Low-rate DoS攻撃を緩和するための代理再送機構の再送タイミング制御

児玉 拓海[1,a)] 久末 瑠紅[1] 稲村 浩[2] 石田 繁巳[2]

概要：サイバー攻撃の DoS 攻撃の 1 種として，通信プロトコルの脆弱性を悪用し通信の品質を低下させる Low-rate DoS（LDoS）攻撃が議論されている．LDoS 攻撃は従来の大量トラフィックを用いる Flooding DoS（FDoS）攻撃とは異なり，低量のトラフィックで攻撃を実現する．LDoS 攻撃手法の 1 つに，TCP の再送制御アルゴリズムを悪用する Shrew 手法がある．Shrew 手法では，TCP の RTO を用いた再送のタイミングと攻撃トラフィックの転送タイミングを同期させることで攻撃を成立させている．攻撃を受ける TCP には変更を加えずに外部ノードを追加し，その代理再送機構を用いて再送タイミングと攻撃トラフィックの転送タイミングの同期を外し，LDoS 攻撃を緩和する手法を提案し実現性を示した．しかし，代理再送タイミングの決定において最適化された LDoS 攻撃のみを想定しており，攻撃パルス幅が変更された場合について十分な検討と対処ができていなかった．本稿では，多様な攻撃パルス幅を持った LDoS 攻撃に対応するために攻撃トラフィックの転送終了を判断し，それ契機に代理再送の開始タイミングを決定する手法を提案する．評価実験を行い，提案手法が従来手法と比較して多様な攻撃パルス幅の条件に対して対処可能であることを示した．

1. はじめに

インターネットの通信を支えるトランスポートプロトコルとして TCP が存在する．TCP は 1981 年に標準化 [1] されてから，現在でも長期的に活躍している．Web の閲覧やメールの送受信など，インターネットで使われる代表的なアプリケーションには，TCP が使用されており暗号化技術と組み合わせることで，高いセキュリティが必要となる通信にも使用されている．TCP は長期的に多くの通信に使用されていることから，TCP の脆弱性に注目し標的とするサイバー攻撃も存在しており，TCP を用いる通信の安全性の向上は必要である．

サイバー攻撃である DoS 攻撃の 1 つとして，Low-rate DoS（LDoS）攻撃が議論されている [2]．LDoS 攻撃は, 通信プロトコルで用いられているアルゴリズムの脆弱性を悪用し, パルス形状の攻撃トラフィックを用いて攻撃を実現する．LDoS 攻撃には，TCP の再送制御で用いる再送タイムアウト（Retransmission Time Out; RTO）の再送タイマ管理アルゴリズムを悪用する Shrew 手法 [2]，Loss-based 輻輳制御アルゴリズムを悪用する Reduction of Quality（RoQ）DoS 手法 [3]，HTTP で用いる KeepAlive のメカニズムを悪用する LoRDAS 手法 [4] などが存在する．

これらの LDoS 攻撃手法はパルス形状の攻撃トラフィックを用いて攻撃を実現しており，平均通信量が低いため，従来の平均通信レートを用いる Flooding DoS（FDoS）攻撃に対する検知機構を回避する攻撃のステルス性を持つ．この特性により，LDoS 攻撃の被害を受けた場合でも，このステルス性によって被害者が攻撃を認知できていないケースも存在する [5] ため，この攻撃に対する耐性を付与することは重要である．

既存研究 [2,6] など，TCP の制御アルゴリズムに変更を加えることで LDoS 攻撃耐性を付与する手法を提案されているが，現在も多くの通信で用いられる TCP の制御アルゴリズムを直接変更することは展開コストが大きいという課題がある．既存の LDoS 攻撃の対策手法について，現行の TCP を変更するコストを避けるために PEP （TCP Performance Enhancement Proxy） [7] を応用した代理再送を用いる手法について，著者らの知る限り議論されていなかった．

そこで著者らは，先行研究 [8] にて攻撃を受ける TCP には変更を加えずに，代理再送機構を実装した外部ノードを追加する新たな LDoS 攻撃緩和手法の初期的検討を行い，攻撃緩和が可能であることを示した．

しかしながら，代理再送の開始タイミングの決定におい

[1] 公立はこだて未来大学大学院 システム情報科学研究科
[2] 公立はこだて未来大学 システム情報科学部
[a)] g2124010@fun.ac.jp

て最適化された LDoS 攻撃パルスのみを想定しており，攻撃パルス幅が変更された場合について十分な検討と対処ができていない．

本稿では，従来手法と比較し多様な攻撃条件に対して対処を可能にするため，代理再送の開始タイミングの変更を提案し，攻撃緩和効果を評価した結果を示す．

本稿の構成は以下のとおりである．まず，2 章にて本稿で扱う Shrew 手法の原理について説明する．3 章では，Shrew 手法の緩和手法に関する関連研究を示し，4 章で提案する代理再送機構を用いた Shrew 手法に対する攻撃緩和手法を説明する．5 章で提案手法の攻撃緩和効果について実験的評価を行い，最後に 6 章にてまとめとする．

2. Shrew 手法の原理

本章では，本稿で扱う LDoS 攻撃である Shrew 手法の原理について述べる．

2.1 DoS 攻撃の概要

DoS 攻撃は，ルータやサーバに攻撃トラフィックを転送することで，通信の妨害や通信品質を低下を発生させるサイバー攻撃の 1 つである．

DoS 攻撃には，大量トラフィックを用いて攻撃する FDoS 攻撃と，低量の攻撃トラフィックを用いる LDoS 攻撃の 2 種類が存在する．FDoS 攻撃は，リンク帯域幅を埋め続けるために攻撃トラフィックを継続して転送するため，平均通信量が非常に大きくなるため検知が容易である．それに対して，通信プロトコルの脆弱性を悪用する LDoS 攻撃には，パルス形状の攻撃トラフィックを用いるため平均通信量が低いという特徴がある．

2.2 LDoS 攻撃の概要

LDoS 攻撃では，パルス形状の攻撃トラフィックを用いて攻撃することで平均通信量を低量にし，攻撃にステルス性を持たせる．LDoS 攻撃に用いられるパルス形状のトラフィックを攻撃長 R，攻撃パルス幅 L，攻撃周期 T として**図 1** に示す．平均通信レートの高さを指標に用いる従来の FDoS 攻撃に対する検知機構では，LDoS 攻撃の検知は難しい [5]．本稿で扱う LDoS 攻撃手法の 1 つである Shrew 手法においてもこれらの特性を持つ．

2.3 TCP の再送タイマ管理アルゴリズム

TCP は，再送タイマを用いて再送処理を時間で制御している．再送タイマでは，送信セグメントに対応する ACK が返送されるまでの待機時間を RTO として設定する．送信セグメントに対応する ACK が返送されるまでの時間が RTO のタイマ値を超えた場合にセグメントを損失したと判断し，再送処理を行う．RTO のタイマ値以内の時間で送信セグメントに対応する ACK が返送された場合，再送タイマ

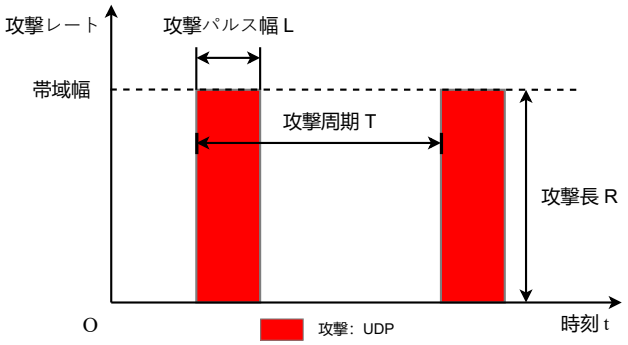

図 1 パルス形状の LDoS 攻撃トラフィック

をリセットする．Fast Retransmit が失敗した後は，RTO を用いたセグメントの再送信が行われる [9]．TCP の再送タイマである RTO は，以下の式 (1) で計算される [10]．

$$\max(minRTO, SRTT + \max(G, 4 \times RTTVAR)) \quad (1)$$

RTO は，$minRTO$ と $SRTT + \max(G, 4 \times RTTVAR)$ の最大値により計算される．$minRTO$ は RTO の初期値のことであり，RFC6298 [10] では 1 秒が推奨値とされている．$SRTT$ は外れ値による影響を軽減するために RTT を平滑化したものである．実環境において多くの場合で $SRTT + \max(G, 4 \times RTTVAR)$ は 1 秒よりも小さい値となる．そのため，式 (1) により計算される RTO は，1 秒が設定され，外部から容易に推測することが可能となっている．

式 (1) で計算される RTO を使用した場合も再送に失敗したとき，指数バックオフを用いる以下の式 (2) で 2 回目以降の RTO が計算される [10]．

$$RTO_n = 2 \times RTO_{n-1},\ RTO_1 = minRTO \quad (2)$$

式 (2) から計算される RTO は，2 回目以降にもパケットのロスの発生した場合，2 倍ずつ増加する．TCP の再送制御では，式 (2) の計算でパケットの送信タイミングを制御することで，より確実なパケットの再送を行っている．しかしながら，RTO の最大値は一般に 60 秒と設定されており，RTO が 60 秒を超えた場合には，TCP はネットワークに異常ありと判断して，コネクションを切断し，セッションタイムアウトとなる．

指数バックオフを用いた再送タイマアルゴリズムの RTO が外部から容易に推測可能である点は，2.4 節で説明する Shrew 手法で悪用されている．

2.4 Shrew 手法

LDoS 攻撃にはいくつかの攻撃手法が存在するが，代表的な攻撃手法として，TCP を標的とする Shrew 手法がある．Shrew 手法では，容易に推測が可能である指数バックオフを用いる再送タイマ管理アルゴリズムを悪用する．Shrew 手法による LDoS 攻撃の原理を**図 2** に示す．Shrew

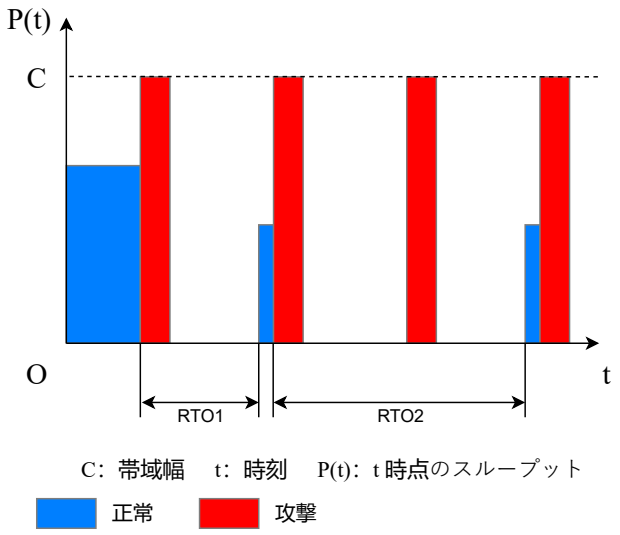

図 2　Shrew 手法の攻撃トラフィック

手法の攻撃成立までの流れは以下のとおりである．まず，送信者は受信者に向けて正常トラフィックの転送を開始する．転送開始後，攻撃者はボトルネックリンク帯域幅を満たすレートで攻撃トラフィックを転送する．転送した攻撃トラフィックはルータにキューイングされ，ルータバッファが攻撃トラフィックによって占有される．これにより，正常トラフィックに含まれているセグメントの損失が発生する．複数セグメントの損失により，送信者による再送タイマを用いた再送処理が行われる．この再送のタイミングは，2.3 節で説明したように予測が容易であるため，攻撃者が攻撃トラフィックの転送タイミングを再送タイミングと同期させることで，再度セグメントを損失させ，さらに RTO を発生させる．このプロセスが複数回繰り返されることで，TCP コネクションのタイムアウトが発生する．

LDoS 攻撃の Shrew 手法では，送信者の再送タイミングと同期するように攻撃トラフィックを転送することで，攻撃を成立させている．このことから，送信者の再送タイミングと攻撃トラフィックの転送タイミングの同期を外すことで，Shrew 手法の攻撃を緩和することができる．

3. 関連研究

Shrew 手法の攻撃対策手法として，代理再送を用いる手法はあまり議論されていない．本章では，これまで議論されている Shrew 手法に対する対策手法について述べ，従来手法の課題を確認する．

3.1 再送タイマ管理アルゴリズムの変更

TCP の再送タイマ管理アルゴリズムに変更を加えることで，Shrew 手法の攻撃を緩和する手法が存在する．

Kuzmanovic らは，RTO 再送と攻撃トラフィックの衝突を回避するために，$minRTO$ を一定の範囲でランダムに選ぶ手法を提案した [2]．しかしながら，$minRTO$ をランダマイズする手法は，TCP の輻輳制御機能を維持するためにランダマイズの幅を大きく取ることができず，攻撃の緩和効果はわずかであることが報告されている．

細井らは LDoS 攻撃に対しての攻撃緩和効果のある RTO 計算アルゴリズムを提案した [6]．提案アルゴリズムでは，RTO の増加方法を式 (3) に変更し，有理数 u を区間 $(0,\ 1)$ の範囲内でランダマイズすることで連続する再送での RTO の値は式 (4) となる．

$$RTO_n = (1+u)RTO_{n-1} \quad (0 < u < 1) \tag{3}$$

$$RTO_n = (1+u)^{n-1} minRTO \tag{4}$$

u に有理数を選ぶことで，RTO 再送の周期が $minRTO$ の整数倍ではなくなるため，RTO 再送と攻撃タイミングの同期が外れる機会が生まれ．LDoS 攻撃の緩和が可能となる．この緩和手法では，従来の RTO の計算アルゴリズムと比較して，LDoS 攻撃による被害を緩和できることが報告されている．しかしながら，この手法では，現行の TCP と $minRTO$ の値は変わらないため，攻撃周期が $minRTO$ の値と等しい場合に，1 回目の RTO 再送は攻撃トラフィックと衝突し失敗する可能性が高い．そのため，攻撃トラフィックとの衝突を回避し攻撃を緩和するまでにパケットの送信が 2 回以上失敗してしまう．

これら 2 つの手法はどちらも攻撃を受ける TCP 自体に変更を加えることが必要であるため，現在多くの通信で用いられている TCP に変更を加えることは展開のコストが大きいことも課題として考えられる．

3.2 最適化された LDoS 攻撃を緩和する代理再送機構

著者らは先行研究 [8] にて，攻撃を受ける TCP には変更を加えずに外部ノードを追加し，その代理再送機構を用いた LDoS 攻撃の緩和手法を提案し実現性を示した [8]．

一般に LDoS 攻撃のトラフィックは UDP トラフィックを用いること，攻撃パルス幅は 0.1–0.3 秒であることから，この手法では，代理再送の開始タイミングを UDP トラフィックの受信により攻撃を検知し，式 (5) で示す λ 秒に再送することで攻撃トラフィックとの衝突を回避し代理再送を行う．

$$\lambda = \frac{1}{3} \cdot minRTO = \frac{1}{3} \qquad minRTO = 1 \tag{5}$$

しかしながら，代理再送の開始タイミングの λ を決定する式 (5) は最適化された LDoS 攻撃パルスのみを想定しており，λ の値が一定となる．そのため，攻撃パルス幅が変更された場合に λ の値を攻撃耐性を持つように更新できず，LDoS 攻撃を緩和することできない．

本稿では，攻撃パルス幅が変更された場合でも攻撃緩和効果を得るために攻撃トラフィックの特性に合わせ，UDP

トラフィックの転送終了を契機に代理再送の開始タイミングを決定する方式を検討する．

4. 代理再送機構による一定タイミングでの再送制御方式の変更

代理再送機構を用いて Shrew 手法の攻撃を緩和するためには，代理再送の開始タイミングが重要である．従来手法では，最適化された LDoS 攻撃のトラフィックのみを想定しており，UDP トラフィックの受信により攻撃を検知し，予測している攻撃トラフィックの転送終了後，即座に代理再送を開始することで攻撃トラフィックとの衝突を回避する．しかしながら，攻撃パルス幅を変更されることによる代理再送トラフィックと攻撃トラフィックの衝突について十分な検討と対処ができていなかったことが課題である．

本稿では，攻撃トラフィックである UDP トラフィックの転送終了を契機に代理再送を開始するタイミングを決定することで，最適化された攻撃パルス幅を持つ LDoS 攻撃以外にも対応可能な代理再送機構を提案する．

4.1 代理再送機構を用いた攻撃緩和手法における代理再送タイミングの重要性

代理再送機構を用いる LDoS 攻撃の緩和には，攻撃トラフィックとの衝突を回避する代理再送タイミングの制御が重要となる．LDoS 攻撃はパルス形状の攻撃トラフィックを用いて攻撃するため，攻撃トラフィックが転送されていない時間が必ず存在する．TCP の RTO を用いる再送処理を悪用する Shrew 手法の攻撃原理から，この時間には攻撃を受ける TCP のセグメント転送は行われない．代理再送機構を用いた攻撃緩和手法では，**図 3** に示す時間で代理再送を成功させることで攻撃によるスループットの低下を緩和する．

代理再送を成功させる上で必要となるのが，攻撃トラフィックとの衝突回避である．攻撃トラフィックとの衝突を回避するために代理再送機構による再送タイミングは，攻撃トラフィックの転送終了後であり，かつ送信者の RTO による再送処理よりも早い時刻である必要がある．これは，図 2 に示した Shrew 手法の攻撃の原理から，攻撃トラフィックの転送タイミングが送信者の RTO による再送処理のタイミングと同期しているからである．したがって，代理再送機構を用いて攻撃緩和効果を得るためには，代理再送トラフィックと攻撃トラフィックとの衝突を回避することが重要となる．

4.2 従来手法より多様な攻撃条件に対処可能な提案手法

従来手法では，LDoS 攻撃で多く用いられる UDP トラフィックの受信により攻撃を検知後，一般に LDoS 攻撃のパルス幅には 0.1-0.3 秒が設定されること [5] から攻撃トラフィックの転送終了時刻を予測し，その時刻に代理再送

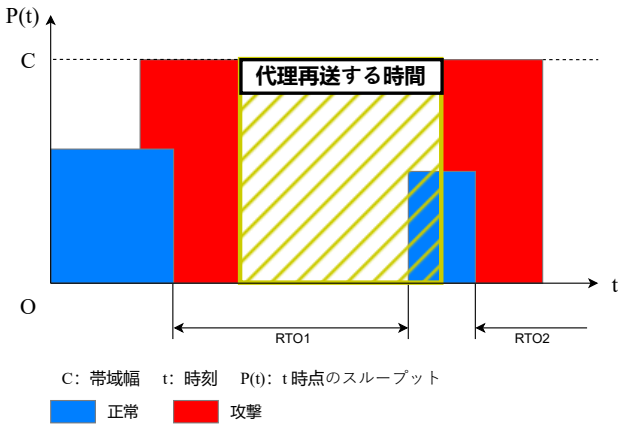

図 3　LDoS 攻撃を緩和するための代理再送を行う時間

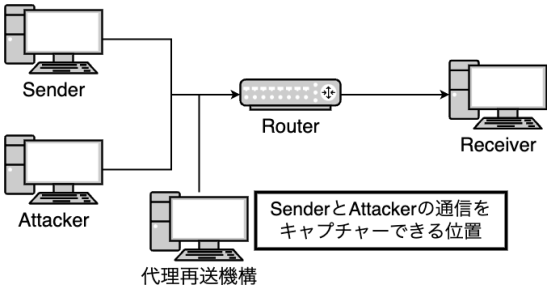

図 4　代理再送機構の設置位置

アルゴリズム 1 パケットキャプチャーとキャッシュ処理

```
Require: cachePacket[]        ▷ パケットをキャッシュするメモリ
Require: cp ⇐ 0               ▷ キャッシュされているパケット数
Require: pcapData             ▷ キャプチャーしたパケット
Require: As ⇐ false           ▷ 攻撃開始のフラグ
Require: At ⇐ 0               ▷ UDP パケットを受信した時刻
 1: function PacketCapture
 2:     if (pcapData = UDP_p) then                    ▷ 条件 1a
 3:         As ⇐ true
 4:         At ⇐ CurrentTime()
 5:     else if (pcapData = TCP_p & As) then          ▷ 条件 1b
 6:         cachePacket[cp] ⇐ pcapData
 7:         cp ⇐ cp + 1
 8:     end if
 9:     ProxyRetransmit()                             ▷ アルゴリズム 2 を実行
10: end function
```

アルゴリズム 2 提案手法の代理再送の開始タイミング制御

```
Require: W ⇐ 100              ▷ 攻撃終了の判断待機時間 (ms)
1: function ProxyRetransmit
2:     if (As & At ≤ CurrentTime() + W) then         ▷ 条件 2
3:         InjectPacket(cachePacket)                 ▷ 関数 2
4:         As ⇐ false
5:         cp ⇐ 0
6:     end if
7: end function
```

を開始することで代理再送と攻撃トラフィックの衝突を回避している．しかしながら，従来手法では攻撃パルス幅を

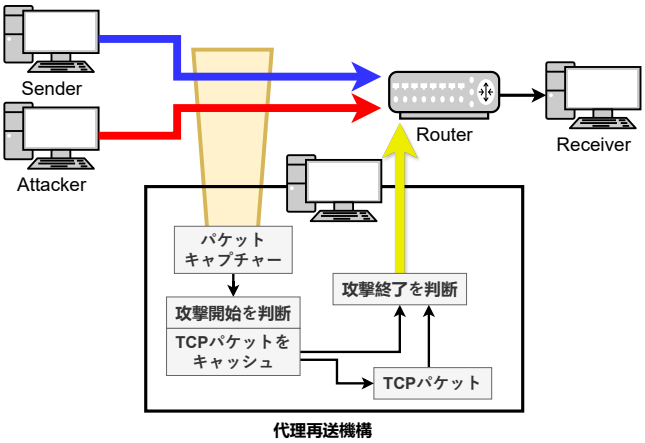

図 5 代理再送機構のシステム構成図

0.3 秒よりも大きく設定された場合でも，代理再送の開始タイミングを変更しないため代理再送のトラフィックと攻撃トラフィックの衝突が発生する．その結果，代理再送が失敗し攻撃緩和効果を得ることができなくなってしまうことが課題である．

そこで本稿では，従来手法と比較して多様な攻撃パルス幅の条件に対して対処を可能とするために，攻撃トラフィックである UDP トラフィックの転送終了を契機に代理再送の開始タイミングを決定する方式を提案する．UDP トラフィックの転送終了を判断してから代理再送の開始タイミングを決定することで，0.3 秒以上の攻撃パルス幅を持つ LDoS 攻撃にも対応可能な代理再送機構を実現する．

本手法では，攻撃トラフィックである UDP トラフィックの転送終了を判断してから代理再送を開始する．本手法の代理再送の開始タイミングの決定方式より，従来手法よりも多様な攻撃パルス幅を持つ LDoS 攻撃に対応可能な代理再送機構を実現し，攻撃緩和性能の向上を目指す．

4.2.1 提案手法の代理再送の開始タイミングの制御

本手法では，多様な攻撃パルス幅の条件に対して対処を可能とするため，攻撃トラフィックである UDP トラフィックの転送終了を契機に代理再送の開始タイミングを決定する．

アルゴリズム 1 とアルゴリズム 2 に，本手法の代理再送機構で実装している代理再送開始までの手順を示す．本手法の代理再送機構は，**図 4** に示す代理再送機構の位置で常にパケットキャプチャーをしており，関数 *PacketCapture* は繰り返し実行される．キャプチャーしたパケットが UDP パケットであった場合，攻撃開始のフラグ *As* を *true* に変更し，現在時刻を UDP パケットの受信時刻 *At* として記録する（条件 1a）．キャプチャーしたパケットが TCP パケットであり，かつ攻撃開始フラグ *As* が *true* の場合は *pcapData* を *cachePacket* にキャッシュする（条件 1b）．本手法では，攻撃トラフィックである UDP パケットの受信時刻 *At* から，次の UDP パケットを受信するまでの時間から，攻撃トラフィックの転送終了を判断する．実装では，関数 *ProxyRetransmit* で UDP パケットを一度受信してから，W ミリ秒の間に次の UDP パケットを受信しなかった場合に攻撃トラフィックの転送が終了したと判断し，キャッシュしている全ての TCP パケットの代理再送を開始する（条件 2）．

本手法の攻撃トラフィックである UDP トラフィックの転送終了を契機に代理再送の開始タイミングを決定する方式により，代理再送トラフィックと攻撃トラフィックの衝突を回避し，0.3 秒以上の攻撃パルス幅を持つ LDoS 攻撃にも対応可能な代理再送機構を実現する．

4.2.2 代理再送機構の実装

図 5 に示すシステム構成図ように代理再送機構は，TCP コネクションにおけるエンドポイントにならず動作する透過型 PEP [7] として実装する．そのため代理再送機構では，代理再送をするときにイーサネットフレーム内に含まれる送信元 MAC アドレスを元々の送信者のアドレスから変更しない．これは，代理再送したパケットに対応する ACK が元々の送信者に返送されるようにするためである．

本手法で用いる代理再送機構は，C 言語で実装する．当初はシングルスレッドでパケットキャプチャーと TCP パケットのキャッシュ，パケットの代理再送をおこなう実装であったが，より正確なタイミングで代理再送を可能とするために実装のマルチスレッド化をおこなう．代理再送機構のパケットキャプチャー，代理再送の処理には libpcap のライブラリ[*1]を使用する．

5. 評価

ここでは，本稿で実装した代理再送機構が従来手法と比較して多様な攻撃条件に対して対処が可能かであるかを検証するために実施した実験的評価とその結果，考察について述べる．

5.1 評価環境

実験の使用機材を**表 1**，使用した評価環境のトポロジを**図 6** に示す．Sender，Receiver，3 台の Attacker の各ノードで用いたプロトコルを**表 2** に示す．Sender と 3 台の Attacker は帯域幅 1 Gbps で Router に接続している．Router はボトルネックリンクで Receiver に接続しており，Sender からのデータを Receiver に向けて転送する．ボトルネックリンクには，通信されるトラフィックを監視する Observer を接続している．Observer は Linux tcpdump のコマンドを使用して，pcap データを取得する．Router は Linux tc コマンドを使用して帯域幅を 10 Mbps，ルータのキューサイズを 300 パケットに設定し，帯域制限をかけて

[*1] TCPDUMP & LIBPCAP, `https://www.tcpdump.org`, アクセス日付: 2024 年 5 月 13 日

表 1　実験で用いた機材

ノード	OS	CPU
Sender	Raspberry Pi OS	ARM Cortex-A72
Receiver	Raspberry Pi OS	ARM Cortex-A72
Attacker1	Raspberry Pi OS	ARM Cortex-A72
Attacker2	Raspberry Pi OS	ARM Cortex-A72
Attacker3	Raspberry Pi OS	ARM Cortex-A72
Router	OpenWRT	Intel(R)N100
Observer	Ubuntu	Intel(R)N100
代理再送機構 (PEP)	Ubuntu	Intel(R)N100

表 2　各ノードで用いたプロトコル

ノード	ネットワーク層	トランスポート層
Sender	IP	TCP
Receiver	IP	TCP
Attacker1	IP	UDP
Attacker2	IP	UDP
Attacker3	IP	UDP

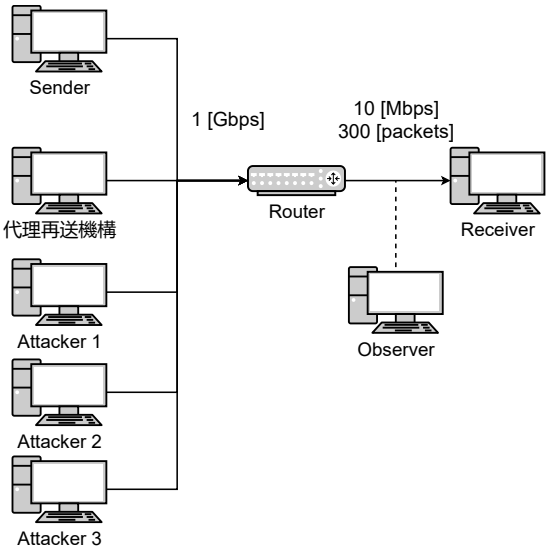

図 6　実験環境のトポロジ

ボトルネックリンクを作成している．

3 台の Attacker は Router に対して，パルス形状の攻撃トラフィックを転送する．LDoS 攻撃で用いられる攻撃トラフィックは多くは UDP パケットである [5] ことから，Attacker の 3 台が送信する攻撃トラフィックは UDP パケットとする．RFC6298 [10] で *minRTO* の推奨値が 1 秒であると定義され，1 回目の RTO のタイマ値は多くの場合で *minRTO* が設定されることから Attacker の攻撃周期 1.0 秒に設定する．攻撃パルス幅の設定は，0.3 秒と 0.4 秒の 2 パターンを評価で用いる．攻撃トラフィックの転送は Sender の送信が終わるまで続けるように設定する．

今回の評価実験では，Sender から Receiver に向けて 10 MB のデータを転送する．Sender のデータ送信開始から送信終了を 1 試行とし，2 パターンの攻撃条件を設定したときの従来手法の導入時，提案手法の導入時，代理再送機構の導入前で，それぞれ 50 試行分の実験の pcap データを，Observer で取得する．取得した pcap データから，平均スループットを算出して攻撃効果を求め，攻撃効果から改善率を計算する．

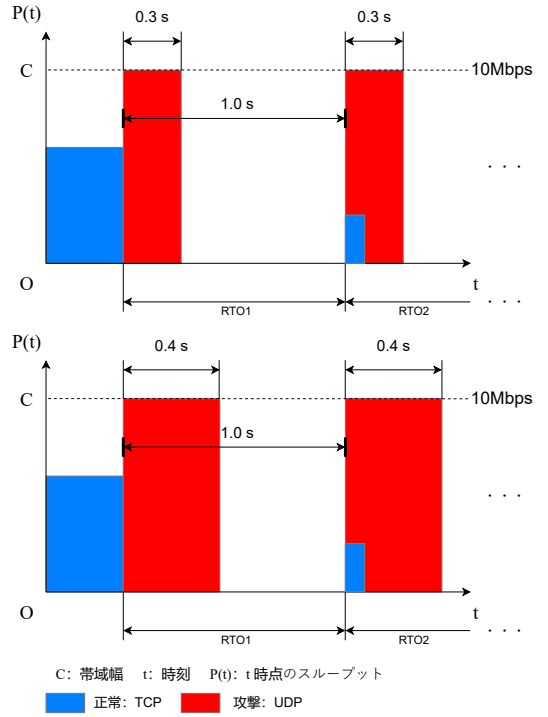

図 7　評価実験に用いる LDoS 攻撃トラフィック
[上：攻撃パルス幅 0.3 秒，下：攻撃パルス幅 0.4 秒]

5.2　評価方法

今回の評価では，2 パターンの LDoS 攻撃条件を設定して従来手法と提案手法のそれぞれの攻撃条件下における攻撃緩和効果を比較し，提案手法が多様な攻撃条件に対応可能かを確認する．**図 7** に評価実験で用いる 2 パターンの LDoS 攻撃トラフィックを示す．実験で用いた LDoS 攻撃は，最適化された LDoS 攻撃である攻撃周期を 1.0 秒，攻撃パルス幅を 0.3 秒に設定したパターンと，攻撃パルス幅に最適化された LDoS 攻撃よりも大きい値である 0.4 秒を設定したパターンである．

5.3　攻撃効果と攻撃の改善率の定義

久末らは，攻撃がない状態の正常トラフィックの平均スループットを T_n ，攻撃下での平均スループットを T_a とし，式 (6) から LDoS 攻撃の攻撃効果を計算している [11]．本稿では，式 (7) から攻撃効果を計算し，評価に用いる．

攻撃下で代理再送機構の導入時の平均スループットを T_p とし，攻撃効果 E_a, E_p を式 (7) を用いて計算する．攻撃効果 E_a, E_p から攻撃の緩和率 R を式 (8) を用いて計算する．

$$E_a = 1 - \frac{T_a}{T_n} \tag{6}$$

$$E_p = 1 - \frac{T_p}{T_n} \tag{7}$$

$$R = \frac{E_a - E_p}{E_a} \tag{8}$$

攻撃効果 E_a, E_p は攻撃による正常トラフィックのスループット低下率を示しており，改善率 R は，代理再送機構の導入前と導入時の攻撃効果から算出した攻撃効果の変化率であり，正常トラフィックのスループット改善率を示している．

5.4 従来の再送タイミングとの比較結果

攻撃パルス幅を 0.3 秒と 0.4 秒に設定したそれぞれの攻撃条件下において，従来手法と提案手法の攻撃緩和効果を比較し，提案手法が従来手法に比べて多様な攻撃条件に対処可能であるかを評価した．

表 3 に攻撃パルス幅を 0.3 秒に設定した場合におけるの従来手法の導入時と提案手法の導入時，代理再送機構の導入前の平均スループット，攻撃効果 E，改善率 R を示す．攻撃パルス幅を 0.3 秒に設定した場合，従来手法では平均スループットは 2.12 Mbps，攻撃効果は 77.9%, 改善率は 11.2%であった．それに対して，提案手法では平均スループットは 3.11 Mbps，攻撃効果は 67.5%, 改善率は 23.0%であった．攻撃パルス幅を 0.3 秒に設定した場合，従来手法と比較すると提案手法の方が高い改善率が得られた．

表 4 に，攻撃パルス幅を 0.4 秒に設定した場合におけるの従来手法の導入時と提案手法の導入時，代理再送機構の導入前の平均スループット，攻撃効果 E，改善率 R を示す．攻撃パルス幅を 0.4 秒に設定した場合，従来手法では平均スループットは 1.13 Mbps，攻撃効果は 88.2%, 改善率は 5.1%であり，攻撃パルス幅を 0.3 秒に設定した場合と比較すると改善率が低下した．それに対して，提案手法では平均スループットは 2.56 Mbps，攻撃効果は 73.3%, 改善率は 21.1%であった．攻撃パルス幅を 0.4 秒に設定した条件下での提案手法の導入時には，攻撃パルス幅を 0.3 秒に設定した場合と比較すると改善率が低下していた．

今回の評価実験では，攻撃パルス幅を 0.3 秒に設定した場合と，0.4 秒に設定した場合の両方で，従来手法よりも提案手法の方が高い改善率が得られた．

5.5 考察

評価実験の結果から，従来手法の最適化された LDoS 攻撃トラフィックのパラメータから一律の代理再送の開始タイミングを決定する方式では，先行研究と同様に攻撃パルス幅を 0.3 秒に設定した条件下では，代理再送機構による攻撃緩和効果が確認できた．しかしながら，攻撃パルス幅を 0.4 に設定した条件下では，攻撃パルス幅を 0.3 秒に設定したときと比較すると攻撃緩和効果が小さいことが確認された．これは，著者らが従来手法の課題として挙げていた代理再送トラフィックと攻撃トラフィックの衝突が原因であると考えられる．

それに対して，提案手法の攻撃トラフィックの終了を契機に代理再送を開始する方式では，攻撃パルス幅を 0.3 秒，0.4 に設定したどちらの条件下においても，従来手法よりも大きい攻撃緩和効果を確認することができた．攻撃パルス幅を 0.3 秒に設定した場合にも，従来手法より大きい攻撃緩和効果を得ることができた理由として，従来手法では代理再送トラフィックと攻撃トラフィックの衝突が発生している場合が存在するが，提案手法では代理再送トラフィックと攻撃トラフィックの衝突が発生していないことが考えられる．攻撃パルス幅を 0.4 秒に設定した場合の結果から，提案手法による代理再送の開始タイミングの決定方式により，攻撃パルス幅が変更されても代理再送トラフィックと攻撃トラフィックの衝突が回避できていると考えられる．しかしながら，代理再送機構を導入した場合の攻撃緩和効果は，著者らの想定よりも小さかった．この要因として，代理再送トラフィックがルータバッファを占有してしまい Sender による通信を妨害してしまっている可能性が考えられる．これを解決するために，代理再送のトラフィック量を調整する必要があると考えられる．

評価実験の結果から，提案手法の攻撃トラフィックの転送終了を契機として代理再送の開始タイミングを決定する方式は，従来手法と比較して多様な攻撃パルス幅の条件に対応可能であることが確認できた．

表 3 平均スループットと攻撃効果と改善率 [攻撃パルス幅:300ms]

攻撃	PEP	Throughput (Mbps)	攻撃効果 E (%)	改善率 R (%)
なし	なし	9.58	0.0	0.0
あり	なし	1.18	87.7	0.0
あり	従来手法 [8]	2.12	77.9	11.2
あり	提案手法	3.11	67.5	23.0

表 4 平均スループットと攻撃効果と改善率 [攻撃パルス幅:400ms]

攻撃	PEP	Throughput (Mbps)	攻撃効果 E (%)	改善率 R (%)
なし	なし	9.58	0.0	0.0
あり	なし	0.68	92.9	0.0
あり	従来手法 [8]	1.13	88.2	5.1
あり	提案手法	2.56	73.3	21.1

6. おわりに

本稿では，多様な攻撃パルス幅を持った LDoS 攻撃に対応するために攻撃トラフィックの転送終了を契機に代理再送の開始タイミングを決定する手法を提案した．提案手法が従来手法と比較して多様な攻撃パルス幅の LDoS 攻撃に対しても攻撃緩和効果を得られるかを評価するために，実機を用いたテストベッドネットワークにて評価実験をおこなった．評価実験では，攻撃パルス幅を 0.3 秒に設定した LDoS 攻撃と 0.4 秒に設定した LDoS 攻撃を用いて，それぞれの攻撃下における従来手法と提案手法の攻撃緩和効果を算出した．今回の評価実験で用いた 2 パターンの LDoS 攻撃条件下において，提案手法を導入することで正常トラフィックのスループットを約 20%改善することができた．評価結果から，従来手法に比べて提案手法の方が大きい攻撃緩和効果が得られることと，多様な攻撃パルス幅を持った LDoS 攻撃に対応可能であることを示した．

参考文献

[1] Postel, J.: Transmission Control Protocol, RFC 793 (1981).

[2] Kuzmanovic, A. and Knightly, E. W.: Low-rate TCP-targeted denial of service attacks: the shrew vs. the mice and elephants, *Proceedings of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, pp. 75–86 (2003).

[3] Guirguis, M., Bestavros, A. and Matta, I.: Exploiting the transients of adaptation for RoQ attacks on Internet resources, *Proceedings of the 12th IEEE International Conference on Network Protocols (ICNP)*, pp. 184–195 (2004).

[4] Adi, E., Baig, Z., Lam, C. P. and Hingston, P.: Low-Rate Denial-of-Service Attacks against HTTP/2 Services, *Proceedings of the 5th International Conference on IT Convergence and Security (ICITCS)*, pp. 1–5 (2015).

[5] Zhijun, W., Wenjing, L., Liang, L. and Meng, Y.: Low-rate DoS attacks, detection, defense, and challenges: A survey, *IEEE Access*, Vol. 8, pp. 43920–43943 (2020).

[6] 細井琢朗，松浦幹太：TCP 再送信タイマ管理の変更による低量 DoS 攻撃被害の緩和効果，コンピュータセキュリティシンポジウム 2013 論文集， Vol. 2013, No. 4, pp. 957–964 (2013).

[7] Griner, J., Border, J., Kojo, M., Shelby, Z. D. and Montenegro, G.: Performance Enhancing Proxies Intended to Mitigate Link-Related Degradations, RFC 3135 (2001).

[8] 児玉拓海，久末瑠紅，稲村　浩，石田繁巳：Low-rate DoS 攻撃の緩和のための代理再送機構の実現性の検討，情報処理学会第 86 回全国大会講演論文集，pp. 3:131–132 (2024).

[9] タネンバウムアンドリューＳ，ウエザロールデイビッドＪ：コンピュータネットワーク 第 5 版，日経 BP 社 (2013).

[10] Paxson, V., Allman, M., Chu, J. and Sargent, M.: RFC 6298: Computing TCP's retransmission timer (2011).

[11] 久末瑠紅，稲村　浩，石田繁巳：攻撃タイミングの誤差を許容する TCP 短時間転送向け Low-rate DoS 攻撃の提案と評価，情報処理学会論文誌， Vol. 65, No. 2, pp. 563–574 (2024).