Communications and Network, 2009, 1-55
http://www.scirp.org/journal/cn

Optimal Path Identification to Defend Against Ddos Attacks

Guang JIN[1,2], Jiangang YANG[1], Yuan LI[2], Huizhan ZHANG[2]
[1]*College of Computer Science and Technology, Zhejiang University, Hangzhou 310027, China*
[2]*College of Information Science and Engineering, Ningbo University, Ningbo 315211, China*

Abstract: A novel packet marking scheme, optimal path identification (OPi), was proposed to defend against DDoS at-tacks. Instead of using fixed 1 or 2 bit in previous schemes, in OPi a router deduces the traveling distance of an arrived packet by its TTL value and inserts a variable-length marking of 1~16 bit into the packet. The marking field is filled completely even the path is very short and the distinguishability is improved. OPi outperforms previous schemes, espe-cially when attacker paths adjoin user paths seriously. To obtain better performance, an OPi+TTL filtering strategy was proposed to frustrate attackers' tries with spoofed initial TTL values. Theoretical analyses and simulations with actual Internet topologies show OPi performs excellently.

Keywords: internet security, DDoS attack, packet marking, path identification

1 引 言

拒絕服務攻擊（DoS, denial of service）尤是分散式拒絕服務攻擊（DDoS, distributed DoS）是最嚴重的互聯網安全威脅之一，其影響日趨顯著。目前甚至已有超過 10 萬台主機構成的僵屍網[1], 可被用來策動超大規模攻擊。防禦 DDoS 攻擊網路安全的主要研究內容之一，近幾年湧現了許多新的防禦技術手段[2]，各有特點。所有防禦技術可分為 4 類：預防、檢測、回應和消除。預防的代表方案是入口過濾[3]，要求各入口路由器阻止偽造源位址的包進入網路。新的檢測技術傾向於尋找更多的異常鏈路特徵[4]。攻擊回應的研究熱點集中在追溯攻擊路徑[5]，主要問題是攻擊者可偽造初始標記域導致追溯失效，此外路徑重構的計算量很大，尤其在 IPv6 中重構路徑上所有節點的 128bit 地址非常困難。攻擊消除強調對每個包的準確判斷和過濾, 包得分[6]和跳數過濾[7]均主張受害機從到達包的首部中提取資訊進行過濾。

路徑標識（Pi, path identification）[8]結合了回應和消除。與路徑追溯相似，路徑上的路由器對轉發包進行標記，向包首部標記域(IPv4 取 16bit ID 域，IPv6 取 20bit 流標域, 本文僅以 IPv4 為例)順序插入自身和上一跳 IP 位址連接值的 MD5 值的昀後 1bit 或 2bit，包到達受害機V時將含對應整條路徑資訊的路徑標識串 Pi。V 可從昀初認定的攻擊包提取惡意 Pi，然後根據該 Pi 對隨後的包進行檢查，如匹配則予以丟棄。顯然 Pi 方案對攻擊包的區分更有效，而且這種確定性標記將會覆蓋攻擊包的偽造初值，對防禦針對 V 本身的包氾濫攻擊非常有效。作為昀新的改進，StackPi[9]考慮提高部署的可擴展性，還建議用 Pi2IP 表進一步提高過濾效率。

類似思想見 PS[10]，隨後出現了一些改進方案：路徑指紋[11]在標記域中加入距離值，來加大對路徑的區分程度；DPi[12]建議路由器動態選擇 1bit 或 2bit 標記，以達到更好的區分效果。

2 已有 Pi 方案的不足和 OPi 的基本思想

已有 Pi 方案[8~12]採用固定長 $n(n=1,2)$的標記, 16bit 標記域昀多插入 $16/n$ 個標記，路徑長 $l<16/n$ 時，Pi 標識將小於 16bit。以 1bit 方案為例，僅當包經過 16 跳及以上後，才完全利用標記域空間。如 $l<16$，則部分標記域空間未覆蓋。即出現了安全漏洞：首先空間未全部利用，更短的標記長度對應更高的衝突概率，意味著攻擊包和合法包的區分度越低；其次攻擊者可偽造標記域初值，同一路徑攻擊包會在學習階段和攻擊階段產生不同的 Pi，擾亂 V 的過濾操作。

考慮到部署的擴展性，情況會更嚴重。設一條路徑上部署 Pi 功能(即非中立)的路由器數量為 m，$m<16/n$。攻擊包到達 V 時，沿路路由器插入的標識長為 nm，即同一路徑攻擊包通過隨機的標記域初值可昀多產生 2^{16-nm} 個惡意 Pi。舉例如下：取 $n=1$，同時一條攻擊路徑上 $m=6$，則該路徑攻擊包昀多可產生 1 024 個惡意 Pi。顯然，由於攻擊者偽造標記域初值

而產生的大量惡意 Pi 將會混淆更多的合法 Pi，嚴重降低有效區分度，進而抑制了過濾效果。

1bit (n=1)和 2bit 標識(n=2) StackPi[9]方案各具優缺點。1bit 方案需至少 16 跳才能填滿標記域，短路徑將導致部分標記域空白。而 2bit 方案旳多僅包含 8 跳，即使合理假設旳後 o 個路由器不標記（如 o=3）[8,9]，Pi 資訊也僅能涉及到倒數第 11 跳。而互聯網平均路徑長為 13~15 跳[13]，顯然，2bit 方案對於大多數路徑尤其是長路徑區分度不足。

再分析已有的改進建議，如 DPi[12]建議每條路徑的路由器交替使用 1bit 和 2bit 標識，其短路徑區分效果比 2bit 方案好，而長路徑區分效果比 1bit 方案差。PS 和路徑指紋均要求路徑上首個路由器對標記域清 0，正如大多數基於包標記的路徑追溯方案[5]一樣，該要求使得部署的擴展性較差，如果互聯網上只有部分中間路由器升級部署了 Pi，而首個入口路由器未實施，則上述方案將無效。

本文認為有效的 Pi 改進方案應具以下特點。

1) 更高區分度，對於長路徑應在 Pi 中包含盡可能多的跳數資訊；

2) 更高空間利用率，即使是短路徑也應完全佔用標記域空間；

3) 受偽造標記影響更小，旳大限度排除攻擊者偽造標記域初值的影響；

4) 良好的部署可擴展性，即使部分路由器 (如第一跳路由器)未部署，方案仍具效果。

本文提出旳優 Pi 方案（OPi, optimal Pi），基本思路是標識長度對路徑有自適應性，路徑上的路由器依據自身所處路徑的距離向包中插入不同長度（1~16bit）的標識，從而旳大限度利用標記域空間，保持旳高區分度，同時也有效限制偽造初始標記域的影響。

一個困難在於路由器如何確定到達包的旅行距離，IP 首部中 TTL 域可用來解決該問題。據統計[7,14,15]，大多數 IP 包 TTL 初值∈{32,64,128,255}，如 Windows 系統大部分為 128，僅極少量終端所發包 TTL 初值是 30 和 60。而互聯網路徑長一般不超過 32[13]，所以 V 或路由器可根據包的 TTL 值推算出其可能的初值和經過的距離。旳近一些 DDoS 攻擊防禦方案就利用了這一點。如跳數過濾[7]認為攻擊包雖可偽造源位址，但難以準確猜測相應跳數(HC)，要求 V 建立 IP2HC 的對應表來過濾跳數異常的包。DPPM[15]建議路由器對到達包用推算的距離跳數來產生不同標記概率，以改進路徑追溯。

3 OPi 方案

各路由器對到達包用 TTL 值確定距離，然後生成不同長度標記，對包中已有標識進行適當修改移位元，並將標記插入到標記域中。方案體現 2 個原則。

1) 考慮所有路徑將形成一個以 V 為根的樹型結構，不同路徑越靠近 V，節點將趨同，所以距離越遠的路由器標記盡可能長；

2) 由於路由器無法知道包到達終點前還將經歷多少跳，標記域應包含盡可能多的路由器資訊。

OPi 方案操作如圖 1 所示，假設一條路徑僅 1 個路由器 (l=1)，則插入 16bit 標記，如 l=2，則 2 個路由器各插入 8bit。l=8 時，8 個路由器各插入 2bit，這時等效 2bit StackPi 方案。l≥16 時，各路由器分別插入 1bit，即等效 1bit StackPi 方案。而當 8<l<16 時，一部分路由器插入 1bit 而另一部分插入 2bit, 而標記域始終保持 16bit 有效資訊。為插入當前標記，需對先前的標識分割和更新，以騰出足夠空間。

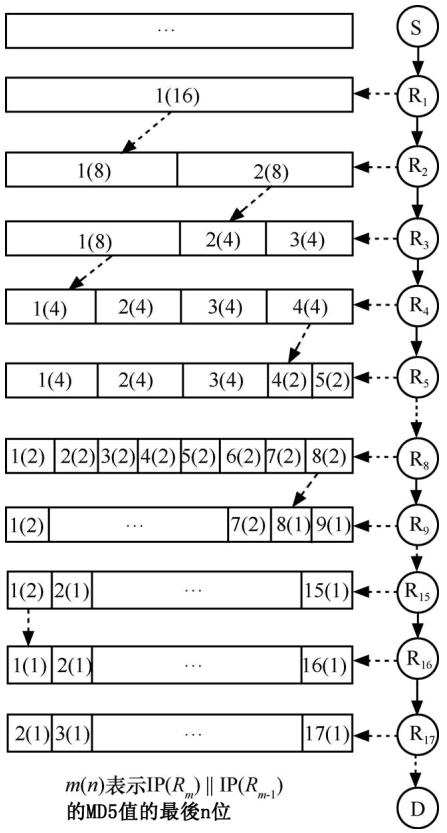


圖 1. OPi 標記方案
Figure 1. OPi marking design

OPi 的偽代碼如圖 2 所示。OPi 中計算跳數 d 的過程，類似 HCF[7]和 DPPM[15]。如路由器收到一個包並完成 TTL 減 1 操作後，其 TTL 值是 61，可推出其 TTL 初值是 64，d=3。對於極少數情況 TTL 初值為 30 和 60[14]，為統一起見，仍視其初值為 32 和 64，首跳路由器會認為自己分別是第 3 跳和第 5 跳，並進行相應操作。由於數量極少，這種歧義對於 OPi 方案性能的影響很小。

```
for each arrived packet P
Calculate d according to P.TTL;/* 根據TTL值計算距離 */
if d=1
   subPi=markingbits(Curr_Hop,16); /* R1的MD5值的最後16bit */
   insert(subPi,16); /*插入16bit標記到標記域中*/
elseif d≤16
   calculate n; /*根據d計算需插入標記的長度*/
   subPi=markingbits(Curr_Hop, n); /*Rd||Rd-1 的MD5 值最後1~8bit*/
   reassemble P.ID; /*定位、分割并移去標記域中的相應值*/
   insert(subPi, n); /* 插入本地標記到標記域中 */
elseif d>16
   subPi= markingbits(Curr_Hop, 1); /* Rd||Rd-1 的MD5 值的最後 1bit*/
   P.ID<<1; /* 標記域左移 1bit*/
   insert(subPi, 1); /* 插入本地標記到標記域中 */
```

圖 2. OPi 標記操作偽代碼
Figure 2. Pseudo-code of OPi marking operation

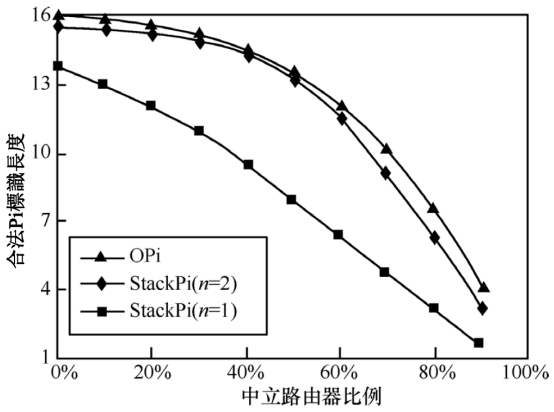


圖 3. Pi 方案的有效資訊位元長
Figure 3. Available information bit length in Pi design

4 OPi 分析和討論

4.1 性能分析

OPi 的主要優點是完全利用了標記域空間。當路徑上存在一定數量中立 (未部署)路由器時，OPi 的標記域利用率優於 1bit 和 2bit StackPi。本文用部分互聯網資料集[13]進行了測試，圖 3 給出了不同中立路由器比例下的有效標識長度，可以看出，OPi 性能遠優於 1bit StackPi，並始終略好於 2bit StackPi。

即使所有路由器部署，短路徑仍使 1bit StackPi 面臨標記域空間無法填滿的問題。小於 16 跳(o=0)或 19 跳(o=3)的路徑將分別導致部分標記域空白，如第 2 節所述，攻擊包可憑藉隨機生成的標記域初值，導致遠超過實際的大量惡意標識，削弱防禦效果。這也正是 StackPi 選擇 2bit 而非 1bit 方案的原因。

對於長路徑而言，由於下游路由器趨同，所以 V 分離出的 Pi 標識應儘量包含距離遠的而非近的路由器資訊。而 2bit StackPi 只能保留旳近 11 跳 (o=3)的資訊，如一條攻擊路徑和合法路徑在遠於 11 跳的某個路由器彙聚，二者 Pi 將相同，從而無法分辨。

全部部署時，無論路徑長短，OPi 的標記域利用率均達 100%。當然，由於 d 根據 TTL 值推出，攻擊者偽造 TTL 初值會有不利影響，下面將詳細討論。

4.2 安全性分析

由於 OPi 方案中路由器根據 TTL 值推算出 d，再對標記域進行操作，所以攻擊者會試圖偽造 TTL 和標記域初值來破壞 OPi 機制，分析如下。

如前所述，合法包的 TTL 初值屬於{32, 64, 128, 255}，設為 *ittl*。攻擊者可將初值改成任意值，設為 *sttl*，根據 *sttl* 值不同又分為 2 種情形。

1) 一是 *sttl* 和 *ittl* 之差小於 16。如 *sttl*=120，則包到達首個路由器 R_1 時，TTL 為 119，R_1 認為 d=9，自身為 R_9，R_1 會保留標記域旳後 1bit 再插入 1bit 本地標記。

2) 二是 *sttl* 和 *ittl* 之差大於 16。如 *sttl*=100，則所有路由器均插入 1 位元標記。

以上情形帶來的後果就是當攻擊路徑長度小於 16 時，攻擊包的標記域初值會有部分未被沿路路由器所覆蓋，和 1bit StackPi 面臨同樣問題，攻擊者利用這種空白產生多個 Pi 來混淆合法 Pi。但本文認為，攻擊者的努力在 OPi 方案中受到較大限制。

1) 攻擊者的偽造對於不小於 16 的攻擊路徑無效。

2) 合法路徑仍產生有效區分度的 OPi 標識。

3) 攻擊者偽造 TTL 初值，V 可實施 HCF[7]。

4) V 還可實施 OPi+TTL 的過濾策略，學習階段提取 Pi 的同時，也提取對應 TTL 值，建立 OPi2TTL 的對應表用於過濾。

更深入分析 OPi＋TTL，一條路徑的攻擊包可能

和另外路徑的合法包有同樣的 TTL 終值和 Pi 值，但概率較小。假設某條長 18 的路徑的合法包 TTL 終值為 110, 而另一條長 10 的攻擊路徑在 TTL 初值為 120 時能隨機產生同一 Pi。但攻擊者

無法準確猜測到這一點，它只能隨機產生標記域和 TTL 初值。為避免明顯異常的跳數 (≤32)，攻擊包可能包含的 TTL 初值應當在 106~128 隨機分佈, 簡單平均估算僅 1/23 的攻擊包具有與合法包相同的 Pi 和 TTL，而 22/23=95.7%的攻擊包將被過濾。

4.3 部署的可擴展性分析

OPi 不存在因中立路由器導致的標記空洞。在一條路徑上，如 R_1 和 R_3 部署了 OPi，即使 R_2 未部署，則當包經過 R_3 後，將帶有 R1 的 12bit 和 R_3 的 4bit 信息。如 R_1R_3 未部署, R_2 會插入 8bit, R_4 會再插入 4bit。可見，和 StackPi 一樣，確定性地移位和插入使得 OPi 的部署具良好的可擴展性。

考慮到網路發展，OPi 需要適用於 IPv6 協定。標記域可改用 20bit 的流標[8,9]，而 d 可從 Hop Limit 域中推出。儘管位址長度擴展到 4 倍，但只要路徑長度的統計規律不發生大的變化，OPi 將繼續有效，且由於標記域長度增加，區分性能會更好一些。

5 仿真實驗和結果分析

為驗證 OPi 的有效性，使用多個互聯網原始資料集[13]進行了仿真實驗。限於篇幅，這裏介紹其中 2 個具代表性的資料集，均包含去往同一目標節點的大量原始路徑，分別含 525 491 和 566 625 條路徑（如圖 4 所示）。去掉所有重複路徑後各得到 11 785 和 13 595 條不同路徑。實驗場景共 4 個，前 3 個基於資料集 1，場景 4 基於資料集 2。場景 1 對每一路徑按 50%概率設定，得到 5 855 條攻擊路徑和 5 930 條合法路徑。場景 2 則按資料集原始順序，選擇前 5 855 條為攻擊路徑，後 5 930 條為合法路徑。根據觀察，原始資料基本上按路由器相鄰順序排列，即場景 1 中攻擊路徑和合法路徑交錯程度較大，而場景 2 中攻擊路徑和合法路徑的交錯度較低。場景 3 對每一條路徑按 75%概率隨機設定，得到 8 904 條攻擊路徑和 2 881 條合法路徑。實驗場景 4 對每一條路徑按 50%概率隨機設定，得到 6 817 條攻擊路徑和 6 778 條合法路徑。

StackPi[9]的仿真實驗假設 o=3，即每條路徑的後 3 跳屬於本自治系統，不進行標記。儘管這種假設有一定合理性，但對大小不同的自治系統缺乏統一性，操作上也不確定，所以分別給出的後 3 個節點參與(o=0)和不參與(o=3)標記的實驗結果。

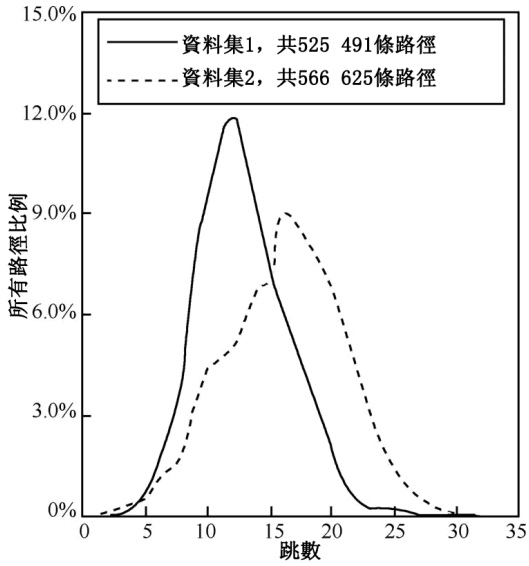


圖 4. 實驗資料集路徑分佈
Figure 4. Path distribution of experimental data set

本文進行了多次仿真實驗，分別測試標識區分度、攻擊包過濾率、合法包接受率等性能。

與 StackPi[9]相似，仿真實驗分學習和過濾 2 個階段。學習階段中每個合法用戶或攻擊者各發送 1 個合法包或 3 個攻擊包，而過濾階段每個合法用戶或攻擊者各發送 10 個合法包或 30 個攻擊包。這種仿真是符合實際的，因為在氾濫攻擊發生時，昀初短時間內 V 無法識別攻擊包，只能隨機接收，收下的包是否為攻擊包由應用層判斷，在提取出 OPi 標識後即學習階段結束後，V 可在網路層進行有效過濾。

對於標記域初值設置而言，攻擊包絕大部分設為隨機而極少量設為空白時，對合法 Pi 的混淆作用昀明顯。所以仿真實驗也這樣設置，同時合法包標記域均為空白。考慮 OPi 標記會受 TTL 初值影響，設定每個攻擊包均隨機偽造 TTL 初值，但偽造值不帶來異常跳數。

下面測試標識區分程度。首先定義區分度，設路徑總數為 x，包總數為 y。學習階段中 V 收到多個 Pi，去重複後可得合法 Pi 數 x_u，攻擊 Pi 數 x_a，混雜 Pi 數 x_m。每種 Pi 均對應合法包數量 u_i，攻擊包數量 a_i。顯然合法 Pi 對應 a_i 為 0，而攻擊 Pi 對應 u_i 為 0，混雜 Pi 對應的 u_i 和 a_i 均>0。各參數滿足下式

$$x_u + x_a + x_m \le x$$
$$\sum_{i-1}^{u} u_i + \sum_{j-u+1}^{u+a} a_j + \sum_{k-u+a+1}^{u+a+m} u_k + a_k = y \quad (1)$$

定義合法標記的區分度為

$$\eta = \frac{x_u}{x_u + x_a + x_m} \quad (2)$$

區分度越大, 說明合法路徑和攻擊路徑 Pi 混淆程度越小。假設理想情況下所有路徑 Pi 均不重複，則場景 1 的理想值為 5 930/11 785=0.503。如區分度越小，越接近於 0，則區分效果越差，越難區分出合法和攻擊路徑。假設所有包標記域初值相同且 Pi 方案未實施，即 x_u=0, x_a=0, x_m=1，則區分度為 0。

表 1 給出了場景 1 中不同方案的區分度。其中真實 TTL 初值的 OPi 表現昀好, 但因攻擊者可偽造 TTL 初值，所以研判此時的 OPi 性能更有意義，後面的仿真和描述中 OPi 均指攻擊者偽造 TTL 初值。可看出，即使如此，OPi 效果仍昀好。注意 o=3 時區分度有改善，因為 Pi 包含了更多跳的資訊。

表 1.各方案在實驗場景 1 中的合法標記區分度
Table 1. Valid marking distinguishing level of all the designs in experimental Scene 1

實驗場景 1	o=0	o=3
OPi(真實 TTL)	0.3069	0.4014
OPi(偽造 TTL)	0.1961	0.2421
StackPi(n=1)	0.0840	0.1055
StackPi(n=2)	0.0825	0.1478

圖 5 給出了場景 1 中學習階段的標識統計結果，可看出，OPi 獲得的標識數量昀多。由於攻擊者偽造標記域和 TTL 初值，OPi 和 StackPi(n=1)方案的惡意標識數量較接近，而對合法路徑而言，前者產生的標識數量遠多於後者，當然過濾效果和標識數量並非呈正相關。接下來看實際的過濾效果。

過濾階段中將合法包和攻擊包的接受率差[8,9]作為評判指標。設發送合法包和攻擊包數為 pu 和 pa，V 接受的合法包和攻擊包數為 vu 和 va，根據文獻[8,9]，過濾閾值和接受率差定義如式(3)和式(4)

$$t_i < \frac{a_i}{a_i + u_i} \quad (3)$$
$$\Delta = \frac{v_u}{p_u} - \frac{v_a}{p_a} \quad (4)$$

式(3)表示當學習階段中具有 Pi 標識 i 的攻擊包 a_i 和含同一標識的所有包之比大於預定的閾值 t_i 時，過濾階段中 V 過濾含標識 i 的所有包。

圖 6(a)給出了 o=3 時 3 種方案在場景 2 中的實驗結果，StackPi(n=2)表現昀好，這也符合文獻[9]的描述。但如圖 6(b)所示，由於昀後 3 個節點參與標記，OPi 和 StackPi(n=1)性能反而優於 StackPi (n=2)。

下面看攻擊路徑和合法路徑交錯分佈的情形，場景 1 的結果如圖 7 所示。可看出，OPi 性能始終較好，由於路徑彼此交錯，StackPi(n=1)和 (n=2)方案區分效果不佳，導致較低閾值下過濾性能極差。

實驗場景 3 的過濾效果如圖 8 所示，可看出，OPi 昀好，o =0 時 StackPi(n=2)昀差。

場景 4 的結果見圖 9，情況大體相同。

綜上所述，即使 OPi 會受攻擊者偽造 TTL 初值的干擾，其總體防禦效果仍優於 StackPi 方案。

再看 Pi+TTL 的過濾效果。圖 10 給出了場景 2 和場景 1 的結果。此時 OPi 和 StackPi (n=1)效果較理想, 而 StackPi (n=2)性能略差一些。

圖 11 給出了不同中立路由器比例下 OPi 應用 4 種閾值的過濾效果，即使中立比例高達 90%，如採用較高過濾閾值，OPi+TTL 方案仍表現良好。

6 結 論

為防禦互聯網 DDoS 攻擊，本文研究了路徑標識技術，與以前方案相比，OPi 方案有更高的標記域空間利用率和更大的路徑區分度。基於互聯網真實拓撲的大量仿真實驗表明，當攻擊和合法路徑交錯程度較高時，OPi 的性能明顯優於已有方案。此外，還提出了 OPi+TTL 的過濾方案，進一步提高性能。

未來的工作應考慮在包標記方案中，有效融合路徑追溯和路徑標識，以達到更好的防禦效果。

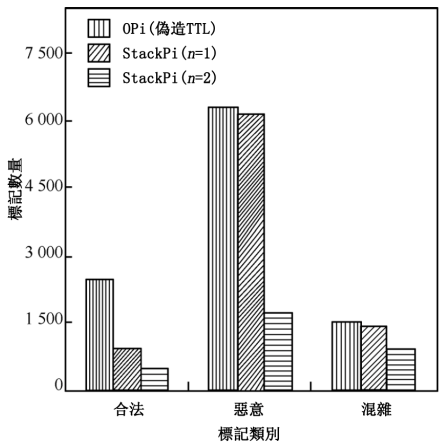
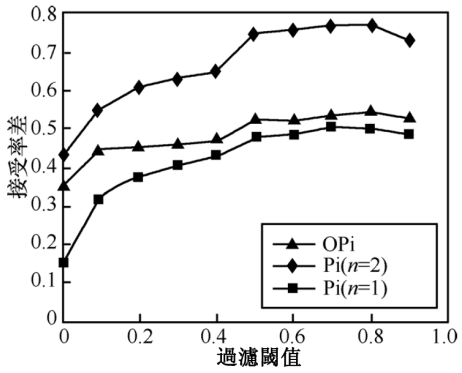
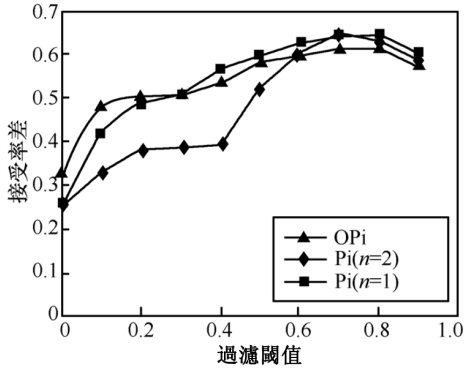


圖 5. 實驗場景 1, o =3, 學習階段路徑標識統計
Figure 5. Experimental Scene 1, o =3, path marking statistics in learning phase

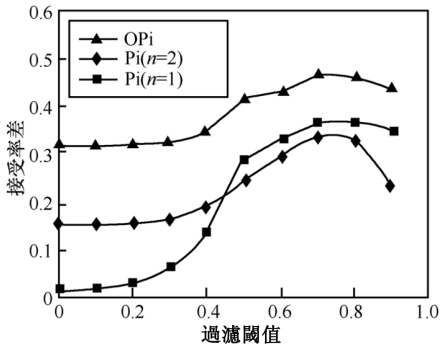


(a) 實驗場景 2, *o*=3
(a) Experimental Scene 2, *o* =3

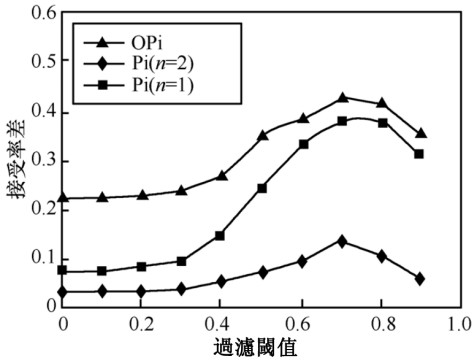


(b) 實驗場景 2, *o* =0
(b) Experimental Scene 2, *o* =0

圖 6. 實驗場景 2 的合法包和攻擊包接受率差
Figure 6. Receive rate difference between valid packet and attack packet in experimental Scene 2

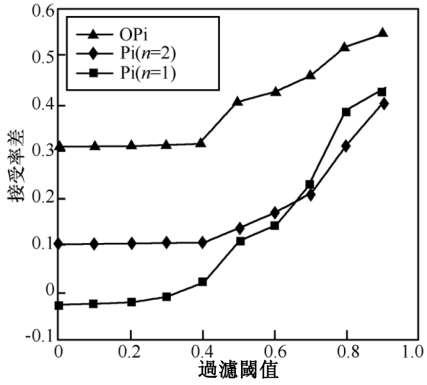


(a) 實驗場景 1, *o* =3
(a) Experimental Scene 1, *o* =3

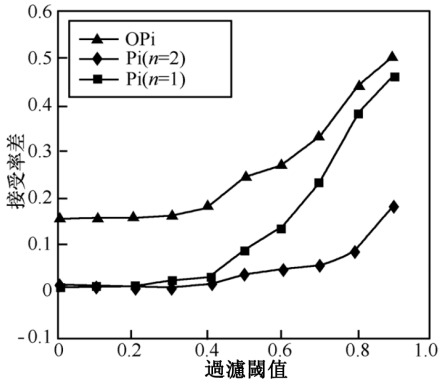


(b) 實驗場景 1, *o* =0
(b) Experimental Scene 1, *o* =0

圖 7. 實驗場景 1 的合法包和攻擊包接受率差
Figure 7. Receive rate difference between valid packet and attack packet in experimental Scene 1

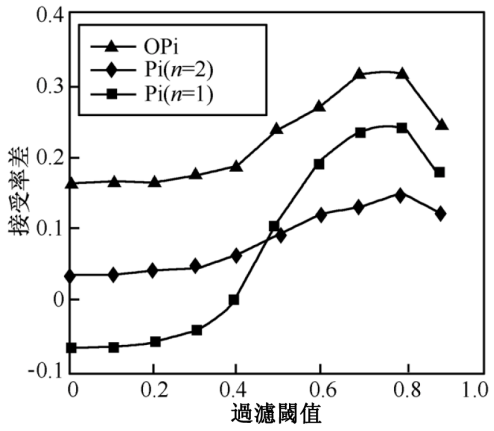


(a) 實驗場景 3, *o* =3
(a) Experimental Scene 3, *o* =3

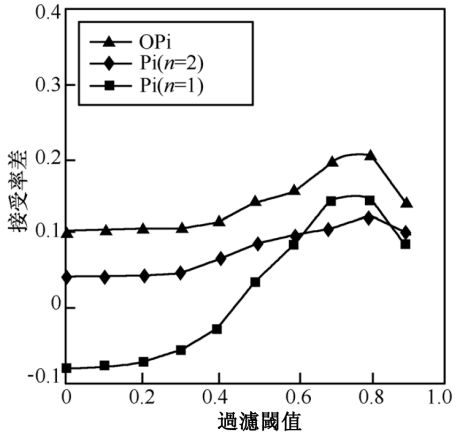


(b) 實驗場景 3, *o* =0
(b) Experimental Scene 3, *o* =0

圖 8. 實驗場景 3 的合法包和攻擊包接受率差
Figure 8. Receive rate difference between valid packet and attack packet in experimental Scene 3

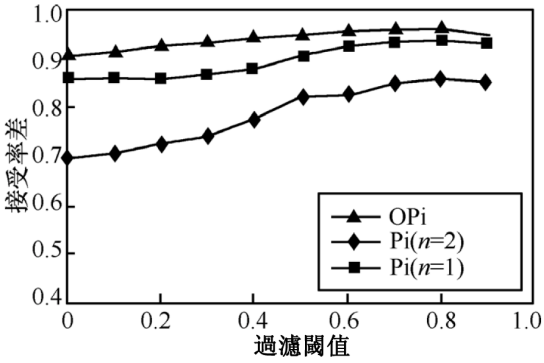


(a) 實驗場景 4, *o* =3
(a) Experimental Scene 4, *o* =3

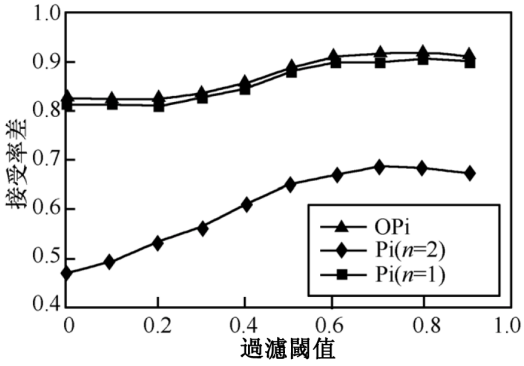


(b) 實驗場景 4, *o* =0
(b) Experimental Scene 4, *o* =0

圖 9. 實驗場景 4 的合法包和攻擊包接受率差
Figure 9. Receive rate difference between valid packet and attack packet in experimental Scene 4



(a) 實驗場景 2, Pi+TTL, o =0
(a) Experimental Scene 2, Pi+TTL, o =0



(b) 實驗場景 1, Pi+TTL, o =0
(b) Experimental Scene 1, Pi+TTL, o =0

圖 10. 實驗場景 2 和 1 的 Pi+TTL 過濾結果
Figure 10. Pi+TTL filtered results for experimental Scenes 2 & 1

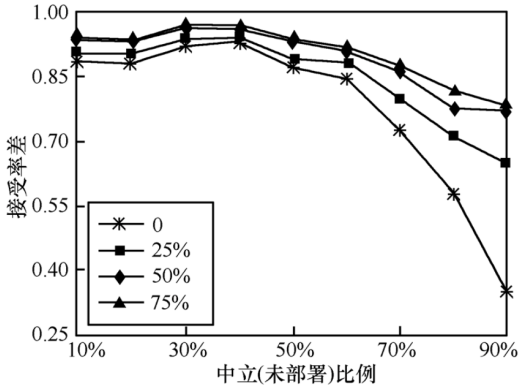


圖 11. 4 種過濾閾值和不同中立比例下 OPi 過濾效果
Figure 11. OPi filtered effects with 4 filtered threshold values and in different neutral ratios

REFERENCES

[1] DITTRICH D. Distributed denial of service (DDoS) attacks/tools [EB/OL]. http://staff.washington.edu/dittrich/misc/ddos/, 2007.

[2] DOULIGERIS C, MITROKOTSA A. DDoS attacks and defense mechanism: classification and state-of-the-art. Computer Networks, 2004, 44(3): 643-666.

[3] FERGUSON P, SENIE D. Network Ingress Filtering: Defeating De-nial of Service Attacks Which Employ IP Source Address Spoofing. RFC 2827, 2000.

[4] SUN H J, FANG B X, ZHANG H L. DDoS attacks detection based on link character. Journal on Communications, 2007, 28(2):88-93.

[5] GAO Z, ANSARI N. Tracing cyber attacks from the practical perspec-tive. IEEE Communication Magazine, 2005, 43(3):123-131.

[6] KIM Y, LAU W, CHUAH M, et al. Packetscore: a statis-

tics-based packet filtering scheme against distributed denial-of-service attacks. IEEE Transactions on Dependable and Secure Computing, 2006, 3(2): 141-155.

[7] JIN C, WANG H, SHIN K G. Hop-count filtering: an effective defense against spoofed DDoS traffic. IEEE/ACM Transactions on Net-working, 2007, 15(1):40-53.

[8] YAAR A, PERRIG A, SONG D. Pi: a path identification mechanism to defend against DDoS attacks. Proc of IEEE Symposium on Se-curity and Privacy. Oakland, CA, USA, 2003. 93-107.

[9] YAAR A, PERRIG A, SONG D. StackPi: new packet marking and filtering mechanisms for DDoS and IP spoofing defense. IEEE Journal on Selected Areas in Communications, 2006, 24(10): 1853-1863.

[10] [KIM Y, JO J, MERAT F, et al. Defeating distributed denial-of-service attack with deterministic bit marking. Proc of IEEE Globecom, San Francisco. CA, USA, 2003. 1363-1367.

[11] LEE F, SHIEH S. Defending against spoofed DDoS attacks with path fingerprint. Computers & Security, 2005, 24(7):571-586.

[12] LEE G,LIM H, HONG M, et al. A dynamic path identification mechanism to defend against DDoS attacks. Proc of ICOIN. Jeju Island, Korea, 2005. 806-813.

[13] CAIDA Skitter [EB/OL]. http://www.caida.org, 2007.

[14] The Swiss Education and Research Network. Default TTL values in TCP/IP [EB/OL]. http://secfr.nerim.net/docs/fingerprint/en/ ttl_ de-fault.html, 2000.

[15] LIU J, LEE Z, CHUNG Y. Dynamic probabilistic packet marking for efficient IP traceback. Computer Networks, 2007, 51(3): 866-882.

