个标记过程中都未被标记, 所以攻击者可能事先在标记域中写入伪造信息来误导追踪。因此让边界路由器对每个进入自治域网络的包进行标记, 并将标记域 addr 位、hashnum 位和 distance 位置 -1, 从而覆盖可能的伪造信息, 确保后续标记的可靠性。

这里引入一个变量 count 来记录应该标记而没有标记的包的个数。后续的路由器接收到经过它的包, 首先产生一个随

机数  $\alpha$ 。1) 若  $\alpha < p$  ( $p$  为标记概率), 再检查 addr 是否为 -1。若为 -1 则数据包为除边界路由器外尚未被任何路由器标记, 可以标记数据包, 即在 addr 中写入 8 位的 ID number, 在 hashnum 中写入 4 位 ID number 的哈希值, 并将 distance 域置 0; 若 addr 不为 -1, 则将变量 count 加 1, 再看 distance 是否为 0, 若 distance = 0, 则将  $R_i$  的 8 位 ID number 与 addr 异或并重新写入 addr 域; 将 distance 加 1。2) 若  $\alpha > p$ , 再检查 addr 是否

为-1。若为-1则数据包为除边界路由器外尚未被任何路由器标记,可以标记数据包,如果  $count > 0$ ,将  $count$  减1并执行标记操作,即在  $addr$  中写入8位的 ID number,在  $hashnum$  中写入4位 ID number 的哈希值,并将  $distance$  域置0;若不为-1,再看  $distance$  是否为0,若  $distance = 0$ ,则将  $R_i$  的8位 IDnumber 与  $addr$  异或并重新写入  $addr$  域;将  $distance$  加1。

具体标记算法如下:

1)在边界路由器处:

For each packet  $w$ :

$Addr \leftarrow -1$

$Hashnum \leftarrow -1$

$Distance \leftarrow -1$

2)在路由器  $R_i (i = 0, 1, 2, \dots, n)$  处先将路由器的 ID number 哈希。

$newnum = hash(ID)$

$count = 0$

For each packet  $w$ :

Get ( $addr, hashnum, distance$ ) from the packet

Let  $\alpha$  be a random number from  $[0, 1]$

If  $\alpha < q$

If  $addr = -1$

$distance \leftarrow 0$

$addr \leftarrow idnum$

$hashnum \leftarrow newnum$

else

$count = count + 1$

If  $distance = 0$

$addr \leftarrow addr \oplus idnum$

end

$distance \leftarrow distance + 1$

end

else

if  $addr = -1$

If  $count > 0$

$count = count - 1$

$distance \leftarrow 0$

$addr \leftarrow idnum$

$hashnum \leftarrow newnum$

else

If  $distance = 0$

$addr \leftarrow addr \oplus idnum$

end

$distance \leftarrow distance + 1$

end

end

end

在回溯路径时:1)找到  $distance = 0$  的包,验证其8位  $addr$  的哈希值是否等于标记进去的  $hashnum$  值,若相等,说明为某一条攻击路径上的路由;否则,说明包被篡改过,不可信,则丢弃。2)当  $distance \geq 1$  时,将  $addr$  的值与上一跳的各个路由 ID number 值分别先异或再哈希,看是否与  $hashnum$  值相等,若相等,则是此 ID number 的上一条路由;否则继续匹配,直到匹配完一跳的所有包。重复过程2),直到匹配到  $distance = maxdis$  为止。

重构算法

Let  $T$  be the upstream topology graph of the victim  $V$

Let  $Tnew$  be a tree with root  $V$

Let  $last := v$

Let  $maxdis := 0$

Countdis[ $d$ ]:=0

Let  $Mat$  be a table of tuples  $mat(distance, 1/0, countdis[distance])$

For each packet  $w$  from attackers get ( $addr, hashnum, distance$ )

$Mat.Insert\ mat\ (w.\ diatacne, 0, countdis[w.\ distance]) = w.\ addr$

$Mat.Insert\ mat\ (w.\ diatacne, 1, countdis[w.\ distance]) =$

$w.\ hashnum$

Increment countdis[ $w.\ distance$ ]

If  $w.\ distance > maxdis$

$maxdis = w.\ distacne$

End

End

For each distance  $d$  in  $mat$

If  $d = 0$

If  $mat(d, 1, countdis[d]) = hash(mat(d, 0, countdis[d]))$

If ( $T[matnew[d, 0, countdis[d]], vicnode] = 1$ )

Insert edge ( $d, v, mat(d, 0, countdis[d])$ )

End

End

Else  $matnew(d, 0, countdis[d]) = mat(d, 0, countdis[d]) \oplus mat(d - 1, 0, countdis[d])$

If  $mat(d, 1, countdis[d]) = hash(matnew(d, 0, countdis[d]))$

If ( $T[matnew[d, 0, countdis[d]], mat(d - 1, 0, countdis[d])] = 1$ )

Insert edge( $d, matnew(d - 1, 0, countdis[d - 1]), matnew(d, 0, countdis[d])$ )

End

End

End

Extract the attack paths ( $v \dots R_i \dots R_{maxdis}$ )

### 3 性能分析

#### 3.1 最弱链

前面介绍了 PPM 和自适应包标记算法中存在的最弱链和次弱链问题。IDCPPM 采用一致概率来标记数据包,每个数据包被  $R_i (i = 0, 1, 2, \dots, n)$  标记后最终达受害者的概率为:  $P_i = p$ , 所以有  $P_0 = P_1 = P_2 = \dots = P_n = p$ 。即 IDCPPM 解决最弱链问题的同时,不存在自适应标记概率中的次弱链问题。理论上保证了回溯需要的数据包最少。

#### 3.2 收敛时间

假设需要  $k$  个 fragment 才能组合出一个边界的 IP, 路由器以概率  $p$  标记数据包, 当重构路径长度为  $d$  时, 可以知道需要的数据包的期望值为  $E(x) < k * \ln(kd) / (p(1-p)^{d-1})$ 。要想组合出一个完整的边信息, 在 PPM 中重构攻击路径至少需要 8 个有效数据包, 在自适应包标记中至少需要 4 个有效的数据包, 而 IDCPPM 方案只需要一个数据包就可。可以看出:

$$E(x)_{IDCPPM} = \ln(d) / (p(1-p)^{d-1})$$

$$E(x)_{\text{自适应}} = 4 * \ln(4d) / (p(1-p)^{d-1})$$

$$E(x)_{PPM} = 8 * \ln(8d) / (p(1-p)^{d-1})$$

$$\text{因此, } E(x)_{IDCPPM} < E(x)_{\text{自适应}} < E(x)_{PPM}$$

#### 3.3 复杂度

重构攻击路径时, 复杂度与标记的地址的分片数相关。假设某个距离  $d$  处有  $m$  个攻击者, 则受害者将收到  $m \times k$  个含有分段信息的数据包。每个偏移都有  $m$  个分片, 共有  $k$  个偏移, 所以在组合的时候可能的 IP 有  $m^k$  种情况, 为了找到正确的路由 IP 需要做 Hash  $m^k$  次。在 PPM 中,  $k = 8$  所以需要 Hash 的次数为  $m^8$ , 复杂度为  $O(m^8)$ ; 在自适应包标记中,  $k = 4$  需要组合的次数为  $m^4$ , 复杂度为  $O(m^4)$ ; 而在 IDCPPM 中,  $k = 1$  不需要组合, 只需要哈希并比较  $m$  次, 即复杂度为  $O(m)$ 。

#### 3.4 误报率

前面已经说明 PPM 算法的误报率为  $1 - (1 - 1/2^{32})^n$ ,  $n = m^8$ ; 自适应包标记算法的误报率为  $1 - (1 - 1/2^{36})^n$ ,  $n = m^4$ , 其中  $m$  为攻击源个数,  $k$  为分片数。IDCPPM 中使用4位哈希函数, 这个4位哈希函数只是用来验证 ID number, 使被篡改过的 ID number 不能被通过验证, 所以误报率为:  $1 - (1 - 1/2^{12})^n$ ,  $n = m$ , 实验结果证明 IDCPPM 算法和自适应算法的误报率都很低, 都比 PPM 都要低。



IDCPM 除了在以上四个方面效果比较好,还有以下两个优点。

1) 标记空间。标记位一直是包标记方法的关键,而实际上能标记的位数是有限的。IDCPM 方案中使用标记域是否为空来判断此数据包是否已经标记,并达到非重复标记的目的,而不是使用标记域中的一位标记位,这样就能有多一位的标记空间来标记更重要的信息。

2) 扩展性。IDCPM 在使用拓扑图的时候,使用 8 位的 ID number 来表示路由器,还能更好地应用于 IPv6。32 位 IPv4 地址即将耗尽,应运而生的 128 位 IPv6 地址替代 IPv4 是大势所趋,但其道路曲折而漫长,这将导致在很长一段时间内 IPv6 和 IPv4 是并存在 Internet 中的,标记 32 位 IP 地址的方法显而易见地不能解决同时标记 128 bit 和 32 bit 的问题,而 IDCPM 所提的方法使用统一分配的 8 bit ID number 能很好地克服其缺点,更好地实施。将来,在 IPv6 的时代,要使用包标记方法就要标记 128 bit IP 地址,在标记域有限的情况下,IP 就需要分成更多的片,重构时就要组合更多的次数,使重构算法的复杂度大幅增加,误报率增大;而新的方法有很好的扩展性,能持续运用。

#### 4 实验结果

研究表明:受害者重构出攻击路径需要的最少的数据包是衡量概率包标记方法的收敛时间,即包标记方法的关键。为了研究说明新方法,选取了一条长度从 2 到 30 跳的路径来实验,让实验运行不同的时间  $t_1, t_2, t_3$ , 分别对每个时间段的实验数据进行 100, 500, 800, 1 000 次取样回溯攻击路径取平均值得到  $aver_{ij} (i = 1, 2, 3; j = 1, 2, 3, 4)$ , 求和取平均值  $aver = (\sum_{i=1}^3 \sum_{j=1}^4 aver_{ij}) / 12$ 。然后依据各自不同的概率以及标记过程模拟了 PPM, 自适应包标记方法和新提出的 IDCPM 标记方法,重构攻击路径需要的包数量结果如图 2, 可以看出实验结果与 3.2 节所分析的结果相一致,也说明 IDCPM 方法更简单,有效。在第 3.4 节分析了误报率,这里给出模拟图,如图 3 所示。

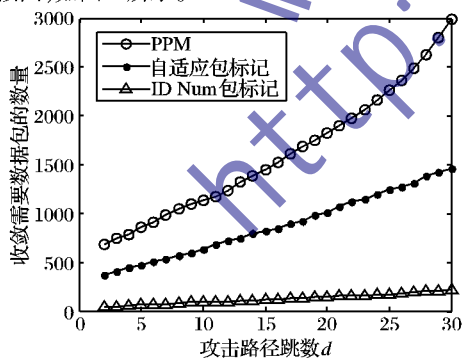


图2 重构攻击路径需要数据包的数量

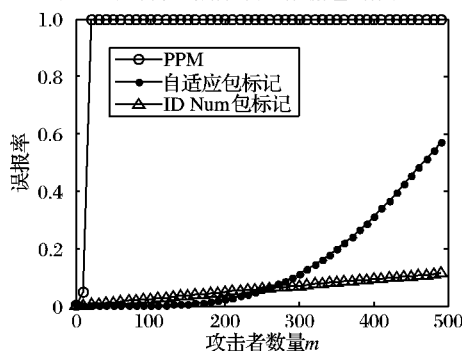


图3 多个攻击者时的误报率

当有 20 个以上的攻击者时,PPM 的误报率几乎为 1,追踪是不可行的。在有 250 个以下的攻击者时,这两种算法的误报率都较低,但是 IDCPM 的误报率比自适应算法略高,当攻击者达到 250 以上时,自适应算法的误报率开始快速升高,而 IDCPM 的误报率仍然比较低。

#### 5 结语

IDCPM 追踪方案使用非覆盖的一致概率标记思想和一个 8 位的 ID number 来表示路由器的相邻路由,能最大限度地解决 PPM 的最弱链和收敛时间长问题,得到比 PPM 和自适应标记算法更好的效果。其不分片的标记思想,可以减少重构路径时组合的次数,降低复杂度,同时误报率还能保持在一个比较低的比率。它还能更好地适用于 IPv6 中,有很好的扩展性。这个方法只能在同一个自治域中进行,能很好地限制攻击者的伪造能力但同时也限制了追踪的范围。

#### 参考文献:

- [1] EHRENKRANZ T, LI J. On the state of IP spoofing defense [J]. ACM Transactions on Internet Technology, 2009, 9(2): 1 - 29.
- [2] SAVAGE S, WETHERALL D, KARLIN A, et al. Practical network support for IP traceback [C] // Proceedings of the ACM SIGCOMM 2000 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication. New York: ACM, 2000: 295 - 306.
- [3] 李德全, 苏璞睿, 魏东梅. 基于路由器编码的自适应包标记 [J]. 软件学报, 2007, 18(10): 2652 - 2661.
- [4] 闫巧, 宁士文. 重叠哈希分片的概率包标记方法 [J]. 计算机工程, 2011, 37(11): 11 - 14.
- [5] 章海聪, 王晓明. 基于路由器接口的 IP 追踪方案 [J]. 计算机应用, 2011, 31(3): 774 - 777.
- [6] 徐劲松. 一种改进的路由包标记追踪方案 [J]. 计算机应用, 2009, 29(5): 1316 - 1320.
- [7] XI ANGY, ZHOU W L, GUO M Y. Flexible deterministic packet marking: An IP traceback system to find the real source of attacks [J]. IEEE Transactions on Parallel and Distributed System, 2009, 20(5): 567 - 580.
- [8] MUTHUPRASANNA M, MANIMARAN G, MANZOR M, et al. Coloring the Internet: IP traceback [C] // Proceedings of the 12th International Conference on Parallel and Distributed Systems. Washington, DC: IEEE Computer Society, 2006: 589 - 598.
- [9] GONG C, SARAC K. A more practical approach for single-packet IP traceback using packet logging and marking [J]. IEEE Transactions on Parallel and Distributed System, 2008, 19(10): 1310 - 1324.
- [10] 揭振, 孙乐昌. 一种新的非重复性包标记 IP 追踪方案 [J]. 计算机工程, 2007, 33(10): 105 - 107.
- [11] LIU J, LEE Z-J, CHUNG Y-C. Dynamic probabilistic packet marking for efficient IP traceback [J]. The International Journal of Computer and Telecommunications Networking, 2007, 51(3): 866 - 882.
- [12] CHEN R L, PARK J M, MARCHANY R, et al. RIM: Router interface marking for IP traceback [C] // Global Telecommunications Conference. San Francisco, USA: IEEE, 2006: 1 - 5.
- [13] SNOEREN A C, PARTRIDGE C, SANCHEZ L A, et al. Hash-based IP traceback [C] // Proceedings of the ACM SIGCOMM 2001 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication. New York: ACM Press, 2001: 3 - 14.
- [14] 李德全, 苏璞睿, 冯登国. 用于 IP 跟踪的包标记的注记 [J]. 软件学报, 2004, 15(2): 250 - 258.
- [15] 李德全, 徐一丁, 苏璞睿, 等. IP 追踪中的自适应包标记 [J]. 电子学报, 2004, 32(8): 1334 - 1337.