低レートのDoS攻撃に対応する改良型ICMP Traceback

高田 友則 † 中山 雅哉 †

† 東京大学大学院 新領域創成科学研究科

ICMP Traceback (iTrace) は，送信元 IP アドレスを偽装したパケットによる DoS 攻撃の攻撃元を，被害者が特定するのに有効な手法である．しかし，iTrace パケットがネットワークに与える負荷を考慮し，iTrace パケットの生成確率を小さな値としているため，攻撃パケットの発生レートが低い DoS 攻撃には適さないという問題があった．本稿では，その問題を解決すべく，ICMP Traceback を改良した ICMP Traceback with Periodical Transmission (iTrace-PT) を提案する．iTrace-PT は，各ルータで一度 iTrace パケットを送ったあて先には，一定時間は再送しないことで，iTrace パケットの生成確率を上げながらネットワークに与える負荷を低く抑えられる．シミュレーションにより，iTrace-PT は，生成確率を上げてもネットワークに与える負荷を低く抑えられ，低い攻撃レートでも攻撃元を特定可能であることを示した．

Enhanced ICMP Traceback against Low Rate DoS Attack

Tomonori Takada † Masaya Nakayama †

† Graduate School of Frontier Sciences, The University of Tokyo

ICMP Traceback (iTrace) is an efficient technique for a victim to specify sources of DoS attack caused by spoofed packets. However, it generates an iTrace packet with a low probability in consideration of the overhead given to the network. Therefore, it is unsuitable for low rate DoS attacks. In this paper, we propose ICMP Traceback with Periodical Transmission (iTrace-PT), which is an improvement of ICMP Traceback technique, to solve the stated problem above. In the iTrace-PT, each routers don't transmit iTrace packets in a constant time to destinations where it is once already transmitted. By doing so, the generation probability of the iTrace packet can be increased and the overhead is kept low. We verify by simulation results that the iTrace-PT keeps the overhead low even if the generation probability of the iTrace packet is increased and it can specify sources of attack even on low rate attack.

1 はじめに

DoS (Denial of Service) 攻撃とは，攻撃者が標的とするホストに対して，大量のパケットを送り，ホスト周辺のネットワークを利用できなくしたり，そのホストの資源を枯渇させたりすることで行っているサービスを妨害する攻撃の総称だが，細かく分類すると攻撃者が単独である場合を DoS 攻撃といい，分散している場合を DDoS (Distributed DoS) 攻撃という．

シマンテック社によれば，DoS・DDoS 攻撃の発生件数は年々増加し，2006 年 7 月 1 日から 12 月 31 日の間には一日平均 5213 件が記録されて [1]，今日のインターネット社会において深刻な問題となっている．特に近年は，インターネットに常時接続する初心者 PC ユーザが増加しているため，利用者が気が付かないうちに，攻撃者から遠隔操作されて DDoS 攻撃の一端として悪用されるケースが急増している．

攻撃者は，攻撃元をすぐに検知されない様に，送信元 IP アドレスを偽装した攻撃パケットを用いることが多い．そのため，被害者は受け取った攻撃パケットから，攻撃元を容易には特定できない．このような送信元 IP アドレスを偽装した攻撃パケットによる DDoS 攻撃の攻撃元を特定するために，これまでマーキング方式 [2] やロギング方式 [3]，ICMP Traceback (iTrace) [4] といった手法が提案されてきた．

これらの手法のうち，ICMP Traceback (iTrace) [4] は，Bellovin らが 2000 年に，IETF (Internet Engineering Task Force) に提案した手法で，ルー

タが一定の確率ごとに，中継パケットと同じあて先IPアドレスを持つICMPパケットを生成する．このICMPパケットのデータ部には，ルータのアドレス等が記述される．このICMPパケットをiTraceパケットと呼ぶ．iTraceパケットを受け取った被害者は，iTraceパケット内のデータから，生成したルータのアドレス情報を得て攻撃元までの経路を特定でき，他の手法に比べて有効性が高い方式だと考えられている．

iTrace手法では，ネットワークに与える負荷を考慮し，iTraceパケットの生成確率を小さな値(1/20000)とするように提案されている．このため，個々の攻撃パケットの生成レートを低く設定したDDoS攻撃や，Low-Rate attack [9]では，各ルータがiTraceパケットを生成するのに非常に長い時間を要することになる．また，通常の通信にまぎれるほど低い攻撃レートの場合は，攻撃者だけの攻撃元特定が難しくなる．

本稿では，上記の問題を解決すべく，ICMP Tracebackを改良したICMP Traceback with Periodical Transmission (iTrace-PT) を提案する．iTrace-PTでは，ルータでiTraceパケットを生成したり，中継する際に，そのあて先IPアドレスを一定時間記憶し，当該時間帯には新たなiTraceパケットを生成しないようにする．この変更に伴い，ネットワークに与える負荷を低く抑えながら，各ルータのiTraceパケット生成確率を高めることを可能とし，低いレートの攻撃者も特定が可能になることが期待される．

シミュレーションにより，iTrace-PTは，生成確率を上げてもネットワークに与える負荷を低く抑えられ，低いレートの攻撃者も特定可能であることを示した．

以下，本稿では，2章でiTrace手法の概念と，低レートDoS攻撃における攻撃元特定に関する問題点について述べ，3章でその問題を解決するiTrace-PTについて述べるとともに，4章で簡単なネットワークモデルにおけるシミュレーション結果とその評価結果を示している．5章は，本論文の結論と今後の課題についてまとめている．

2 iTrace手法の概要

ICMP Traceback (iTrace) [4]は，Bellovinらが2000年にIETFに提案した，ICMPを用いたTraceback手法である．パケットがルータに到着すると，そのルータのIPアドレス等をデータ部に記載したiTraceパケットを一定の確率で生成し，中継したパケットと同じあて先に送る．被害者は，受信したiTraceパケットのデータを基にして，攻撃元までの経路を特定できる．

iTrace手法では，被害者から攻撃者までの経路上の全ルータからのiTraceパケットを収集する必要があるため，攻撃元の特定に長く時間を要する場合がある．そのため，iTrace手法を改良した提案が幾つかなされている [5, 6, 7, 8]．例えば，ICMP Traceback with Cumulative Path (iTrace-CP) [5]は，iTraceパケットを生成したルータから被害者までの経路情報を一度にデータ部に記載して送るため，攻撃者の直近ルータからのiTraceパケットを受信するだけで，攻撃元までの経路を特定できる．

iTraceを用いる手法は，ルータがパケットの中継に伴って一定の確率で，新たにiTraceパケットを生成することになるため，ネットワークに与える負荷を考慮する必要がある．Bellovinらは，ネットワークに与える負荷を0.1%以下に抑えるために，各ルータが生成するiTraceパケットの生成確率を1/20000に定めている．このため，個々の攻撃パケットの生成レートを低く設定したDDoS攻撃や，Low-Rate attack [9]では，各ルータがiTraceパケットを生成するのに長い時間を要する問題が生じることとなる．例えば，個々の攻撃レートが50ppsのDDoS攻撃を考えると，被害者は攻撃者の直近ルータからのiTraceパケットを1パケット受け取るのに平均で400秒の時間が必要となる．さらに，全ての攻撃元を特定するためには，全攻撃者の直近ルータからのiTraceパケットの受信を待つ必要があり，誤判定を防ぐために同じルータからの複数のiTraceパケットを収集する必要があるため，より長い時間が必要になる．

また，通常の通信にまぎれるほど低い攻撃レートで攻撃が行われた場合，攻撃者だけの攻撃元特定が困難であるという問題も生じる．

3 iTrace-PTについて

前章で述べた様にiTrace手法は，低レートでのDoS攻撃に対する脆弱性を有する．主な問題点は，以下の二つである．

- 被害者がiTraceパケットの受信に時間を要する
- 被害者が攻撃元を特定することが困難

1つ目の問題は，iTraceパケットがネットワークに与える負荷を0.1%以下に抑えるために，各ルー

タの iTrace パケット生成確率を 1/20000 に設定している点にある．そこで，iTrace-PT では，iTrace パケットの生成確率を上げながらネットワークに与える負荷を低く抑えるため，各ルータが一度 iTrace パケットを送ったあて先には，一定時間は再送しないという生成方式を提案する．

iTrace-PT では，2 つ目の問題も解決することが可能となるが，その詳細は次章で示す．

3.1 iTrace パケット生成アルゴリズム

iTrace-PT では，一度送った iTrace パケットのあて先 IP アドレスをルータで一定時間記憶する．ただし，単純にあて先 IP アドレスを記憶すると搭載するメモリ量が膨大に必要となるため，Bloom Filter を用いて一定のメモリ量で実現する機構を用いることとする．

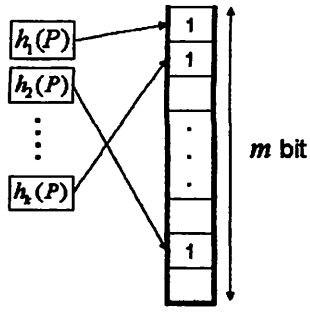

Fig. 1 Bloom Filter

Bloom Filter は，ある要素が集合のメンバーであるかどうかの判定を少ないメモリ量で実現するための手法であり，以下の様にする．Bloom Filter のデータ構造は，m ビットのビット配列であり，最初は，全てのビットを 0 にしておく．ある要素 P を Bloom Filter に追加するときは，k 種類のハッシュ関数を用いて，k 個のキー値を求め，対応する配列のビットを 1 に変更する (図 1)．ある要素 P が Bloom Filter に記憶されているかどうかを確認するときは，要素 P に対する k 個のキー値を求め，対応したビット配列の要素が全て 1 になっている場合には，要素 P が記憶されていると判定する．

また，提案する iTrace-PT における iTrace パケットの生成アルゴリズムを図 2 に示す．

ルータ R が，パケット a を受信すると，まず，iTrace パケットかどうか判定する．もし，iTrace パケットならば，a のあて先 IP アドレスを Bloom Filter に追加し，iTrace パケット a のデータ部に，ルータの IP アドレス R を追加し，送信する．

```
let BF be a hashtable of Bloom Filter
let p be a generation probability of iTrace packet
let t be a hash clear interval
let m be a timer
at Router R
for each received packet a
  if isiTrace(a) = true then
    set(a.dest,BF)
    append(a, R)

  let x be a random number from [0,1]
  if x ≤ p then
    if isset(a.dest,BF) = false then
      generate_iTrace(a)
      set(a.dest,BF)

  if m ≥ t then
    clear(BF)
    m = 0
```

Fig. 2 iTrace packet generation algorithm

iTrace パケットでない場合は，$[0, 1]$ の乱数 x と iTrace パケットの生成確率 p の値とを比較して，$x \leq p$ となり，かつ a のあて先 IP アドレスが Bloom Filter に記録されたことがない場合，a に対応した iTrace パケットを生成し，a のあて先 IP アドレスを Bloom Filter に追加する．

一定時間 t 経ったら，Bloom Filter のビット配列の全ビットを 0 にクリアするとともに，タイマー m を 0 に戻す．今後，Bloom Filter のビット配列の全ビットを 0 にクリアする処理をハッシュクリアすると呼ぶ．この様に，定期的にハッシュクリアを行うことで，ルータ R から定期的な iTrace パケットの送信を実現している．

4 シミュレーションによる評価

本章では，前章で提案した iTrace-PT によるネットワークに与える負荷と攻撃元の特定能力に関して，簡単なモデルでシミュレーションを行った評価結果についてまとめている．なお，本シミュレーションは Bloom Filter のメモリサイズが十分に大きく，衝突は起きないものとする．

4.1 ネットワークに与える負荷

表1は，図3の線形トポロジにおいてiTraceパケットの生成確率を変化させたときのiTrace-PTにおけるルータR1の検出時間とネットワークに与える負荷を示している．本シミュレーションでは，攻撃レートを100pps, ハッシュクリア間隔を100秒とし，1000回行った時の結果を示している．

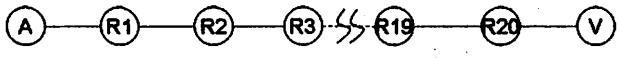

Fig. 3 Linear Topology

Table 1 Characteristic of various generation probabilities

	平均(秒)	標準偏差(秒)	Overhead(%)
確率1/1000	9.96	10.02	0.03637
確率1/2000	20.17	20.65	0.03595
確率1/5000	52.16	51.43	0.03428
確率1/10000	103.08	103.98	0.03175
確率1/20000	199.68	192.69	0.02698

表1にはiTraceパケットの生成確率が大きくなるにつれ，R1からのiTraceパケットを受け取るまでの時間が速くなるものの，ネットワークに与える負荷はほとんど変わらないということが示されている．これは，iTrace-PTでは，iTraceパケットを一度送ったあて先にはハッシュクリアされるまでは再送されないためであり，ネットワークへの負荷はハッシュクリア間隔によって基づいた一定の値に抑えることができる．しかし，生成確率が1/10000以下になると，iTraceパケットを生成する平均時間がハッシュクリアする時間を超えてしまうため，R1からのiTraceパケットが一度も被害者で受信されない状況となるため，ネットワークへの負荷は低い値になっている．

図4は，図3の線形トポロジにおいて中継ルータ数を変化させたときのネットワークに対する負荷の変化を理論値とシミュレーションの結果で示している．シミュレーションは，生成確率1/1000, 攻撃レート100pps, ハッシュクリア間隔100秒とし1000回行った．中継ルータ数nのとき発生するiTraceパケットの総数の期待値$A(n)$は，被害者側からk番目のルータが最初にiTraceパケットを送ったとすると，iTraceパケットの総数の期待値が，$A(n-k)+1$となるため，(1)式のように表せる．

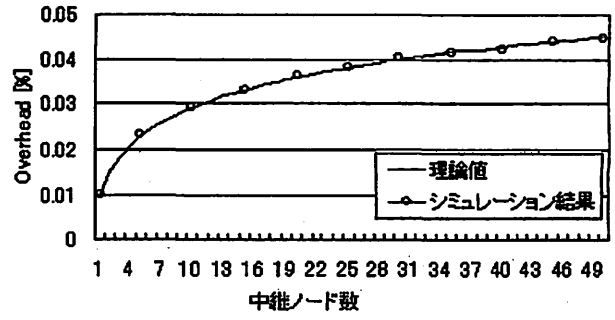

Fig. 4 Traffic overhead ($p = 1/1000, f = 100, t = 100$)

$$A(n) = \frac{1}{n}\sum_{k=1}^{n}\{A(n-k)+1\} \qquad (1)$$

これより，(2)式が導かれる．

$$A(n) = A(n-1) + \frac{1}{n}, \quad A(0) = 0 \qquad (2)$$

ゆえに，生成確率をp, 攻撃レートをf(pps), ハッシュクリア間隔をt(秒)とすれば，$t > \frac{1}{pf}$のとき，ネットワークに与える負荷は，$\frac{A(n)}{ft}$となる．

中継ルータ数が20のとき，ネットワークに与える負荷は約0.036%となり既存iTrace手法の36%程度に抑えることができる．また，中継ルータ数が50と大きな規模になってもネットワークに与える負荷は0.1%を超えていない．

4.2 攻撃元の特定能力

iTrace-PTでは，C回のハッシュクリアが行われたとき，閾値をαとして，$C\alpha$個以上のiTraceパケットを発生させたルータを，攻撃者の直近ルータであると特定することができる．これは，図3に示すように，攻撃者Aから被害者Vまでに$R_i(1 \leq i \leq 20)$のルータが置かれている単純な通信モデルでは，攻撃者Aからk番目のルータR_kのiTraceパケットがハッシュクリア間隔内に被害者Vに届く確率$P(k)$が，以下のように表せることによる．

$$P(k) = \frac{p(1-p)^{k-1}\{1-(1-p)^{kft}\}}{1-(1-p)^{k}} \qquad (3)$$

図5は，(3)式で与えられる理論値とシミュレーションを行った結果である．シミュレーションは，生成確率1/1000, ハッシュクリア間隔100秒，攻撃レート100ppsとし，ハッシュクリアを10000回行ったときのk番目のルータが生成したiTraceパケットの発生個数分布である．

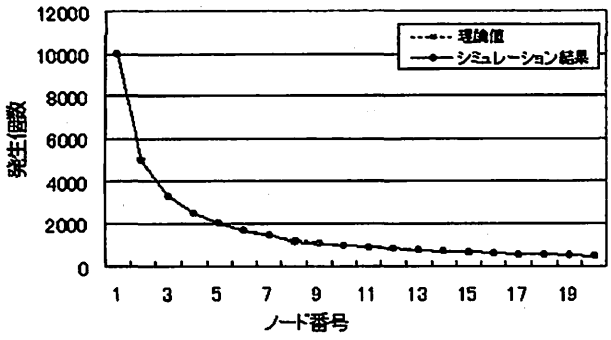

Fig. 5 Distribution of the number of generated iTraces

この図より，閾値 α を，$0.5 < \alpha \leq 1$ とすれば，攻撃者の直近ルータを判別できることが分かる．

以下で，この攻撃元特定手法により，3 章で挙げた 2 つ目の問題点が解決される理由を述べる．iTrace-PT による攻撃元特定手法は，既存の iTrace 手法とは異なり攻撃パケット数に基づいて攻撃元を特定するのではなく，攻撃の持続時間に基づいて攻撃元の特定が行われていることになる．つまり，高ビットレートの攻撃も低ビットレートの攻撃も，C 回のハッシュクリアの時間だけ攻撃パケットが継続している場合，攻撃者の最寄りルータからの iTrace パケットは C 個届くことになる．一方，正当な通信が攻撃パケットよりも高ビットレートだったとしても，通信が攻撃計測時間より短ければ，当該ルータから受け取る iTrace パケットは，C 個より少なくなり，攻撃元と誤認される確率は低く抑えることができる．ここでは，DDoS 攻撃の様な比較的長期間に及ぶ攻撃を前提としているため，本提案方式は有効に機能するはずである．このように，iTrace-PT では，攻撃者の通信が通常の通信にまぎれるほど小さくても，攻撃時間が長ければ，攻撃者だけを特定することができる．

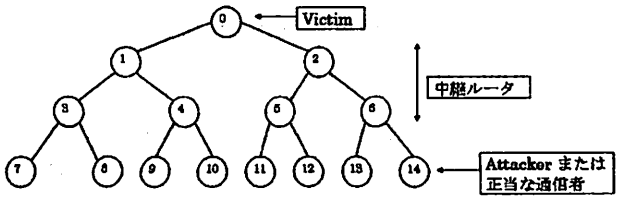

Fig. 6 Binary Tree Topology

この攻撃元特定手法について，DDoS 攻撃を想定したトポロジ (図 6) を用いて攻撃者だけがいる状況でシミュレーションを行った．評価項目は，以下の式で表される False Positive と False Negative である．攻撃者の最寄ルータを攻撃元とし，それ以外は攻撃元でないとする．

$$FalsePositive = \frac{\text{特定した中で攻撃元でないノード数}}{\text{攻撃元と特定したノード数}}$$

$$FalseNegative = \frac{\text{特定できなかった攻撃元の数}}{\text{検出すべき攻撃元の数}}$$

図 7，図 8 に iTrace-PT の攻撃元特定における False Positive，False Negative をそれぞれ示す．

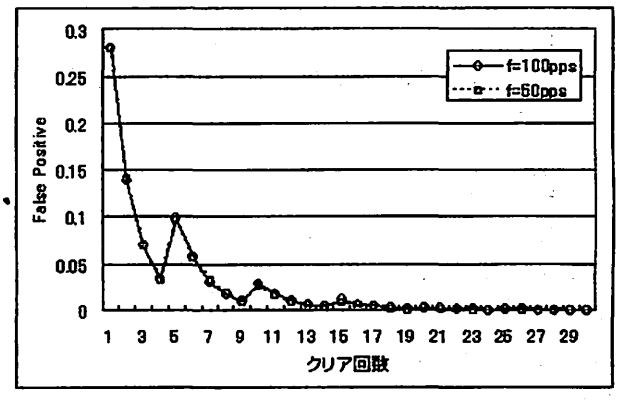

Fig. 7 False Positive

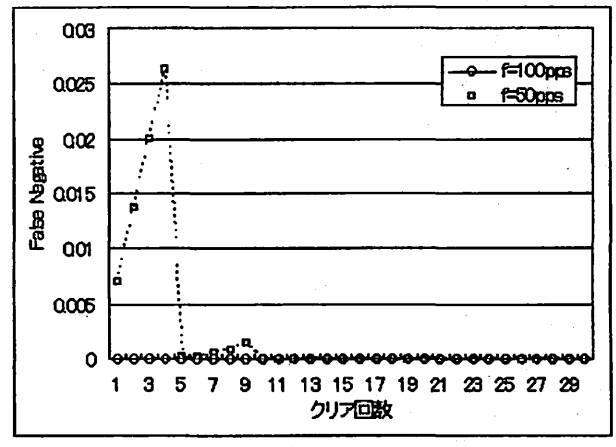

Fig. 8 False Negative

シミュレーションは，生成確率 1/1000，中継ルータ数 10，個々の攻撃レート 50,100(pps)，ハッシュクリア間隔 100 秒，閾値 0.8 とし 30 回行った．また，攻撃者はリーフノードの 1/2 だけ (左側ノードのみ) に配置し，一定レートで攻撃を行う．

図 7 の False Positive の結果から，攻撃レートによらず同じような攻撃元特定能力があることがわか

る．12回目のハッシュクリア時には，False Positiveは1%を切る値まで減少している．閾値を0.8とするとクリア回数が5の倍数時，攻撃元の特定条件は一回前と同じになり，攻撃元の特定条件が緩くなるため，False Positiveが増えている．

図8のFalse Negativeの結果から，攻撃レートが低くハッシュクリアされるまでにiTraceパケットが送られないケースでは，短時間で全攻撃者の最寄りルータを検知できないことが分かる．しかし，このような場合でも10回程度の時間をかければ全ての攻撃元を特定できることが示されている．

これら二つの結果を総合すると，ハッシュクリアを10回程行えば，全ての攻撃元を特定でき，そのときのFalse Positiveは，約2.6%に抑えることもできる．

5 まとめ

本稿では，iTraceを改良した手法，ICMP Traceback with Periodical Transmission (iTrace-PT)を提案した．iTrace-PTは，一度iTraceパケットを送ったあて先には，ハッシュクリアするまで再送しない手法である．これにより，iTraceパケットの生成確率を上げてもネットワークに与える負荷を低く抑えることができる．

シミュレーションにより，iTrace-PTは，iTraceパケットの生成確率を上げてもネットワークへの負荷を低く抑えられ，DDoS攻撃のように個々の攻撃レートが低くても，攻撃持続時間が長ければ攻撃元を特定できることを示した．具体的には，1024の攻撃者，個々の攻撃レート50ppsのDDoS攻撃を，10回のハッシュクリアで，False Positive約2.6%で，全ての攻撃元を特定することができた．iTrace-PTは，既存手法に比べ非常に有効な手法であるといえるが，ルータにおいてあて先IPアドレスを記憶するためメモリを必要とする点や，攻撃レートが極端に低いワーストケースを考えると，ネットワークに与える負荷が増加し，小さな閾値を設定しなければ，攻撃元を特定できないという問題がある．

今後の課題としては以下のようなことが挙げられる．

- 適切なパラメータの設定
  iTraceパケットの生成確率やハッシュクリア間隔，閾値の値，Bloom Filterのメモリサイズによって，iTrace-PTの性能が変わるため適切な値を選択する必要がある．
- iTraceパケット認証の簡略化
  攻撃者が偽のiTraceパケットを送れないようにする方法の検討が必要である．例えば，公開鍵認証方式等でiTraceパケットを認証する方法が考えられるが，公開鍵認証方式の処理負荷が非常に高くなるため，新たな工夫が必要となる．

参考文献

1) Symantec Corporation, "Symantec Internet Security Threat Report", September 2006, http://www.symantec.com/region/jp/istr/

2) D. Song, A. Perrig, "Advanced and Authenticated Marking Schemes for IP Traceback", Proc. IEEE INFOCOM, 2001.

3) A.C. Snoeren et al, "Hash-Based IP Traceback", ACM SIGCOMM 2001, August 2001.

4) S.M. Bellovin, "ICMP Traceback Messages", Internet draft: draft-vellovin-itrace-00.txt, March 2000.

5) H.C.J. Lee, V.L.L. Thing, Y. Xu, M. Ma, "ICMP Traceback with Cumulative Path, An Efficient Solution for IP Traceback", Proc. 5th International Conference on Information and Communications Security (ICICS '03), pp.124-135, October 2003.

6) V.L.L. Thing, H.C.J. Lee, M. Sloman, J. Zhou, "Enhanced ICMP Traceback with Cumulative Path", Proc. 61st IEEE Vehicular Technology Conference, May 2005.

7) A. Mankin et al, "On Design and Evaluation of "intention-driven" ICMP Traceback", Proc. IEEE International Conference on Computer Communications and Networks, April 2001.

8) B. Wang, H. Schulzrinne, "Multifunctional ICMP Messages for e-Commerce", Proc. IEEE EEE, Mar 2004.

9) A. Kuzmanovic, E. Knightly, "Low-Rate TCP-Targeted Denial of Service Attacks (The Shrew vs. the Mice and Elephants)", Proc. ACM SIGCOMM 2003, August 2003.