社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE.

トラヒックマトリクス推定を用いた攻撃元特定手法

大下　裕一†　　阿多　信吾††　　村田　正幸†

† 大阪大学 大学院情報科学研究科
†† 大阪市立大学 大学院工学研究科
E-mail: †{y-ohsita,murata}@ist.osaka-u.ac.jp, ††ata@info.eng.osaka-cu.ac.jp

あらまし　近年，公開サーバに対する分散サービス拒否 (DDoS) 攻撃はますます深刻を増しており，早急な対策が望まれる．DDoS 攻撃に対する根本的な防御として，攻撃元を特定たのち，エッジルータにおいてフィルタリング等を行うことが有効である．しかし，既存の攻撃元特定手法では，ルータの大規模な置き換えが必要となるものや，攻撃元と正常な通信相手の区別が難しい等の問題があり，広く利用されていないのが現状である．そこで本稿では，既存のルータを用いて実現可能であり，トラヒック増加の原因となる送信元を検出する新たな攻撃元検出手法を提案する．提案手法では，SNMP などによる定期的なルータからトラヒック観測情報を収集し，各送信元宛先間のトラヒック変化量を推定する．そして，その推定結果をもとに，トラヒック増加の原因となる攻撃元の特定を行う．本稿では，シミュレーションにより，提案手法が正確に攻撃元を特定することができることを確認する．

キーワード　DDoS 攻撃, トレースバック，トラヒックマトリクス，SNMP

Identification of Attack Nodes from Traffic Matrix Estimation

Yuichi OHSITA†, Shingo ATA††, and Masayuki MURATA†

† Graduate School of Information Science and Technology, Osaka University
†† Graduate School of Engeneering, Osaka City University
E-mail: †{y-ohsita,murata}@ist.osaka-u.ac.jp, ††ata@info.eng.osaka-cu.ac.jp

Abstract　Distributed denial-of-service attacks on public servers have recently become more serious. The most effective way to prevent the attack traffic is to identify attacking nodes and detach (or block) attack nodes at egress routers of them. Existing traceback mechanism, however, are not widely used today because of e.g., replacements of many routers to support traceback capability, or difficulties to distinguish attack and legitimate traffic. In this paper, we propose a new scheme to enable a traceback from a victim to attack nodes. More specifically, we identify egress routers to which attack nodes are connecting by estimating traffic matrix between arbitral source-destination edge pairs. By monitoring traffic variations obtained by the traffic matrix, we identify the edge routers forwarding attack traffic which have a sharp traffic increase to the victim. We also evaluate the effectiveness of our proposed scheme through simulation, and show that our method can identify attack sources accurately.

Key words　Distributed Denial of Service (DDoS), Traceback, Traffic matrix, Simple Network Management Protocol (SNMP)

1. はじめに

近年，インターネットの急速な発展により，ネットワークを介したさまざまなサービスが提供され，その利便性は増すばかりである．その一方で，悪意を持った第三者がサービスを提供する計算機に対して攻撃を行い，一般ユーザの利用を妨げるサービス拒否 (Denial of Service; DoS) 攻撃が深刻な問題となっている．

攻撃の技術も近年ますます高度化しており，各地に分散された複数の攻撃ノードが同時に同一サーバを攻撃する DDoS (Distributed DoS) 攻撃と呼ばれるものが主流になりつつある．DDoS 攻撃においては，攻撃者は計算機のもつ脆弱性を悪用して複数の端末に不正に侵入し，攻撃を実行するプログラムを実行可能状態で待機させる．そして，一斉に攻撃命令を送ることにより，複数端末から同時に攻撃が開始される．このため，各端末が生成する攻撃トラヒックがさほど影響を与えないもので

あっても，同時に攻撃を行う端末が非常に多ければ，サーバやネットワークへの影響は深刻になる．

DDoS 攻撃への根本的な対策は，攻撃ノードが接続されたエッジノードにおいて，攻撃ノードとの接続を切断，あるいは攻撃トラヒックをフィルタリングなどにより遮断することが有効である．しかし，現在の IP ネットワークでは，送信元 IP アドレスを容易に偽装できるため，実際の攻撃において，送信元アドレスの情報をもとに攻撃元を特定することは難しい．

このため，攻撃トラヒックを発生させた攻撃元ノードを特定する手法について検討がなされてきた．一般的に，あるノードに到着したパケットの送信元を逆探索して特定する行為を IP トレースバックと呼ぶ。代表的な IP トレースバックの手法として，一定数のパケットが通過するごとにルータが ICMP パケットを生成し宛先に送り，それをもとに攻撃元を特定する方法 [1], [2] や，パケットのヘッダに通過したルータの情報を確率的に書き込み，その情報から攻撃元を特定する手法 [3]〜[5]，各ルータにおいて，転送したパケットのハッシュ値を保存し，その情報を遡ることで攻撃元を特定する手法 [6], [7] などが提案されている．しかし，これらの手法はルータの置き換えを必要とする，あるいはパケットの送信元を特定することはできるものの，DoS 攻撃ではどのパケットが攻撃パケットなのかの判別することは難しいために，攻撃元と正常な通信相手の区別が難しい等の問題があり，広く利用されていないのが現状である．

DoS 攻撃では，攻撃者は被害者に対し，大量のパケットを送ることでその通信を阻害する．そのため，トラヒック量の変化を観測することにより，攻撃の検出・攻撃元の特定が可能である．また，トラヒックの増加量に基づく攻撃元の特定は，トラヒックを急増させない正常な通信相手を攻撃元と誤検出することはないと考えられる. [8] では，各リンクのトラヒック量の観測結果を元にした攻撃元特定の手法を提案している．この手法では，すべてのリンクの到着レートを，時間的な変化の成分とそれ以外の成分に分け，時間的な変化成分以外の影響がもっとも大きい送信元・宛先間のトラヒックを攻撃トラヒックとみなす．この方法は攻撃元が単一であれば有効であるが，攻撃元が複数ある DDoS 攻撃の場合は，仮定しなければならない攻撃元の組み合わせ，攻撃元ごとの攻撃レートの組み合わせが多数存在し，攻撃の経路を確定することは難しい．

もし，各リンクのトラヒック量のみではなく，各送信元・宛先間のトラヒックを観測することができれば，トラヒック急増の原因となるトラヒックの送信元を特定することができることから，攻撃元の特定および対策が可能となる．

そこで，本稿では，送信元・宛先間のトラヒック量の変化を観測することによる攻撃元特定手法を提案する．送信元・宛先間のトラヒックの正確な観測は，ネットワークのエッジルータにおいて送信元・宛先ごとのフロー統計情報の観測を必要とし，実現が容易ではないため，本稿では，送信元・宛先間のトラヒックの変化量は，送信元宛先間のトラヒック量 (トラヒックマトリクス) をリンクのトラヒック量から推定する手法 [9] を応用することにより推定を行う．

提案手法では，リンクのトラヒック量は Simple Network Management Protocol (SNMP) などを用いてルータから収集することが可能である．このため，既存のネットワークにも容易に適用可能である．また，トラヒックの増加をもとに攻撃元を判断するため，より高い精度で正常な通信相手を攻撃元と誤検出することなく，攻撃元を特定することができる．

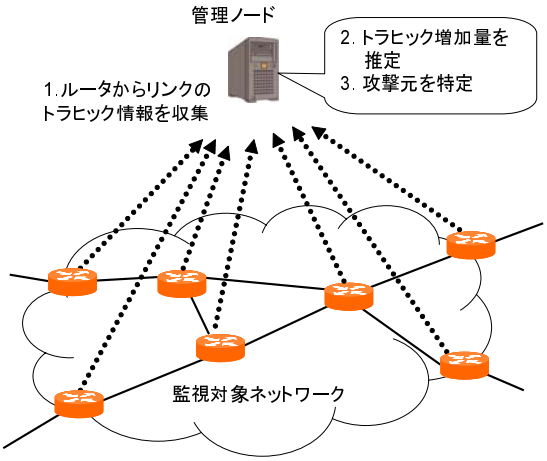

図 1　提案手法の概要

以降，まず，2. で攻撃元特定のためのトラヒックマトリクス推定手法を提案する。次に 3. で推定したトラヒックマトリクスを用いた攻撃元特定手法について述べる。4. で，提案手法の評価を行う．最後に，5. でまとめと今後の課題について述べる．

2. 攻撃元特定のためのトラヒックマトリクス推定手法

提案手法では，送信元・宛先間のトラヒック量を観測することにより，大量のパケットを被害者宛に送っている攻撃元の特定を行う．しかし，ネットワークのエッジルータにおいて，送信元・宛先ごとのフロー統計情報を観測することは容易ではない．そこで，送信元・宛先間のトラヒックの変化を，各リンクのトラヒック量の変化から推定することを考える．

提案方式の概要を図 1 に示す．提案手法では，ネットワークの監視・攻撃元特定を行う管理ノードを導入する．以降，このノードを管理ノードと呼ぶ．攻撃元特定ノードは以下の手順を定期的に繰り返すことで，攻撃元の特定を行う．

（1） 監視しているネットワーク内の各ルータからリンクのトラヒック量の情報を得る

（2） 提案するトラヒックマトリクス推定手法により，送信元・宛先間のトラヒック増加量を推定する

（3） 推定されたトラヒック増加量をもとに攻撃元を特定する

以下，2. でトラヒック増加量の推定方法，3. で推定されたトラヒック増加量をもとにした攻撃元特定方法の詳細について述べる．

2.1 Gravity model [9] による推定

X をリンクのトラヒック量，A をネットワークのルーティング情報をあらわす行列とすると，トラヒックマトリクス T との間には以下の関係が成り立つ．

$$X = AT \tag{1}$$

式 (1) のうち，X は SNMP を用いて取得可能であり，A は OSPF の Link State Advertisement (LSA) などのルーティングメッセージの観測 [10] や，それができない場合であっても，ルーティングの設定をもとにしたシミュレーション [11] により取得可能である．そこで，式 (1) が成り立つ T を求めることができれば，トラヒックマトリクスを推定することができる．しかし，この関係が成り立つ T は多数存在するため，トラヒックの特性を仮定する必要がある．[9] では，gravity model を用いてトラヒックマトリクスの推定を行っている．gravity model では，送信元・宛先間のトラヒックの流量は，その入り口となるリンクのトラヒック量，出口となるリンクのトラヒック量に比例すると仮定している．そのため，l_i^{in} を入り口 i で観測されたトラヒック量，l_j^{out} を出口 j で観測されたトラヒック量とすると，i から j へのトラヒック量 $t_{i,j}$ は式 (2) のように推定可能である．

$$t_{i,j} = l_i^{\text{in}} \frac{l_j^{\text{out}}}{\sum_k l_k^{\text{out}}} \tag{2}$$

ここで，i から j へ攻撃トラヒックとして t_{attack} のレートが加わった場合を考える．入り口 i で観測された正常なトラヒック量を l_i^{in}，出口 j で観測された正常なトラヒック量を l_j^{out} とあらわし，i から j への正常なトラヒック量 $t_{i,j}$ は式 (2) を用いて正確に推定できるものと仮定する．この場合，i から j への攻撃を加えたトラヒック量 $t'_{i,j}$ は次のように推定される．

$$t'_{i,j} = (l_i^{\text{in}} + t_{\text{attack}}) \frac{l_j^{out} + t_{\text{attack}}}{\sum_k l_k^{\text{out}} + t_{\text{attack}}} \tag{3}$$

そのため，i から j へのトラヒックの増加量は以下のように推定可能である．

$$t'_{i,j} - t_{i,j} = \frac{t_{\text{attack}}^2 + t_{\text{attack}}(l_i^{in} + l_j^{out})}{\sum_k l_k^{\text{out}} + t_{\text{attack}}} \tag{4}$$

たとえば，ネットワーク全体の総トラヒック量が 20 GByte，i, j で観測される正常なトラヒックがいずれも 2 GByte，攻撃トラヒックが 1 GByte であった場合は，i から j へのトラヒック増加量は 0.23 GByte と小さく推定されることになる．

このように，gravity model を用いたトラヒックマトリクス推定では，攻撃による出口のトラヒックの増加は，送信元のトラヒック量にしたがって分配されてしまい，DDoS 攻撃のように多数の攻撃ノードから特定のノードへのトラヒックが集中するような場合，トラヒック増加の原因となる送信元を把握することが難しい．そこで，トラヒックの増加分を正確に把握するために，トラヒック増加量に注目した推定手法を提案する．

2.2 トラヒック増加量にもとづく推定

提案する攻撃元特定手法は，重要となる情報は，トラヒックの増加量である．そこで，トラヒックの増加を正確に把握するために，各リンクごとのトラヒックの増分をもとに，送信元・宛先間のトラヒック増加量の推定を行う．

時間をスロット化して考え，n 番目の時間に観測されたリンクのトラヒック量を要素とする行列を L_n とし，リンクの正常時のトラヒック量を要素とする行列 $\bar{L}_n$ とする．また，各リンクのトラヒックの正常時と比べた増加量を要素とする行列 G_n を定義する．

$$G_n = L_n - \bar{L}_n \tag{5}$$

そして，観測された G_n をもとに，式 (2) を負の値が存在した場合でも適用可能なように拡張した式 (6) を用いて，観測時刻 n における i から j へのトラヒックの増加量 $f'_{i,j,n}$ を推定する．ここで，$g_{i,n}^{\text{in}}$ は，G_n の要素のうち入り口リンク i におけるトラヒックの増加量，$g_{j,n}^{\text{out}}$ は出口 j におけるトラヒックの増加量とする．

$$f'_{i,j,n} = \begin{cases} g_{i,n}^{\text{in}} \dfrac{g_{j,n}^{\text{out}}}{\sum_{\{k:(g_{k,n}^{\text{out}}>0)\}} g_{k,n}^{\text{out}}} & (g_{i,n}^{\text{in}} > 0, g_{j,n}^{\text{out}} > 0) \\ -\left| g_{i,n}^{\text{in}} \dfrac{g_{j,n}^{\text{out}}}{\sum_{\{k:(g_{k,n}^{\text{out}}<0)\}} g_{k,n}^{\text{out}}} \right| & (g_{i,n}^{\text{in}} < 0, g_{j,n}^{\text{out}} < 0) \\ 0 & (\text{others}) \end{cases} \tag{6}$$

しかし，$f'_{i,j,n}$ は，出入り口以外のリンクの観測結果は反映されていない．そのため，途中のリンクのトラヒック情報を $f'_{i,j,n}$ に反映させることで，より送信元宛先間のトラヒック増加量の推定精度が高くなると考えられる．そこで，トラヒック増加量 F_n についても，式 (1) と同様，式 (7) の関係が成立することを利用し，式 (7) が成り立ち，かつ，式 (6) での推定結果からもっとも近い値を最終的な推定結果とすることで，途中のリンクの情報を反映する．

$$G_n = AF_n \tag{7}$$

この途中のリンクの観測結果を反映させた最終的なトラヒック増加量の推定結果は式 (8) を用いて得ることができる．ここで，A^{-1} は A の擬似逆行列とし，F'_n は $f'_{i,j,n}$ を要素とする行列とする．擬似逆行列は Scilab [12] の関数を用いて計算可能である．

$$F_n = F'_n + A^{-1}(G_n - AF'_n) \tag{8}$$

2.2.1 正常なトラヒック量の定め方

提案するトラヒック増加量の推定には必要な正常時のトラヒック量が必要である．正常時のトラヒック量 $\bar{L}_n$ としては，$\bar{L}_{n+1} = \alpha L_n + (1-\alpha)\bar{L}_n \quad (0 < \alpha < 1)$ のように過去の観測されたトラヒック量の重みつき平均をとることが考えられる．しかし，一時的なトラヒックの急増が $\bar{L}_n$ に反映されてしまうと，$\bar{L}_n$ が高い値となってしまう．$\bar{L}_n$ が高い値になると，トラヒックの急増が起きた場合であっても，$L_n - \bar{L}_n$ は高い値とならず，その後のトラヒック急増の検出に悪影響を与える．そこで，トラヒック急増の影響を除外して，$\bar{L}_n$ を定める必要がある．

トラヒック急増の影響を除外する方法として，観測されたトラヒックの増加値 $g_{i,n}$ に，過去の $g_{i,k}$ の平均，分散をもとに閾値を定め，閾値を超えていない箇所のみ，$\bar{L}_{n+1}$ に反映するという方法が考えられる．しかしながら，トラヒックマトリクスを推定するためには，すべてのリンクの観測結果が式 (7) が

成り立つ必要があるため，それぞれのリンクで独立に，閾値を定めることはできない。そこで，時刻 n の各フローごとのトラヒック増加量推定結果を用いて，$\bar{T}_{n+1}$ を定める。

まず，式 (9) のように，$\hat{f}_{i,j,n}$ を定める．ここで，F_n の i から j 宛てに対応する要素を $f_{i,j,n}$ とあらわし，$f_{i,j,n}$ の最近 N 個の平均，標準偏差をそれぞれ $\mu_{i,j}$，$\sigma_{i,j}$ とあらわす．β は，トラヒックの急増を判断する閾値を決めるパラメータである．

$$\hat{f}_{i,j,n} = \begin{cases} f_{i,j,n} & (f_{i,j,n} < \mu_{i,j} + \beta\sigma_{i,j}) \\ 0 & (\text{others}) \end{cases} \tag{9}$$

これにより，$\hat{f}_{i,j,n}$ はトラヒックが急増し，その増加量が閾値 $\mu_{i,j} + \beta\sigma_{i,j}$ を超えた場合は 0，それ以外の場合は推定されたトラヒック増加量 $f_{i,j,n}$ となる．

そして，以下のように，$\bar{L}_{n+1}$ を定める．ここで，$\hat{F}_n$ は $\hat{f}_{i,j,n}$ を要素とする行列とする．

$$\bar{L}_{n+1} = \alpha(\bar{L}_n + A\hat{F}_n) + (1-\alpha)\bar{L}_n \tag{10}$$

これにより，トラヒックが急増した箇所の影響をうけず，正常時のトラヒック量を定めることができる．

3. 攻撃元特定手法

DoS 攻撃が行われると，攻撃者から被害者宛のトラヒックが増加する．また，トラヒック増加量が大きい箇所ほど，ネットワークへの悪影響も大きい．そこで，被害者宛のトラヒックを増加させている送信元を攻撃元とみなすことができる．しかし，攻撃元が分散している場合，各攻撃元が生成する攻撃レートがさほど大きくない場合であっても，攻撃元の数が多ければ，攻撃の被害は大きくなる．そのため，到着レートの増加量に単純に閾値を定めるだけでは，攻撃元を特定できない場合がある．そこで，すべての攻撃元から被害者宛の出口への攻撃レートの総計を推定し，その推定された攻撃レートの総計と各送信元から被害者宛のトラヒック増加量をもとに，攻撃元の特定を行う．攻撃レートの総計をもとにすることにより，攻撃元が分散している場合であっても，適切に攻撃元を特定することが可能である．

攻撃開始時には，ネットワークにおける被害者側の出口リンクでは，攻撃レートの分，トラヒックが増加する．そのため，出口リンクの観測により，攻撃トラヒックの総レートの推測が可能である．しかし，正常なトラヒックにおいても，増加傾向・減少傾向のある時間帯があり，また，大きな増加・減少傾向以外にも正常なトラヒックも絶えず増減を繰り返している．そのため，攻撃レートを出口リンクでのトラヒック増加量と推定すると，正常なトラヒックが増加している時には，実際よりも高く攻撃レートを推定してしまい，その結果，攻撃元以外の送信元を攻撃元と判断してしまう．

そこで，式 (11) のようにリンクの観測されたトラヒック増加量 g^{out} をもとに，検出すべき攻撃レート $\tilde{g}^{\text{out}}$ を定める．ここで，g^{out} の最近 N 個の平均を μ^{out} とあらわすとする．また，γ は，正常なトラヒックの変動を表すパラメータである．これにより，μ^{out} で正常なトラヒックの増加・減少傾向を，γ でそれ

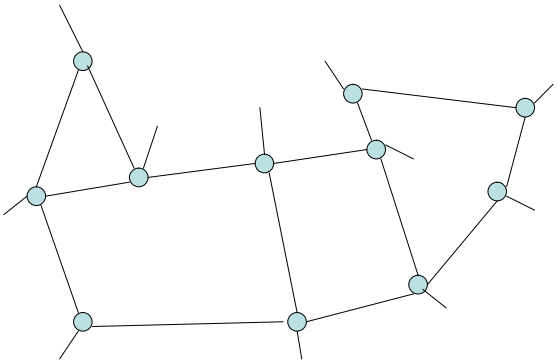

図 2　Abilene のバックボーントポロジ

以外の正常なトラヒックの変化の影響を吸収することができる．

$$\tilde{g}^{\text{out}} = g^{\text{out}} - \mu^{\text{out}} - \gamma \tag{11}$$

そして，式 (12) の条件が成り立つ場合，送信元 i を攻撃元とみなす．ここで，f_i は推定されたトラヒックマトリクス F_n の要素のうち送信元 i から被害者宛の推定されたトラヒック増加量をあらわすものとする。.

$$\sum_{(k:f_k>f_i)} f_k \leqq \tilde{g}^{\text{out}} \tag{12}$$

ここで，被害者宛の送信元ごとのトラヒック増加量の推定値 f_k を降順に並べ，j 番目までの和を $t^{\text{top}(j)}$ とあらわすとする．$t^{\text{top}(j)}$ は $\tilde{g}^{\text{out}}$ より小さく，$t^{\text{top}(j+1)}$ は $\tilde{g}^{\text{out}}$ より大きい場合，式 (12) より，上位から $j+1$ 番目の箇所まで攻撃元とみなされる．このとき，検出された攻撃元からの攻撃レートは $t^{\text{top}(j+1)}$ となり，$\tilde{g}^{\text{out}}$ より大きい．そのため，実際の攻撃レートの合計を t^{attack} とすると，未検出の攻撃元からの攻撃レートは $t^{\text{attack}} - \tilde{g}^{\text{out}}$ 以下となる．ここで，f^{normal} を正常なトラヒックの増加量とすると，$t^{\text{attack}} - \tilde{g}^{\text{out}}$ は $f^{\text{normal}} - \mu^{out} - \gamma$ と表すことができる．そのため，γ を適切に設定することにより，未検出の攻撃元からの攻撃レートを抑えることができる．

4. 提案手法の評価

本章では，シミュレーションにより，提案手法の評価を行う．シミュレーションで用いたネットワークトポロジを図 2 に示す．送信元・宛先間のトラヒック量は，大阪大学のゲートウェイでの観測結果をもとにした値を定めた．まず，大阪大学のゲートウェイでパケットの送信元アドレスの上位 16 ビットごとの到着レートを 60 秒間隔で取得した．そして，図 2 では送信元・宛先の組は 110 通りであため，観測され上位 16 ビットの送信元アドレスを 110 個のグループにわけ，そのグループのトラヒック量の合計を送信元・宛先ごとのトラヒック量とした。また，正常なトラヒック量を定める際のパラメータとして，トラヒックの時間変化を吸収可能な値である $\beta = 3$ とした．

また，本稿では，false-positive，false-negative という評価指標を用いて提案手法の評価を行う．本稿では，false-positive は攻撃元であるのに攻撃元として特定されなかった箇所とし，false-negative は攻撃元でないのに攻撃元と誤って判断された箇所とする．さらに，false-positive rate，false-negative rate

表 1　攻撃元数と false-positive，false-negative

攻撃元の数	false-negative の数 (false-negative rate)	false-positive の数 (false-positive rate)
1	0 (0.00)	2 (0.01)
2	0 (0.00)	0 (0.00)
3	0 (0.00)	3 (0.02)
4	3 (0.05)	4 (0.04)
5	12 (0.15)	4 (0.05)

を以下のように定義する．

$$\text{false-negative rate} = \frac{\text{false-negative の数}}{\text{攻撃元の数}}$$

$$\text{false-positive rate} = \frac{\text{false-positive の数}}{\text{攻撃元ではない送信元の数}}$$

4.1　攻撃元の数と false-positive，false-negative の関係

攻撃元の数と false-positive，false-negative の関係を調べるために，攻撃元の数を変えてシミュレーションを行った．このシミュレーションでは，複数の攻撃元から一箇所の宛先に対して，擬似的に攻撃を生成し，異なる 16 種類の時間帯のトラヒックに混入した．ここでは，攻撃レートの総和を 1000 Packets/sec と固定し，攻撃元を 1 から 5 に変化させた．また，各送信元からの攻撃レートは均一であるとする．つまり，攻撃元が 1 箇所の場合は，攻撃元ひとつあたりの攻撃レートは 1000 Packets/sec，攻撃元が 5 箇所の場合は攻撃元ひとつあたりの攻撃レートは 200 Packets/sec となる．またこのシミュレーションでは γ を 200 Packets/sec と設定した．

表 1 は，攻撃検出結果を示したものである．これより，いずれの攻撃元数の場合であっても，ほぼすべての攻撃元を特定可能であることが分かる．また，false-positive が数箇所あるが，被害者近くの出口へのトラヒックが急増している箇所である．この場合，その急増したトラヒックの経路が，その送信元から攻撃の被害者宛の経路と共通している部分が多く，トラヒックマトリクス推定の際に，誤差が生じてしまったものと考えられる．

4.2　γ と false-positive，false-negative の関係

図 3，図 4 に攻撃元の数が 4 の場合の，γ と false-positive，false-negative の関係を示す．図より，γ を小さな値に設定すると，false-negative は減るが，false-positive は増え，逆に γ を大きな値に設定すると，false-positive は減るが，false-negative は増えることがわかる．しかし，適切な γ を定めることにより，false-positive を低くしつつ，ほとんどの攻撃元を特定することが可能となることが分かる．適切な γ の値は次節で考察を行う．

また，図 3 と図 4 の false-negative を比較すると，false-negative は，攻撃レートの総計が 1000 Packets/sec の場合の方が大きいことが分かる．

4.3　γ と検出可能な攻撃レート，未検出の攻撃元からの攻撃レート

攻撃レートと，γ，検出可能な攻撃元の数の関係を調べるために，さまざまな攻撃レートの攻撃を擬似的に発生させたシミュレーションを行った．ここでは，攻撃元の数は 4 箇所とし，そのレートは均一であるとした．また，このシミュレーションにおいても，攻撃は 16 種類の異なる時間帯に加えた．

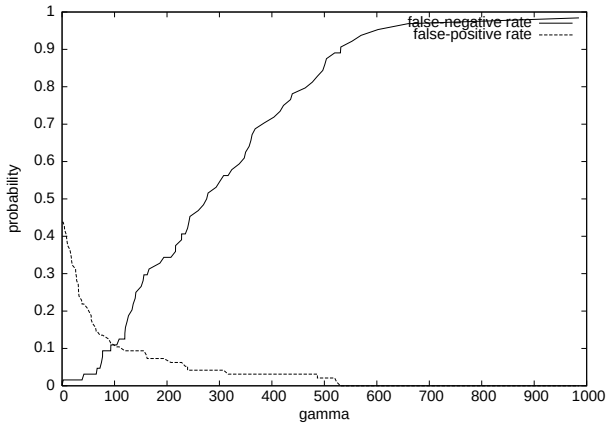

図 3　γ と false-negative･false-positive (攻撃レートの総計 500 Packets/sec)

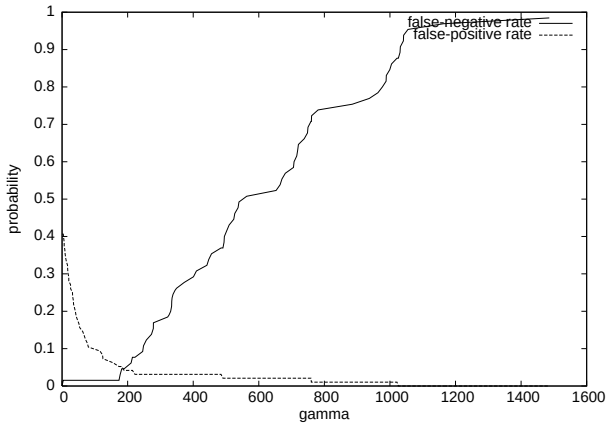

図 4　γ と false-negative･false-positive (攻撃レートの総計 1000 Packets/sec)

図 5 は，横軸に攻撃レートの総計をとり，すべての時間帯において，4 箇所の攻撃元のうち，1 箇所の攻撃元を特定可能な γ の値，2 箇所の攻撃元を検出可能な γ の値，3 箇所の攻撃元を検出可能な γ の値，4 箇所の攻撃元を特定可能な γ の値を示したものである．この図より，より小さな攻撃を特定するためには，γ を小さな値に定める必要があることが分かる．また，γ を同じ値に設定した場合であっても，攻撃のレートが高くなると，より多くの攻撃元を検出可能である．例えば，γ を 100 と設定した場合，攻撃レートの総計が 200 Packets/sec であれば，攻撃元のうち，1 箇所は特定可能であり，さらに攻撃レートが高い 600 Packets/sec であれば，攻撃元のうち，3 箇所が特定可能である．

図 6 は γ に対して，未検出の攻撃元からの攻撃レートの総計 (すべての時間帯における最大値，平均値) と，その γ を指定したときの false-positive を示したものである．この図より，γ を小さな値にすることにより，未検出の攻撃元からの攻撃レートは小さくなるが，false-positive が増え，逆に，γ を大きな値に設定すると false-positive は少なくなるが，未検出の攻撃元からの攻撃レートの総計も高くなることが分かる．

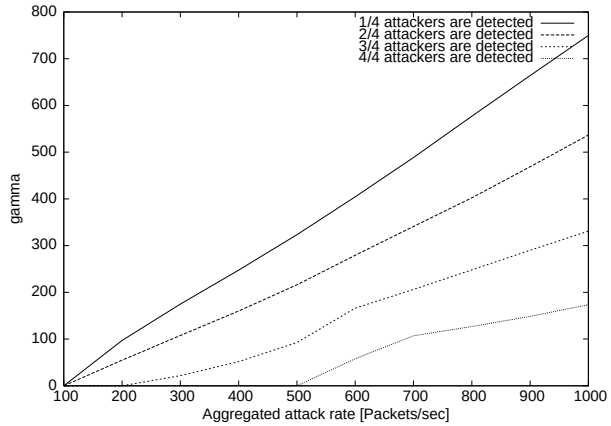

図 5　攻撃レートとそれを検出可能な γ

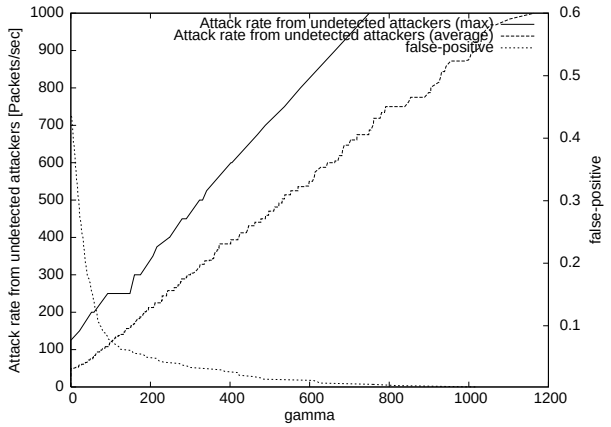

図 6　γ と未検出の攻撃レート

また，γ の値と未検出の攻撃元からの攻撃レートの関係に注目すると，未検出の攻撃元からの攻撃レートの平均は γ とほぼ等しくなっており，密接な関係があることが分かる．未検出の攻撃元の最大値は，γ が 300 以下の場合であれば，$\gamma + 100$ 程度，γ が 300 以上の値では，$\gamma + 200$ 程度の値となっている．未検出の攻撃元からの攻撃レートが $\gamma + 100$ の値になるのは，出口のリンクで観測される正常なトラヒックの増加量がもっとも小さい箇所では -100 Packets/sec となっており，$\tilde{g}^{\mathrm{out}}$ が実際の攻撃レートよりも 100 Packets/sec 小さな値に推定されることがあるためである．また，γ が 300 以上の箇所で，未検出の攻撃元からの攻撃レートが $\gamma + 200$ と推定されるのは，トラヒック増加量推定の際に ± 50 Packets/sec 程度の誤差が生じてしまうことが原因である．この場合，ある攻撃元は実際のレートより 50 Packets/sec 高く，別の攻撃元は実際の攻撃レートよりも 50 Packets/sec 低く推定され，その結果，さらに 100 Packets/sec ほど未検出の攻撃元からの攻撃レートが増えてしまう可能性がある．しかし，トラヒック増加量の推定値が $f^{\mathrm{normal}} - \mu - \gamma$ 以上である送信元は確実に攻撃元とみなされるため，γ を十分に小さな値に設定することにより，トラヒック増加量の推定に誤差があっても，攻撃元を正確に特定可能である．

このように、未検出の攻撃元からの攻撃レートは γ の値と密接な関係があるため、攻撃により増加しても影響を受けないトラヒック量を定めることにより、未検出の攻撃元からの攻撃レートをその影響をうけないトラヒック量以下になるように、適切な γ の値を定めることが可能であると考えられる．

5. まとめと今後の課題

本稿では，トラヒックマトリクス推定を用いた，新たな攻撃元特定手法を提案した．提案手法では， SNMP を用いて収集したリンクのトラヒック情報をもとに，送信元・宛先間のトラヒック増加量を推定し，それをもとに，トラヒックを急増させる攻撃元を特定する．シミュレーションにより，提案手法が正確に攻撃元を特定できていることを示した．今後の課題としては，実際のネットワークでのデータを用いた評価が挙げられる．

文　献

[1] S. Wu, L. Zhang, D. Massey, and A. Mankin., “On design and evaluation of intention-driven ICMP traceback,” in *Proceedings of IEEE International Conference on Computer Communications and Networks*, Apr. 2001.

[2] B.-T. Wang and H. Schulzrinne, “A denial-of-service-resistant IP traceback approach,” in *Proceedings of IEEE Symposium on Computers and Communications*, June 2004.

[3] S. Savage, D. Wetherall, A. Karlin, and T. Anderson, “Practical network support for IP traceback,” in *Proceedings of ACM SIGCOMM 2000*, Aug. 2000.

[4] D. X. Song and A. Perrig, “Advanced and authenticated marking schemes for IP traceback,” in *Proceedings of IEEE INFOCOM 2001*, Apr. 2001.

[5] K. Law, J. C. Lui, and D. K. Yau, “You can run, but you can’t hide: An effective methodology to traceback DDoS attackers,” in *Proceedings of International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems*, Oct. 2002.

[6] C. Partridge, L. A. Sanchez, C. E. Jones, F. Tchakountio, B. Schwartz, S. T. Kent, and W. T. Strayer, “Single-packet IP traceback,” in *Proceedings of the ACM SIGCOMM 2001*, Aug. 2001.

[7] T.-H. Lee, W.-K. Wu, and T.-Y. W. Huang, “Scalable packet digesting schemes for IP traceback,” in *Proceedings of IEEE International Conference on Communications 2004*, June 2004.

[8] A. Lakhina, M. Crovella, and C. D. February, “Diagnosing network-wide traffic anomalies,” in *Proceedings of ACM SIGCOMM 2004*, Aug. 2004.

[9] Y. Zhang, M. Roughan, N. Duffield, and A. Greenberg, “Fast accurate computation of large-scale ip traffic matrices from link loads,” in *Proceedings of ACM SIGMETRICS*, June 2003.

[10] D. Watson and C. Labovitz, “Experiences with monitoring OSPF on a regional service provider,” in *Proceedings of IEEE International Conference on Distributed Computing Systems (ICDCS)*, Mar. 2003.

[11] A. Fedmann, A. Greenberg, C. Lund, N. Reingold, and J. Rexford, “NetScope: Traffic engineering for IP networks,” pp. 11–19, Apr. 2000.

[12] “Scilab development team.” available at `http://www-rocq.inria.fr/scilab/`.