社団法人 情報処理学会　研究報告
IPSJ SIG Technical Report

2004－MBL－31　(15)
2004－ＩＴＳ－19　(15)
2004／11／12

ネットワークサービス不能攻撃に対する追跡手法の実装と評価

河村　栄寿[1)]・岡崎直宣[2)]

ネットワークのブロードバンド化が進み，常時格安の料金でインターネットに接続できる環境が整ってきた．しかし，それと同時にインターネットの安全性を脅かす様々な脅威，中でもサービス不能(Denial of Service)攻撃が問題となっている．通常，この DoS 攻撃で送信されるパケットの送信元アドレスは偽装されており，攻撃者の特定は困難である．本稿では，DoS 攻撃パケットの送信元を追跡する技術の一つであるマーキング方式について，そこで課題となる攻撃経路を特定する際の追跡のために必要となる情報量を削減し，また，その情報から攻撃者を特定するまでにかかる時間を改善する手法を提案し，数式モデルと実験による評価を行った．

An experimental evaluation of trace-back techniques of Denial of Service attacks

Shigehisa KAWAMURA[1)], Naonobu OKAZAKI[2)]

Abstract

Recently, frequency of Denial of Service attacks are increased, and it is difficult to trace packets with incorrect, or "spoofed", source address. Savage et al. proposed a method to trace flooding attacks by "marking" packets. This method, however, has some problems including the time consuming for gathering packets to re-configure the attacking paths.

We proposed a method to solve this problem by extending marking area in each packet.

In this paper, we will implement the both algorithms to estimate the usability of them. From the results of some experiments, we will conclude the effectiveness of the extended method.

1. はじめに

近年，多くのユーザがインターネットに常時接続できるようになった．しかし，インターネットの安全性を脅かす攻撃も頻発するようになり，なかでも特定のホストに一斉にパケットを送信させる DoS（Denial of Service）攻撃やその分散型である DDoS（Distributed DoS）攻撃が深刻な被害を与えている．これらの攻撃への対策は，ネットワークセキュリティにおける最重要課題の一つであるが，DoS や DDoS 攻撃を阻止する有効な手段はいまだに確立されていない．一般的な対策は，攻撃者に最も近いルータで帯域制限やパケットフィルタリングを行う方法があるが，IP プロトコルでは送信者が IP アドレスを自由に決めることができるので，多くの場合攻撃者は送信元アドレスを偽装（IP spoofing）する．このため，攻撃対象は攻撃パケットからは攻撃者を特定することはできない．この問題点の解決策として，ルータを逆探索し DoS や DDoS 攻撃の発信源を追跡するための技術で

1) 宮崎大学大学院工学研究科・Graduate School of Engineering, University of Miyazaki
2) 宮崎大学工学部・Faculty of Engineering, University of Miyazaki

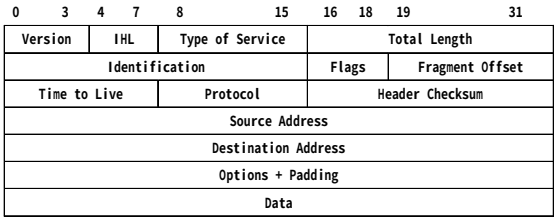


図1 IPv4 のパケット構造

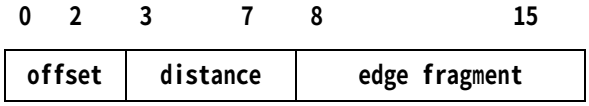


図2 分割後の識別子フィールド

ある IP トレースバックという手法がある．IP トレースバックにはいくつかの手法が提案されている [1]．その手法の一つとして Savage らの方法[2]（以下マーキング法）がある．マーキング法は流れているパケットそのものに追跡のための情報を付与し攻撃対象へと送る．マーキングされたパケットを受け取った攻撃対象は，パケットに付与された情報を元に攻撃経路を再構築する．しかし，複数の攻撃経路を再構築する際には，再構築するために十分な数のパケットを収集するための時間がかかるという課題(以下パケット収集問題)と，再構築に必要な計算量が莫大になるという課題(以下計算量問題)がある．これらの課題を改善した技術としてマーキング法を拡張した手法(以下拡張マーキング法)が提案されている [3),4)]．拡張マーキング法では，マーキングするフィールドを拡張し，さらにマーキングされたパケットかどうかを識別するフィールドも設けるため，マーキング法の課題を改善することが期待できる．しかしながら，マーキング法，および拡張マーキング法ともに実装された例がないため，その実用性は十分に検討されていないのが現状である．

本稿では，マーキング法と拡張マーキング法の数式モデルによる比較と，両手法のアルゴリズムを実装した実験による比較を行った．以下，本稿では 2.でマーキング法と拡張マーキング法について述べる．さらに，3.で実験と評価について述べ，4.で考察を与える．5.はまとめである．

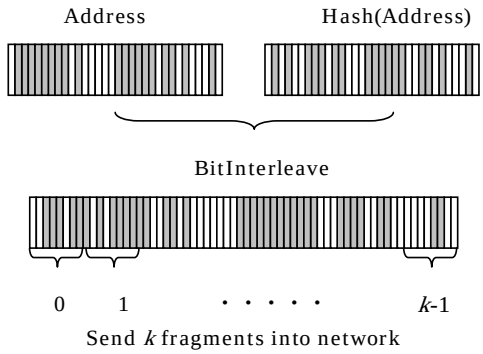


図3 ビットインターリーブ

2．マーキング法と拡張マーキング法

ここでは，マーキング法と拡張マーキング法について述べる．両手法とも，マーキングをする手順と再構築をする手順の二つのフェーズから構成される．

以下 2.1 と 2.2 でマーキング法におけるマーキング手順と再構築手順についてそれぞれ述べ，2.3 で拡張マーキング法について述べる．

2．1 マーキングの手順

図１に IPv4 のパケット構造を示す．IP ヘッダのうち識別子（Identification）フィールドを追跡のための情報を格納するために利用する．

マーキング法では，図２のように識別子フィールドを３つに分割し，追跡のための情報を格納する．分割後の各フィールドの役割と追跡情報を格納する手順を以下に示す．

【マーキング手順】

■ 準備段階

(1)ルータが起動時に自分自身の IP アドレスのハッシュ値を計算しておく．

(2) IP アドレスとハッシュ値をビットインターリーブする（図３）．このとき，IP アドレスは奇数ビット，ハッシュ値は偶数ビットに挿入するものとする．

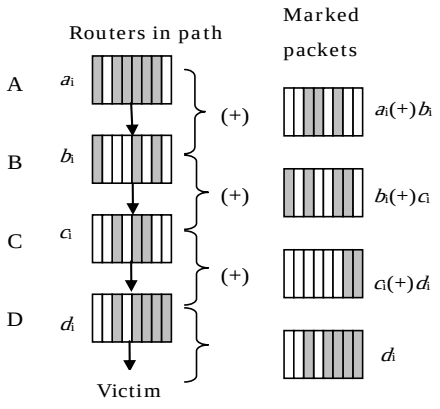
(3) ビットインターリーブした 64 ビットの値（以下，インターリーブ値）を k 個に分割する．マーキング法では k=8 である（k=2^i, i=1, … , 5）．

■ 初期マーキング

(1) ルータは自分自身を通過するパケットのうち確率 p でマーキングするかどうか選択する．

<a>マーキングされたパケットについて以下を行う．

・インターリーブ値を k 個に分割した値（以下，フ



(+) : XOR

x_i : i th fragment of interleaved value x

i = (0,・・・,k-1)

x = a, b, c, d

A,B,C,D : router

図4 攻撃対象へ届くマーキング情報

ラグメント）の一つをランダムに選び，パケットの識別子フィールドのエッジフラグメント（edge fragment）フィールドに書く．このランダムに選択された値が，インターリーブ値のどの部分にあったかを示す番号をオフセット（offset）フィールドに書き，マーキングしたルータから攻撃対象まで何個のルータを通過したかを示すディスタンス（distance）フィールドの値を 0 として下流のルータに送る [b].

<b>マーキングされなかったパケットについては以下の終端マーキングまたは転送ルータによる処理を行う．

■ 終端マーキング

(1) ディスタンスフィールドが 0 であるパケットを受け取ったルータは，1 ホップ前のルータによりマーキングされたと判断して，自分自身のインターリーブ値のフラグメントのうち，受け取ったパケットのオフセットフィールドの番号と同じエッジフィールドの値と XOR（排他的論理和）を求める．この値をエッジ ID と呼ぶ．エッジ ID をエッジフラグメントフィールドに上書きする，

(2) ディスタンスフィールドを一つ増やす．

■ 転送ルータによる処理

(1) 攻撃対象までの間でこのパケットが通過するルータではディスタンスフィールドの値を 1 ずつ増やしていく．

□

[b] distance フィールドが 5 ビットであるので 31 ホップまで記録できる．これはほとんどのパケットがインターネットを流れる際，31 ホップのパスより短いことを前提としている [2].

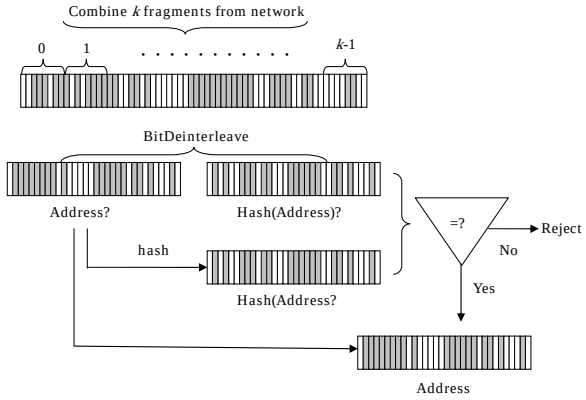


図5 復元手順

図４は，攻撃対象の直前に 4 つのルータ(A,B,C,D)があった場合に，それらによってマーキングされる情報を示している．A によってフラグメント a_i がマーキングされたパケットは B によってそのマーキングが $a_i(+)b_i$ の値に上書きされる．C 以降のルータで上書きされなかった場合にはそのパケットが攻撃対象へ送られる．同様に B によってフラグメント b_i がマーキングされたパケットが C によってそのマーキングが $b_i(+)c_i$ に上書きされ，攻撃対象へとそのパケットが送られる．攻撃対象の直前の D がマーキングをした場合については，そのままフラグメント d_i がマーキングされたパケットが攻撃対象へと送られる．

2．2 攻撃経路の再構築

攻撃対象に集められたマーキングパケットを再構築する過程において，攻撃経路が一直線である場合には，各ルータ同士のマーキングパケットのディスタンスフィールド値が重複することは無いため，攻撃経路は一意に決まる．しかし複数の攻撃者が存在し各攻撃経路が部分的にでも独立していると，他のルータが送ったマーキングパケットと同じディスタンスフィールド値とオフセットフィールド値を持つ場合があり得る．そのため，同じディスタンスフィールド値の任意のフラグメントを結合してインターリーブ値を復元し,その奇数ビットに割り当てられた IP アドレスから求められたハッシュ値が偶数ビットに割り当てられたハッシュ値と同じ値になれば復元されたインターリーブ値は正当なものであると判断する．同じ値にならなかった

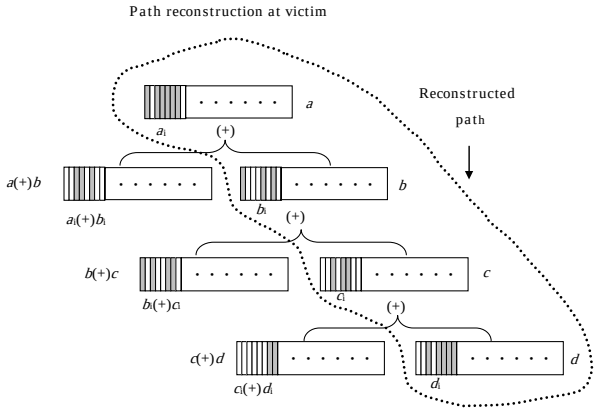


図6　攻撃経路の再構築手順

場合は，フラグメントの違う組み合わせを試し，正当なインターリーブ値と認められるまで同じことを繰り返す必要がある(このようにして正当なインターリーブ値を導出する手順を以下では復元手順と呼ぶ)．ハッシュは復元されたインターリーブ値が正当か識別する役割を持つ．図５は復元手順を示している．以下に，攻撃経路を再構築する手順を述べる．

[再構築手順]

（1）攻撃対象はディスタンスフィールド値が 0 のマーキングパケットを集め，復元手順によりディスタンスフィールドの値が 0 のインターリーブ値を導出する．

（2）ディスタンス i のインターリーブ値が求められているとする．

ディスタンスフィールドの値が i+1 のエッジ ID を集め，オフセットフィールドの値が重ならないすべての組み合わせを試し，それぞれディスタンスフィールドの値が i のインターリーブ値と XOR を求める．これらの値から復元手順により，ディスタンスフィールドの値が i+1 のインターリーブが求められる．

(3)　攻撃者に近いルータのインターリーブ値を導き出すまで，(2)の手順を繰り返す

□

図６に図４で示したマーキングパケットから攻撃経路を再構築する手順を示す．まず，攻撃対象に最も近いルータ(D)によってマーキングされたディスタンスフィールドの値が 0 であるマーキングパケットから，対応するインターリーブ値が復元手順により求められる．以降，攻撃者に最も近いルータ(A)のインターリーブ値を導き出すまで同様の処理が繰り返される．

このとき，同じディスタンスフィールドを持つ組み合わせの異なるインターリーブ値があった場合は図 5 の復元手順によって正当なインターリーブ値のみを組み合わせの対象にする．

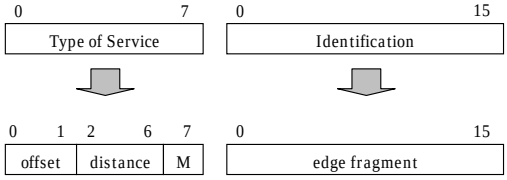


図7　サービスタイプフィールドと識別子フィールドへの割り当て

2．3　拡張マーキング法

拡張マーキング法はマーキング法におけるパケット収集問題と計算量問題を解決するために提案された[3),4)]．ここで計算量問題とは，分割されたマーキングパケットを復元する過程において，マーキングされていないパケットの組み合わせもすべて調べなければならないため莫大な計算量を要してしまうということを指す．

拡張マーキング法は，IP ヘッダのサービスタイプ（Type Of Service）フィールドもマーキング情報を格納するために用いる．図 7 は２つのフィールドのマーキング情報の割り当てを示している．識別子フィールドの全１６ビットをエッジフラグメントとするためフラグメントの数は k=4 となる．そして，サービスタイプフィールドには M ビットを設ける．ここで M ビットはマーキングパケットかそれ以外のパケットを識別す役割を持つ．拡張マーキング法においてもマーキング法と同様に，マーキング手順と再構築手順からなる．ただし，M ビットを設けたことでマーキングパケットのみを計算の対象にすることができるためフラグメントの組み合わせ数を減らすことができる．

拡張マーキング法のアルゴリズムを図８および図９に示す．

図８において，(*1)はマーキングされたパケットがそれ以外のパケットかを識別する役割を持つ．そして，ディスタンスフィールドが０であるパケットを受け取ったルータがマーキングを行う際（同(*2)）に，オフセットフィールドの値を参照してエッジ ID をエッジフラグメントフィールドに書いた後（同(*3)）に，このビットに M=1 の値を設定（同(*4)）する．

図９はインターリーブ値の復元手順を表しており，

Marking procedure at router R:

```
let R' = Bitintereave (R, Hash (R))  ;
let k be the number of non-over lapping fragment in R' ;
for each packet w
  let x be a random number from [0..1)  ;
  let M be a number from [0,1] ;                (*1)
  if x < p then
    let n be random integer from [0..k-1] ;
    let f be the fragment of R' at offset n ;
    write f into w.frag ;
    write 0 into w.distance ;
    write n into w.offset ;
    write 1 into w.M ;
  else
    if w.distance = 0 then                      (*2)
      let f be the fragment of R' at offset w.offset ; (*3)
      write f (+) w.frag into w.frag ;          (*4)
      increment w.distance ;
```

図8 マーキング手順のアルゴリズム

集められた攻撃パケットから M=1 であるパケットのみを対象としてフラグメントの組み合わせを試す（同図(*5)）ようにする.

3. 評価

本章では，パケット収集問題と計算量問題に着目し，マーキング法と拡張マーキング法の比較を行う．パケット収集問題に関しては，数式モデルによる評価と実験による評価を行い，計算量問題に関しては数式モデルによる評価を行う.

3.1 パケット収集問題

以下 3.1.1 で数式モデルによるパケット収集問題の評価を行い，3.1.2 で実験によるパケット収集問題の評価を行う.

3.1.1 数式モデルによる評価

全てのルータが確率 p でマーキングをするものとする．このとき，攻撃対象から距離 d のルータによりマーキングされ，かつ，そのパケットが下流のルータによって上書きされないで攻撃対象に届く確率は $p(1-p)^{d-1}$ である.

攻撃対象から d ホップ離れたルータから一つのマーキングパケットを受け取るために観察しなければならない攻撃パケット数の期待値 Q は $Q = 1/p(1-p)^{d-1}$ である．d 個のうち，少なくとも一つのルータから攻撃対象にマーキングパケットが送られてくる確率は $dp(1-p)^{d-1}$ となる．簡単化のため，マーキングパケットを送る確率をすべてのルータについて同じであると想定すると，d 個のうち少なくとも一つのルータから一つのマーキングパケットを受け取るまでに観察しなければ

Path reconstruction procedure at victim v:

```
let FlagTbl be a table of tuples  (frag,offset,distance,M)  ;
let G be a tree with root v ;
let edges in G be tuples  (start, end, distance)  ;
let maxd : = 0 ;
let last : = v ;
for each packet w from attacker
  FlagTbl.Insert (w.frag,w.offset,w.distance,w.M)  ;
  if w.M = 0
    continue;                                   (*5)
  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
      if Hash (OddBits (z))  = EvenBits (z)  then
      insert edge  (last, OddBits (z) ,d)  into G ;
      last := OddBits (z)  ;
      lastz := z ;
  for d:=1 to maxd
  z := z (+) lastz ;
      if Hash (OddBits (z))  = EvenBits (z)  then
      insert edge  (last, OddBits (z) ,d)  into G ;
      last := OddBits (z)  ;
      lastz := z ;
remove any edge  (x,y,d)  with d ≠ distance from x to v in G ;
extra path  (Ri..Rj)  by enumerating acyclic paths in G ;
```

図9 再構築のアルゴリズム

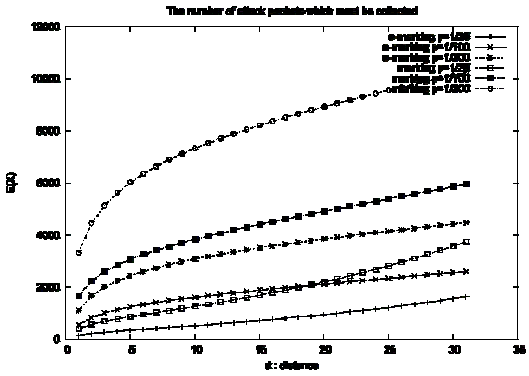


図 10 集めなければならないパケットの期待値

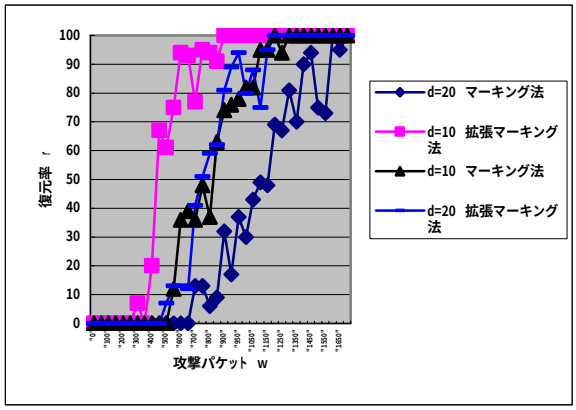


図 11 実験結果

ならない攻撃パケットの数の期待値 E_p は $E_p < 1/dp(1-p)^{d-1}$ となる．さらに，攻撃対象に集められたマーキングパケットの中から何回の試行をすれば d 種類分のマーキングパケットを集めることができるかという数の期待値 E_c は，k 個に分割されていることを考慮すると，$E_c = kd(\ln(kd))$で与えられる．よって攻撃経路を再構築するために必要な攻撃パケットの期待値 E_x は，

$$E_x = E_p \times E_c < kdQ(\ln(kd))/d \qquad (1)$$

となる．式(1)は，マーキング法，拡張マーキングともに成り立つ．図 10 は各ディスタンスフィールドの値に応じて，攻撃経路を再構築するために必要なパケットの期待値がどのように変化していくかを示している．ここで p は 1/25，1/100，1/200 とする．同図より p = 1/25 においては，ディスタンスフィールドの値が増加しても，拡張マーキング法の方が常に E(X)が少ないことがわかる．また P = 1/100，P = 1/200 においても同様である．

3．1．2 実験による評価

プログラミングは攻撃経路が一直線の場合を想定し，RedHat Linux9 上で C 言語で行い，約 500 行の規模になった．実験方法は，横軸に攻撃パケット数 w，縦軸に復元率 r をとり，ルータの数が 10 個の場合（d=10），20 個の場合（d=20）について実験を行い，拡張マーキング法，マーキング法ともに攻撃パケット数の増加にともないどのように復元率が変化していくのかを調べた．ここで，復元率 r は攻撃パケット数が w 個の試行を n 回実行したとき，s 回攻撃経路を完全に再構築できた場合に復元率 $r=s/n$ と定義する．本実験では，ルータのマーキングする確率 p は 1/25 としている．

図 11 は，d=10 および d=20 の場合のマーキング法，拡張マーキング法の攻撃パケット数に対する復元率を表す．同図より，d=10 および d=20 の場合，マーキング法より拡張マーキング法のほうが少ない攻撃パケット数で再構築することができることがわかる． さらに，両手法とも d=10 の場合のほうが d=20 の場合より少ない攻撃パケット数で再構築できることがわかる．

3．2 計算量問題

攻撃経路を再構築する際の計算量はフラグメントの組み合わせ数に大きく依存するため，ここでは組み合わせの数を計算量として評価する．以降で，拡張マーキング法におけるフラグメントの組み合わせの数を検証し，マーキング法の場合と比較する．

一般に識別子フィールドの値は実装に依存して不定である．そのため，マーキング法では攻撃対象から d ホップ離れたルータが実際にマークしたパケットかどうかを Q 個のうちから識別できない．したがって，フラグメントを結合してハッシュを計算する際に，全ての組み合わせを試す必要がある．例えば，d ホップ離れたルータが m 個あるとすると，組み合わせの候補となるフラグメントの種類が Qm あることになる．したがって，これらの k 個の組み合わせは$(Qm)^k$ 通りである．ただし，ここでは一つのフラグメントのビット数は 64/k なので Qm の上限は $2^{64/k}$ である．一方，拡張マーキング法では，同様に d ホップ離れたルータのマーキングパケットを得るために必要なパケットは Q 個であるが，そのうち M ビットの値により，マーキングパケットかその他のパケットかを識別できる．したがって，攻撃対象から d の距離に m 個のルータがある場合のフラグメントの組み合わせは m^k 通りで

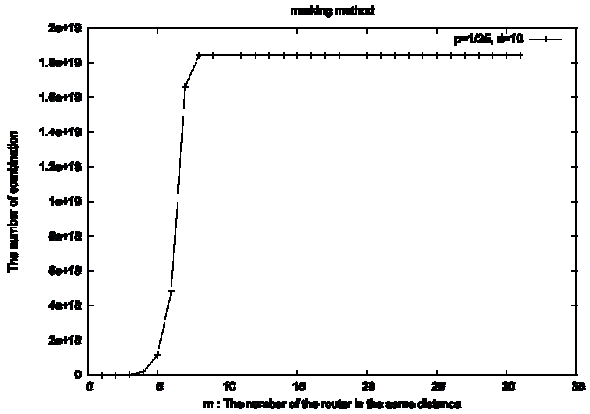


図 12 マーキング法の組み合わせ数

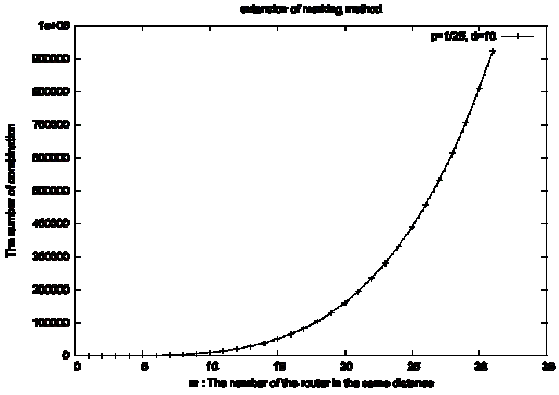


図 13 拡張マーキング法の組み合わせ数

ある．例えば，p = 1/25，d = 10，m = 10 の場合，マーキング法では Qm の値は上限の $2^{64/8}$ = 256 となり，組み合わせの数は$(256)^8$　　1.8×10^{19} 通りある．一方，拡張マーキング法では m^4 = 10^4 通りの組み合わせである．したがって，拡張マーキング法はパケットを集めた後，攻撃経路の再計算がマーキング法に比べ少なく十分実用的な値になると考える． 図 12 にマーキング法の組み合わせ数の変化，図 13 に拡張マーキング法の組み合わせの変化をそれぞれ示す．ここで p=1/25，d=10,とし，m は 1 から 32 まで変化するものとする．同図よりマーキング法では d=10，m=8 の場合に組み合わせの数が上限の 1.8×10^{19} に達し，計算量が爆発的に増えてしまうことがわかる．一方，拡張マーキング法ではｄ=10，m=8 の場合，組み合わせ数が 4096 とマーキング法に比べ，大幅に少なく，また m の数が増加しても計算量が大幅に増加しないことが分かる．

4．考察

4．1 パケット収集問題の評価に関する考察

3.1.1 で述べた数式モデルによる評価より，拡張マーキング法ではマーキング法に比べ攻撃経路を再構築するために必要な期待値が常に少ないことがわかった．このことから，拡張マーキング法ではより攻撃時間の短い DoS 攻撃に対しても追跡が可能となる．また図 10 より拡張マーキング法ではマーキング法と比較して p の値を低めに設定しても同様の効果が得られる．一般に DDoS 攻撃では数秒で数千から数万のパケットが攻撃対象へと届くため [2]，p の値を低めに設定できる拡張マーキング法ではよりルータに負荷をかけずに実行できると考えられる．

3.1.1 で述べた数式モデルによる評価において攻撃経路を再構築するために必要なパケットの期待値は，ルータの数が 10 個のとき，拡張マーキング法で 550 個以下，マーキング法で 1300 個以下であり，ルータの数が 20 個のときは，拡張マーキング法で 950 個以下，マーキング法では 2200 個以下であった．これに対して，3.1.2 で示した実験による評価においても数式モデルの場合とほぼ同じ数のパケットを収集すればパスを再構築できることがわかった．

これらにより，攻撃経路を再構築するために必要なパケットの期待値は拡張マーキング法のほうがマーキング法に比べより少なく，有効であると考えられる．

4．2 計算量問題の評価に関する考察

3.2 で述べたようにマーキング法は攻撃経路が増えるにつれて計算量が爆発的に増えてしまう．一方，拡張マーキング法では攻撃経路が複数ある場合でも，組み合わせの計算量はマーキング法に比べ緩やかに増加するので，再構築処理を実行するコンピュータへの負荷が低く，攻撃者の追跡時間を短縮できると考えられる．

5．まとめ

本稿では，IP トレースバック技術のうち，パケット収集問題と計算量問題という二つの観点から，マーキング法と，拡張マーキング法の数式モデルと実装による比較を行った．パケット収集問題に関しては，数

式モデル，実験結果ともに拡張マーキング法の方が集めなければならないパケットの数が少ないことがわかった．また，計算量問題で関しても同様に拡張マーキング法の方が組み合わせの数による計算量が大幅に少ないことがわかった．以上のことにより，拡張マーキング法の方がマーキング法に比べ追跡にかかる時間が短く，またルータやコンピュータの負荷を抑えながら実行できると考えられる．

今後の課題としては，攻撃経路が複数の場合における収集しなければならないパケット数の期待値と組み合わせの数による計算量について実験による評価を行なう必要がある．また，ルータの数，マーキングする確率等の様々なパラメータを変化させた場合の評価を今後行っていきたい．

参考文献

1）門林 雄基, 大江 将史, "IP トレースバック技術," IPSJ Magazine, Vol.42, No.12, pp.1175-1180, 2000.

2） S. Savage, D. Wetherall, A. Karlin, T. Anderson, "Practical Network Support for IP Traceback," Proc. SIGCOMM '00, pp.295-306, 2000.

3）河村 栄寿, 岡崎 直宣, 中谷 直司, 厚井 裕司, 朴 美娘, "ネットワークサービス不能攻撃の追跡手法に関する一検討," 情報処理学会火の国シンポジウム 2003 論文集, pp.168-175, 2003.

4）岡崎 直宣, 河村 栄寿, 朴 美娘, "サービス不能攻撃の経路追跡手法の効率化に関する検討," 情報処理学会論文誌, Vol.44, No.12, pp.3197-3201, 2003.