

待ち行列推定に基づくパケットロス攻撃検知の輻輳強度依存性

細井 琢朗^{1,a)} 松浦 幹太^{1,b)}

概要：ネットワーク内のルータを乗っ取って通信を操作する攻撃の一つに、そのルータを通過する任意のパケットを削除し通信を妨害する、パケットロス攻撃がある。輻輳による通常のパケット廃棄がある中でこのパケットロス攻撃を高精度で検知する方式が Mizrak らによって提案されたが、その性能評価はまだ網羅されていない。本稿では彼らの方式によるパケットロス攻撃の検知性能について、輻輳の程度に対する依存性を調べた結果を報告する。

HOSOI TAKUROU^{1,a)} MATSUURA KANTA^{1,b)}

1. はじめに

インターネットの通信は、ネットワークの結節点であるルータが正しい経路に導くことで成り立っている。ネットワークに接続された他の機器と同様、このルータもネットワークを通じた攻撃を受け、乗っ取られることがある。攻撃者に乗っ取られたルータは、そこを通過するパケットを操作することで、ネットワーク内の通信へ攻撃を仕掛けることができる。この攻撃方法には大別して、ネットワーク制御面での攻撃と、ネットワークデータ面での攻撃の二つがある。前者はルータのルーティングテーブルの操作など、破壊的な影響が懸念される攻撃を含むため、それらに対する研究がこれまで多くなされてきている。一方後者は、サービス妨害攻撃や中間者攻撃、リプレイ攻撃などを含む。中でも通過する任意のパケットを削除するパケットロス攻撃は、選択的に行うことで削除パケット量の少なさに対して大きな被害を与える能力を持つ。例えば、TCP のコネクション確立のためにまず出される TCP SYN パケットをあるサーバへの分だけ削除することで、このサーバを利用しようとしているユーザにタイムアウトまでの比較的長い時間待たせることを強いられる攻撃ができる [2]。

このパケットロス攻撃を検知するには、あるばずのパケットが無いことを検知する必要があり、他の攻撃検知技

術に比べて困難な問題になっている。また、パケットの廃棄自体は、インターネットプロトコルでも通信の混雑に対応するために正規に行われる。パケットロス攻撃はこれと区別して検知されなければならない。

初期の検知方法は、送信側から送られるパケットの送信数と実際に受け取ったパケット数の違いからパケットの廃棄を検知し、その廃棄パケット数がある閾値を超えた場合に攻撃と判断するものであった。この方式は適切な閾値の設定が難しいだけでなく、攻撃者はこの閾値未満であれば攻撃として検知されずにパケットを削除することができてしまう問題がある。

別の対策設計方針として、混雑によるパケット廃棄のモデルを立て、それを基に正規のパケット廃棄と攻撃を区別する方法もある。しかし、この方法では攻撃検知に十分な精度で混雑によるパケット廃棄をモデル化するのは難しいという問題があった。

その後 Mizrak らによって、ルータ内のパケット転送の待ち行列（キュー）を推定することで、高い検知性能を持つ、現実的なパケットロス攻撃検知方式が提案された [1]。この方式は、あるルータにおいてパケットロス攻撃が行われたかどうかを、通信混雑による正規のパケット廃棄と区別して検知する。パケットの廃棄が通信混雑による正規のものかどうかの区別は、隣接するルータでの通信の観測情報からパケット転送の待ち行列の塞がり具合を推測した結果を用いて、統計学的に行う。

この待ち行列推定に基づくパケットロス攻撃検知方式

¹ 東京大学
The University of Tokyo
a) hosoi@iis.u-tokyo.ac.jp
b) hosoi@iis.u-tokyo.ac.jp

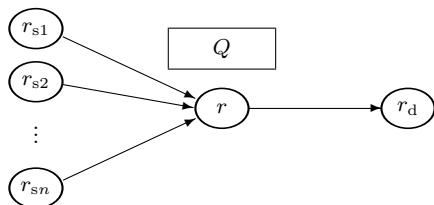


図 1 ネットワーク構成図。

は、実際の通信に対してリアルタイムに実行可能であることが実装実験で確かめられており、少ない通信負荷増加で高い検知精度を示した。しかし実装実験のため、その性能評価はまだ網羅されていない。この方式が含む調整可能なパラメータや通信環境を表すパラメータに対する検知性能の依存性の調査の初めとして、我々は文献 [3] で幾つかの指標についてその検知性能への影響を簡単に調べた。その結果、適度に正規のパケット廃棄がある通信状況（正規のパケット廃棄が送信パケット数の数%～十数%程度）では、通信帯域幅や待ち行列の容量を変化させても検知性能は大きく変わらないことが分かった。しかし実際には、通信量が少なく正規のパケット廃棄が全く無い状況から通信過多のために多くのパケットが正規に廃棄される状況まで、さまざまな通信状況が起こりうる。そこで本稿では、この検知方式によるパケットロス攻撃の検知性能について、輻輳の程度に対する依存性を調べた結果を報告する。

2. 攻撃検知方式

本節では、Mizrak らによって提案された待ち行列推定に基づくパケットロス攻撃検知方式 [1] について、その攻撃検知方法を大まかに説明する。

2.1 ネットワークモデル

パケットロス攻撃を行っているかどうかを調べる対象のルータを r とする。 r は、隣接するルータ $r_{s1}, r_{s2}, \dots, r_{sn}$ から、隣接するルータ r_d へパケットを転送する（図 1）。転送されるパケットは、ルータ r 内の待ち行列 Q に一旦入れられ、その先への転送が可能になると、順に取り出され、ルータ r_d へ送信される。 Q が順番を待っているパケットで埋まっていると、新たに到着したパケットは入ることができず、廃棄される（混雑によるパケット廃棄）。各パケットが上流側のルータ r_{sx} から中間のルータ r を通り下流側のルータ r_d に届くまでには、待ち行列 Q での順番待ちで費やす時間の他に、中間のルータの送受信処理の時間と、パケット長と回線の帯域幅に応じた、各ルータ間での転送時間が掛かる。

攻撃検知は一定時間毎に、隣接するルータでの通信の観測情報を照らし合わせて行う。そのため、各ルータの時計は全て同期していると仮定する。また、各回線の帯域幅や

対象とするルータの待ち行列の容量なども全てのルータが知っていると仮定する。

2.2 通信の観測情報

送信側のルータ $r_{s1}, r_{s2}, \dots, r_{sn}$ からは、受信側のルータ r_d へ、送信したパケットの情報が以下の組で送られる。

$$\{ (\text{パケットのフィンガープリント}), (\text{パケット長}), (\text{発信時刻}) \} \quad (1)$$

この通信情報は、一定時間毎にまとめられ、各送信側ルータから揃って r_d へ送られる。

これに対応して、受信側のルータ r_d でも同様に通信を観測し、送信側と同じ時刻に同じ時間毎でまとめておく。この場合は、(1) 式の最後には（受信時間）を入れる。

元の検知手法では、これらの通信情報は署名などを施した上で送信される。これにより、この検知手法のプロトコルに従わないルータがあっても、それをネットワーク制御面での攻撃として検知できるようになっている。この部分は通信混雑によるパケット廃棄とパケットロス攻撃の区別には関係しないため、本稿では割愛する。

2.3 検知手順

受信側のルータ r_d で送信側の通信情報と受信側の通信情報が集まったところで、この時間区間における受信側のルータ r の待ち行列 Q へのパケットの出入りから、 Q 内に溜まっているパケットの総量の推測値 q_{pred} を以下の手順で順次計算する。

- (1) 通信情報を一つの配列にまとめ、送受信時間の早い順に並べる。
- (2) (1) の配列から順に一つずつパケット情報を取り出し、以下の方法で q_{pred} を更新する。
 - (a) このパケット情報が受信側のものなら,

$$q_{\text{pred}}(t_{\text{current}}) = q_{\text{pred}}(t_{\text{prior}}) - (\text{パケット長})$$
 - (b) このパケット情報が送信側のもので、受信側にも対応するパケット情報があるなら,

$$q_{\text{pred}}(t_{\text{current}}) = q_{\text{pred}}(t_{\text{prior}}) + (\text{パケット長})$$
 - (c) このパケット情報が送信側のもので、受信側には対応するパケット情報がないなら,

$$q_{\text{pred}}(t_{\text{current}}) = q_{\text{pred}}(t_{\text{prior}})$$

これがパケット廃棄の検知にあたる。

(2c) のパケット廃棄が通信混雑による正規のものか、パケットロス攻撃によるものかは、 Q が満杯かどうかを q_{pred} で推測して判定する。これば単純な比較

$$(Q \text{ の容量}) < q_{\text{pred}} + (\text{パケット長}) \quad (2)$$

でも判定できるが、その場合 q_{pred} の推定の不正確さから

来る判定間違いが多く出てきてしまう。これを抑えるため、文献 [1] では統計学に基づいた判定を行う。

$$\mu = (q_{\text{pred}} \text{ の真値からのずれの平均}) \quad (3)$$

$$\sigma = (q_{\text{pred}} \text{ の真値からのずれの標準偏差}) \quad (4)$$

この時間区間に内に n 個のパケット廃棄が見つかったとする。各廃棄パケットについて、以下の方法で計算した信頼値と予め設定した有意水準の比較を行い、パケットロス攻撃かどうかを判定する。

$$y = \frac{(Q \text{ の容量}) - q_{\text{pred}} - (\text{パケット長}) - \mu}{\sigma} \quad (5)$$

$$c = \frac{1 + \text{erf}(y/\sqrt{2})}{2} \quad (6)$$

$$c < s_{\text{single}} \text{ ならば, 混雑による廃棄} \quad (7)$$

$$c \geq s_{\text{single}} \text{ ならば, パケットロス攻撃} \quad (8)$$

個々の廃棄パケットについての判定ですべて攻撃でないと判定された場合は、それに続けて、この時間区間に内 n 個のパケット廃棄全体についてもパケットロス攻撃によるパケット削除が含まれているかどうかを判定する。ここでも以下の方法で信頼値を計算し、これと予め設定した有意水準を比較して攻撃の判定を行う。

$$z = \frac{(Q \text{ の容量}) - (q_{\text{pred}} \text{ の平均}) - (\text{パケット長の平均}) - \mu}{\sigma \sqrt{n}} \quad (9)$$

$$c = \frac{1 + \text{erf}(z/\sqrt{2})}{2} \quad (10)$$

$$c < s_{\text{comb}} \text{ ならば, 全て混雑による廃棄} \quad (11)$$

$$c \geq s_{\text{comb}} \text{ ならば, パケットロス攻撃を含む} \quad (12)$$

これにより、この時間区間に内に対象とするルータ r がパケットロス攻撃をおこなったかどうかが判定できる。

3. 調査方法

文献 [1] では、提案された待ち行列推定に基づくパケットロス攻撃検知方式の性能を、実機への実装を使った実験で評価していた。この評価方法は実時間での処理の検証が可能であるなどの利点を持つが、方式が含むパラメータに対する振る舞いを検証することは多くの場合困難である。そこで本稿では文献 [3] と同様にシミュレーション実験により性能評価を行う。

ネットワークの構成は文献 [1] の実装実験に倣い、図 1 の構成で、送信側のルータを二つとした。

文献 [1] の実装実験において、通信混雑による正規のパケット廃棄を発生させるために大きなデータのダウンロードを主な通信にしたところ、ほとんどのパケットが最大パケット長のものになることが分かった。そこで今回のシ

ミュレーションでは簡単のために、全てのパケットは同一のパケット長 (1500 bytes) を持つものとする。

送信側からのパケットの送信は簡単のため、凡そ一定間隔 (単位時間当たり 1 パケット) で行うこととする。ただし監視対象とするルータ r での待ち行列の塞がり具合に変化を与えるため、この間隔は乱数により送信毎にある程度揺らがせる。

通信の輻輳は、通信量に対して回線の帯域幅の大きさが小さい場合、または通信の数に対してルータの処理能力が小さい場合に起こる。回線の帯域幅を小さくするとパケット転送の遅延が増え、同期して動作するこの検知方式に余計な不都合を与える。そのため本稿では、中間のルータ r の転送処理能力 (単位時間当たりに送信できるパケット数) を変えることで輻輳を発生させた。

パケットロス攻撃は、監視対象とするルータ r で適宜適当なパケットをその先へ送らずに削除することで行った。通信混雑による正規のパケット廃棄と区別して、これを見逃さずに発見できれば検知は成功となる。一方、これを見逃すと偽陰性発生になる。また、これ以外の箇所を攻撃として判定すると偽陽性として数える。

シミュレーションは待ち行列が空の状態で始め、検知の時間区間で 10 区間分進める分を一回の実行とする。一つのパラメータ設定に付き 10 回の実行を行った結果を集計して性能評価を行う。

4. 予備実験

このパケットロス検知方式は統計学的な検知方式であり、待ち行列 Q の推定値 q_{pred} の誤差の分布の標準偏差を使って検知を行う ((8) 式、(12) 式参照)。そこで本稿でも検知実験を行う前に、この標準偏差の値を実験により導く。

この実験は、パケットロス攻撃検知実験と同じシミュレーション実験をパケットロス攻撃が無い状態で実行することで遂行する。中間のルータ r の転送処理能力 (単位時間当たりに送信できるパケット数)、検知の時間区間、中間のルータ r の待ち行列の容量を変えたときのこの標準偏差の値は図 2、図 3 のようになった。今回の実験ではパケットは上流側のルータ一つに付き単位時間当たり一個送られて来る。上流側のルータは二つあるため、これらの図において横軸の値が 2 を超えるとほぼ輻輳は無くなる。一方横軸の値が 2 を下回ると輻輳は必ず起こるようになる。

この実験結果では、待ち行列の容量が小さい (パケットが二個しか入らない) 場合は誤差分布の標準偏差は中間のルータ r の転送処理能力にあまり依存しないが、待ち行列の容量が大きい (パケットが 20 個以上入る) 場合は横軸の値が 2 を超えるところでは容量が小さい場合と同様の値になり、横軸の値が 2 を下回ると一桁程度大きな値になっている。これは輻輳が起き続けている通信状況では、待ち行列で待たされる時間が容量の大きさだけ長くなり、待ち

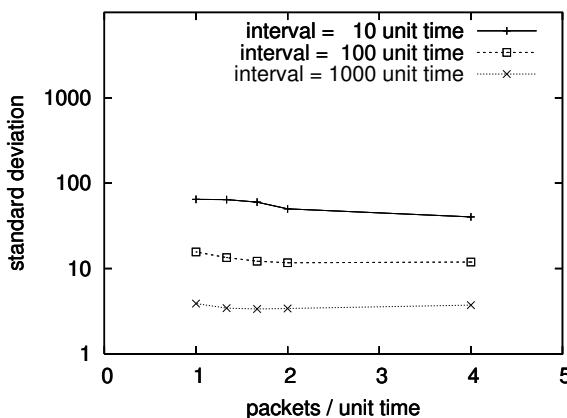


図 2 待ち行列推定の誤差の標準偏差 (容量が 4000 bytes の場合).

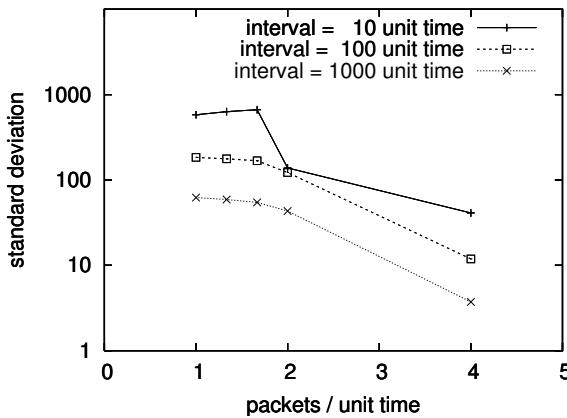


図 3 待ち行列推定の誤差の標準偏差 (容量が 40000 bytes の場合).

行列の推定のずれが大きくなるためと考えられる。実際の通信状況は輻輳がある場合も無い場合もあるため、検知性能の評価実験ではこの両方の標準偏差の値についてそれぞれ実験を行った。

検知の時間区間の長さが長くなると、待ち行列推定の誤差分布の標準偏差の値は小さくなっている。これは標本数が増えることで平均値に収束していくこの種の観測値の性質を反映したものである。

この予備実験から、このパラメータ設定の辺りでは 100 パケット程度の通信が行われればネットワークの状況は充分に安定することが分かった。そこで実験の手間も考え、検知性能の評価実験は検知の時間区間の長さを 100 単位時間にして行うこととした。パケットは上流側ルータのそれぞれから単位時間に一個程度発信されるので、検知の各時間区間ににおいて中間のルータ r に 200 個程度のパケットが届く。パケットロス攻撃の検知にはこの程度のパケット数があれば十分である。またこの時間区間の長さは短いほうが、検知を素早く、また細かく行うことができ、セキュリティ技術として都合が良い。

5. 結果

文献 [1] の検知方式によるパケットロス攻撃の検知性能が輻輳の程度によりどれだけ変わるのがをシミュレーション

を行って調べた。シミュレーションの一回の実行は検知の時間区間 10 個から成り、一つのパラメータ設定につき 10 回の実行を行い、その結果を集計して性能評価とした。

パケット長は全て 1500 bytes と一定にし、上流側の二つのルータから単位時間に約一個送信した。これを中間のルータで集め、下流側のルータへ転送する。その際、パケットロス攻撃として二つの時間区間でそれぞれ一個のパケットを削除した。

検知の時間区間は 100 単位時間とした。中間のルータに届くパケットは検知の時間区間当たり約 200 個となる。待ち行列の大きさは予備実験と同じく 4000 bytes と 40000 bytes の二通りで実験を行った。検知の際に使う待ち行列推定の誤差分布の標準偏差については、予備実験の結果から 12 と 160 の二つの値についてそれぞれ実験を行った。検知に用いる二つの有意水準は文献 [1] と同じ値にした。

$$s_{\text{single}} = 0.999 \quad (13)$$

$$s_{\text{comb}} = 0.9 \quad (14)$$

輻輳は中間のルータの転送処理能力（単位時間当たりに送信できるパケット数）を変えることで発生させた。上流側からは単位時間当たり約二個のパケットが送信されるので、この転送処理能力が単位時間当たり 2 を下回ると輻輳が必ず発生する。

表 1 検知結果 (容量 : 4000 bytes , 標準偏差 : 12.0).

パケット転送能力 (個/単位時間)	1.00	1.33	1.67	2.00	4.00
総パケット数	20030	19896	19972	20052	19961
混雑による廃棄数	10012	6551	3348	750	0
攻撃による削除数	20	20	20	20	20
総区間数	100	100	100	100	100
攻撃検知成功区間数	20	20	20	20	20
攻撃無し判定成功区間数	0	3	5	20	72
誤検知 (濡れ衣) 区間数	80	77	75	60	8
誤検知 (見逃し) 区間数	0	0	0	0	0

表 2 検知結果 (容量 : 40000 bytes , 標準偏差 : 160.0).

パケット転送能力 (個/単位時間)	1.00	1.33	1.67	2.00	4.00
総パケット数	19733	19729	19695	19891	20024
混雑による廃棄数	9718	6384	3016	4	0
攻撃による削除数	20	20	20	20	20
総区間数	100	100	100	100	100
攻撃検知成功区間数	20	20	20	20	20
攻撃無し判定成功区間数	0	0	0	5	78
誤検知 (濡れ衣) 区間数	80	80	80	75	2
誤検知 (見逃し) 区間数	0	0	0	0	0

それぞれの実験結果は表 1, 表 2 のようになった。なお、それぞれのパラメータ設定で標準偏差の値が異なる実

験（容量が 4000 bytes で標準偏差の値を 160.0 としたものと、容量が 40000 bytes で標準偏差の値を 12.0 としたもの）の結果は、それぞれこれらの表と完全に同じものになった。これは今回の性能評価の範囲ではこの程度の標準偏差の値の違いが検知結果に影響しないことを示す。本来であれば、待ち行列推定誤差分布の標準偏差の値が一桁大きくなると、有意水準との比較に用いる信頼値の計算において誤差関数の引数が一桁小さくなり、それに対応して信頼値が小さくなる。そのため検知のための判定が甘くなり、誤検知（見逃し）が増えるはずである。しかし今回の実験結果では全てのシミュレーション実行において誤検知（見逃し）は無かった。この食い違いの解明のためには各判定を詳細に調べる必要がある。

それぞれの検知結果については、まずパケットロス攻撃があった全ての時間区間を正しく攻撃のあった区間と判定し、誤検知（見逃し）は一件も無かった。この結果は待ち行列の容量やパケットの転送処理能力、即ち幅轍の程度に依存しない結果となった。

一方、誤検知（濡れ衣）の区間数はパケットの転送処理能力（単位時間当たりに送信できるパケット数）、が 2 を下回って下がるに従い大きくなつた。その程度は待ち行列の容量が大きいほうが顕著である。即ち幅轍により誤検知（濡れ衣）が発生していることが分かる。特に幅轍が強い（パケット転送能力が小さい）とパケットロス攻撃が無かつた時間区間を全て誤検知（濡れ衣）てしまつてゐる。この部分では、攻撃があった区間も正しく攻撃として検知されている結果から考えると、幅轍が強いため攻撃の有無に係わらず全ての時間区間を攻撃があつたものと判定しているように見える。これを確かめるには、一件々々の判定がどのようになされたのか詳細に調べる必要がある。

幅轍が無く、誤検知（濡れ衣）が少なく抑えられている実験結果からは、幅轍が無いところでもこの検知方式が正しく動作することが分かる。

6. まとめ

本稿では、文献 [1] で提案された待ち行列推定に基づくパケットロス攻撃検知方式について、幅轍の程度に対する検知性能の変化を調べた。その結果、幅轍の有無は検知の際の判定式に用いる待ち行列推定誤差分布の標準偏差の値を変えるがその範囲で標準偏差の値を変えても検知結果が変わらないこと、攻撃のある時間区間の検知は幅轍に関係なく正しく行えること、幅轍が無いと誤検知も少ない一方、幅轍が生じると攻撃の無い時間区間の誤検知（濡れ衣）が多く発生することが分かった。

幅轍が強く全ての時間区間を攻撃ありと判定してしまうこの問題の解明には、それぞれの判定を詳細に調べる必要があり、今後の課題として残る。文献 [3] での実験からは、検知の時間区間の境界において通信の遅延によるパケット

の時間区間の飛び越えが発生し、それが誤検知（濡れ衣）に繋がるという問題点が分かっている。上記の課題点の原因がこの問題点にある場合、その解消には、第 2.3 節で説明したパケット廃棄の検知において隣接する時間区間のパケット情報も利用することや、下流側のルータでの通信観測情報作成の際に待ち行列での遅延も正確に推定するなど、検知方式の大幅な改良が必要になる。

参考文献

- [1] A. Mizrak, S. Savage, and K. Marzullo, “Detecting Malicious Packet Losses”, In IEEE Transactions on Parallel and Distributed Systems, Vol.20, No.2 (February 2009).
- [2] A. Kuzmanovic and E.W. Knightly, “Low-rate TCP-targeted Denial of Service Attacks: the Shrew versus the Mice and Elephants”, Proceedings of ACM SIGCOMM'03, pp.75-86 (August 2003)
- [3] 細井 琢朗, 松浦 幹太, “待ち行列推定に基づくパケットロス攻撃検知方式のパラメータ依存性について”, コンピュータセキュリティシンポジウム 2012 (CSS2012) 論文集, 発表 2B3-1 (CD-ROM) (2012 年 10 月)