クラウドコンピューティングにおける監視用トラフィックに対するLow-rate DoS攻撃の可能性

久末 瑠紅[1,a)]　野上 竜杜[2]　稲村 浩[2]　石田 繁巳[2]

概要：検知機構を回避し攻撃を実現する特性をもつ DoS 攻撃として LDoS (Low-rate DoS) 攻撃がある．筆者らはこれまでに，周期性をもつ短時間転送トラフィックに対し攻撃タイミングの推定を行い，推定値に誤差がある場合でも成功確率が向上する LDoS 攻撃の実現可能性を示した．クラウドコンピューティングにおいて，システムの健全性を確認するために周期的なトラフィックを用いた監視が行われるが，この監視用トラフィックが攻撃対象になる可能性が存在する．本稿では，この監視トラフィックに対するタイミング予測に基づく自動化された LDoS 攻撃手法を提案する．攻撃成功に必要な判断ロジックは，外乱トラフィックにより発生する推定タイミング誤差の評価と，提案シナリオにおける攻撃成功要因の分析に基づく．さらに，提案攻撃手法に対する緩和策についても議論する．

1. はじめに

オンプレミスと比較し，サービスのデプロイが容易であることや初期コストが低いことから，クラウドコンピューティングサービスの需要は増加してしている [1]．クラウドサービス事業者においては SLA (Service level agreement) を担保することが求められており [2]，クラウド環境が正常か監視し続けるオブザーバビリティが重要である．オブザーバビリティ技術は，クラウド事業者のコストを削減しつつクラウド利用者への品質担保を実現しているため，信頼性が重要となる．

オブザーバビリティ項目の 1 つに，システムの機器類やネットワークなどが正常に動いているかを判別する死活監視がある．死活監視は，ロードバランサ等が提供する機能の 1 つであり，定期的に通信を行いサーバからのレスポンスの有無でサーバの死活状況を確認する．通信が一定回数タイムアウトした際にサーバが機能していないと判断し，トラフィックの振り分けを停止するなどの処理を行う．

死活監視で用いる監視トラフィックの 1 つとして TCP の 3WHS (3-way hand shake) があり，監視対応に周期的に接続し 3WHS の応答を確認することで監視対象の正常動作を判断する．例えば，GCP (Google Cloud Platform) では監視対象に対して 5 秒ごとに 3WHS の処理を実施することでシステムが正常に動作していることを周期的に監視している [3]．しかしながら，監視に用いるトラフィックの送信タイミングに周期性があり外部から予測し易いことから，TCP の 3WHS を用いる監視手法では TCP の脆弱性を悪用し攻撃される可能性が存在する．

2003 年から，検知機構を回避し攻撃を実現する特性をもつ新たな DoS 攻撃として LDoS (Low-rate DoS) 攻撃がある [4,5]．LDoS 攻撃はパルス形状のトラフィックを用いることで，大量トラフィックを用いて攻撃する従来の FDoS (Flooding DoS) と比較し平均帯域使用率が低く，ネットワークベース FDoS 攻撃検知機構による検知を回避する攻撃のステルス性をもつ．このステルス性によって，LDoS 攻撃を受けた場合でも被害者が攻撃を認知できないケースが存在する [5]．

著者らは先行研究 [6] にて，短時間転送向け LDoS 攻撃として Fawe-Shrew (First-attack pulse width expansion Shrew) 手法を提案し，攻撃対象の通信開始タイミングに誤差が含まれる場合においても LDoS 攻撃が可能であることを示している．Fawe-Shrew 手法は，攻撃の初期パルスのみ幅を延伸することで，攻撃トラフィックが発生させる輻輳により攻撃対象トラフィックが通信阻害を引き起こすためのマージンを確保する．これにより，従来手法と比較し推定に含まれる誤差の許容範囲が大きい．

3WHS 処理は，セッションタイムアウトによる接続の再確立時に検出することが可能であり，実際にアクティブセッションハイジャックと呼ばれるサイバー攻撃手法では

[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
Grad. Sch. Systems Information Science, Future Univ. Hakodate
[2] 公立はこだて未来大学 システム情報科学部
Sch. Systems Information Science, Future Univ. Hakodate
[a)] g3124007@fun.ac.jp

この特性を利用して攻撃を実現している [7]．さらに，周期性をもつトラフィックに対して攻撃タイミングの推定を行い，攻撃タイミングに誤差がある場合でも Fawe-Shrew 手法を用いることで短時間転送に対し攻撃の成功確率が向上することを示した [8]．

本稿では，この監視トラフィックに対するタイミング予測に基づく自動化された LDoS 攻撃手法を提案する．攻撃成功に必要な判断ロジックは，外乱トラフィックにより発生する推定タイミング誤差の評価と，提案シナリオにおける攻撃成功要因の分析に基づく．さらに，提案攻撃手法に対する緩和策についても議論する．本研究のコントリビューションは次の 2 点である：

- 外乱トラフィックにより発生する推定タイミング誤差の評価と，提案シナリオにおける攻撃成功要因の分析に基づく LDoS 攻撃手法を提案し，攻撃の成功による異常判定回数が増加したことを確認した．
- 監視トラフィックから周期性を排除し，疑似乱数を用いたランダマイズを加えることにより提案シナリオへの攻撃緩和効果を示した．

本稿の構成は次のとおりである．1 章にて背景と目的を示した．2 章では，TCP に対する LDoS 攻撃の研究を示し，本研究の位置付けを示す．3 章では，提案手法の該当シナリオを示すと共に，攻撃の原理を述べる．4 章では，提案手法による攻撃効果と緩和手法について評価し，その結果について議論する．最後に，5 章にてまとめとする．

2. 関連研究

本章では，まず TCP に対する LDoS 攻撃手法を紹介し，本研究で扱うシナリオにおいて必要な要素を示す．次に，示した要素のうち既存研究で示されている要素と，本研究で取り扱う要素を述べ，本研究の位置付けを明らかにする．

2.1 TCP に対する Low-rate DoS 攻撃の原理

TCP が再送制御で用いる再送タイマ管理アルゴリズムがもつ周期性を悪用し LDoS 攻撃として Shrew 手法がある．

TCP では，輻輳が起きパケットの損失が起きた際に再送を行う．TCP セグメントを送信し，一定時間応答がない場合にパケットの損失が起きたと判定し再送を行う．このとき再送タイマが用いられ，この再送タイマ切れを RTO (Retransmission timeout) と呼ぶ．RTO の初期値は RFC6298 [9] により，次の式で設定される：

$$RTO = \max\{minRTO, SRTT + \max(G, 4 \times RTTVAR)\} \tag{1}$$

ここで，$SRTT$ は平滑化された RTT (Round trip time)，G はオペレーティングシステムに設定されているクロック粒度，$RTTVAR$ は RTT の平均偏差である．

再送に失敗した場合再び再送を行うが，n 回目の RTO の値 RTO_n は，次の式にて定義される：

$$RTO_n = 2 \cdot RTO_{n-1},\ RTO_1 = minRTO \tag{2}$$

RTO の上限値は 60 秒に設定されており，$minRTO = 1$ の場合，$n \geq 7$ となるとタイムアウトが発生し，TCP コネクションが切断される．$minRTO$ は RFC6298 [9] にて 1 秒が推奨されており，RTT は 1 秒に比べ小さい場合が多いため，$minRTO$ が RTO の初期値として用いられることが多い．

Shrew 手法はこの特性を悪用し，1 秒周期で 0.2–0.3 秒ほどの攻撃トラフィックを送信することで再送タイミングでの輻輳を発生させ，TCP セグメントを損失させ続けることでサービスを妨害する [10]．

Shrew 手法において，トラフィック衝突により輻輳を発生させるためには，攻撃トラフィックレートがボトルネックリンク帯域幅よりも大きい必要がある [4, 5]．攻撃トラフィックレートがボトルネックリンク帯域幅よりも小さい場合，リンク帯域幅に余裕があり通常セグメントが通過可能な状態となる．これにより，セグメントに含まれる ACK の値が更新され，RTO による再送制御が発生せず攻撃効果が低くなる．この特性から，Shrew 手法を用いて攻撃する際にはボトルネックリンク帯域幅より高いレートの攻撃トラフィックを用いる．すなわち，クラウドコンピューティングにおける監視用トラフィックに対し LDoS 攻撃を行うためには，ボトルネックリンクの計測と攻撃開始タイミングの推定が課題となる．

2.2 クラウドコンピューティングにおけるボトルネックリンクの計測

クラウドコンピューティングのサービスモデルでは，サービス提供者がテナント (顧客) の必要に応じて仮想マシンを提供する．サーバ上のコンピューティングリソースは仮想マシンを通して分割されるが，ネットワークリソースについてはテナント間で直接共有される形となる．このことから Feng らは，ネットワークリソースがテナント間で共有されるという特性が Shrew 手法に適していると考えた [4]．しかしながら，データセンタネットワーク (DCN; Data center network) において，ネットワークのボトルネックリンク帯域幅は動的であり，一過性のものである．

そこで，送信側仮想マシンを送信先までのフロー経路でグループ化する Loss-based アルゴリズムを採用し，ボトルネックリンク推定が可能であることを示している [4]．中間スイッチバッファを輻輳させるほどフローレートが高い場合，フローパスの論理ホップ数に応じて損失率も単調に増加した．この特性により，同じボトルネックを通過するフローは同じレベルの輻輳が発生するため，バックグラウンドトラフィックの存在にかかわらず，輻輳発生時の損失率に類似する値を記録する．この観測は，どの仮想マシン

が同じスイッチの下に存在しているか，あるいは最も長いノードのフロー経路を共有しているか判断するために使用できる．さらに，中間スイッチバッファで輻輳が発生するほどフローレートが高い場合，フロー経路の論理ホップ数に応じて損失率も単調に増加することができる．この観測結果を用いて，どの仮想マシングループが他の仮想マシングループよりも宛先から遠いかが明らかとなる．スイッチが利用できる最大のバッファサイズは，バーストトラフィックを処理するキャパシティと同義となるため，この値をボトルネックリンク帯域幅として使用できる．

測定した仮想マシングループのフロー経路とボトルネック帯域幅を用いて，クラウド DCN 内で Shrew 手法を実行した．検証の結果，攻撃対象となった仮想マシンのダウンリンクにおける TCP スループット損失率が最大で 83 %上昇したことが示されている．このことから，Feng らの手法によるボトルネックリンク推定は LDoS 攻撃成功に必要な精度を持つことが認められる．

2.3 通信の盗聴による攻撃開始タイミングの推定とその誤差

Duncan らは悪意のある内部者による攻撃によりネットワークの通信が盗聴できるとしている [11]．OSS に対するサプライチェーン攻撃は年々増加傾向にあり [12]，攻撃者が内部に侵入する可能性がある．

推定した通信タイミングと実際の通信タイミングには，ネットワークのジッタや RTT，混雑状況などの要因によって誤差が含まれることが考えられる．短時間転送に対する LDoS 攻撃において，このような誤差は攻撃成功に影響する．そこで先行研究 [6] にて，我々は正確な攻撃タイミングの推定が難しいことを課題とし，初期パルス幅のみを拡大することで攻撃のステルス性を維持し攻撃タイミング推定の許容誤差性能を向上させる Fawe-Shrew 手法を確立した．

Fawe-Shrew 手法を短時間転送への攻撃に用いる際には，初期パルス拡大の設定値を攻撃成功に必要な最小値にする必要がある．これは，初期パルス幅を必要以上に拡大した場合，帯域占有率が上昇し，LDoS 攻撃の特徴の 1 つであるステルス性が損なわれるためである．

そこで先行研究 [8] にて，具体的な攻撃タイミングの推定方法と，ステルス性を維持するための初期パルス幅決定方法について初期的評価を行った．評価の結果，初期パルス幅を拡大することで，トラフィック衝突が生じる範囲が拡大され，攻撃の実現可能性が向上することが示された．しかしながら，先行研究 [8] では外乱トラフィックにより発生する推定タイミング誤差の評価や提案シナリオにおける攻撃成功要因の分析などはしていない．さらに，提案攻撃手法に対する緩和策についても議論されていない．

2.4 研究課題とアプローチ

関連研究から，クラウドコンピューティングにおいてのボトルネックリンク推定は可能であり，通信内容の盗聴による 3WHS のタイミングの推定は可能であると考えられる．特に，通信の周期性により攻撃タイミングの推定がし易い場合には，初期パルス幅の拡大により攻撃開始タイミングにおける許容誤差性能の向上が期待できる．しかしながら，攻撃対象に対し周期性推定を用いて LDoS 攻撃手法の具体的な判断ロジックやその緩和策は確立されていない．

そこで本研究では，クラウドコンピューティングサービスなどで提供される死活監視機能が周期性を持つトラフィックを送信する特性を利用し，攻撃タイミングを推定し攻撃を行う LDoS 攻撃手法を確立するとともに，提案手法に対し死活監視機能の周期性を排除することで得られる攻撃緩和効果を示す．

3. 監視トラフィック消失 LDoS 攻撃 (RiPTL) 手法

本稿では，クラウドコンピューティングにおける監視トラフィックに対し，通信の周期性推定を用いた輻輳による攻撃効果を最大化できる新たな LDoS 攻撃として，監視トラフィック消失 LDoS 攻撃 (RiPTL; Ridding probe traffic LDoS attack) 手法を提案する．

3.1 攻撃の前提条件

提案手法の実現には通信内容の傍受が前提となっている．近年需要が急増しているクラウドコンピューティングサービスにおいて [1]，Duncan らは悪意のある内部者による攻撃によりネットワークの通信が盗聴できるとしている [11] 特に 3WHS 処理は，セッションタイムアウトによる接続の再確立時に検出することが可能であるため，攻撃に悪用される可能性が高い [7]．OSS に対するサプライチェーン攻撃は年々増加傾向にあり [12]，攻撃者が内部に侵入する可能性も高くなっていくと考えられる．さらに，監視トラフィックで用いる 3WHS の TCP ヘッダは，暗号化されないため情報の収集が可能であり，攻撃対象である監視トラフィックの識別に使用可能であると考えられる．これらのことより，通信内容の盗聴による攻撃タイミングの推定は可能であると考えられる．

クラウドコンピューティングサービスアーキテクチャの特性として，同一リージョンのネットワーク内における遅延が小さいことに加え，ネットワークリソースが共有されることが挙げられ [13]，LDoS 攻撃の対象となっている [4]．このことから，ネットワーク遅延等による攻撃タイミングの推定誤差を抑え，攻撃が可能であると考える．

3.2 想定攻撃シナリオ

攻撃対象となる監視トラフィックが通信されるネット

ワークセグメントに配置された機材へマルウェアなどにより侵入し，その機材を攻撃ノードとして利用する．攻撃ノードは，監視トラフィックを識別し，監視トラフィックの通信を妨害できるよう攻撃トラフィックを送信する．

3.3 攻撃アルゴリズム

提案する RiPTL 手法は「監視トラフィック分析機能」と「攻撃パラメータ設定機能」の 2 つにより構成される．まず，監視トラフィック分析機能により，攻撃対象となる監視トラフィックを検知し送信周期を特定する．その後，攻撃パラメータ決定機能にて，ボトルネックリンク帯域幅などの LDoS 攻撃に必要な情報を収集し，攻撃トラフィックモデルを形成するパラメータの値を決定する．最後に，確定した攻撃トラフィックモデルを用いて攻撃を行う．以降では，各機能について詳細に説明する．

3.3.1 監視トラフィック分析機能

監視トラフィック分析機能では，攻撃対象となる監視トラフィックがどのような周期で送信されるかを特定する．

はじめに，攻撃対象の TCP リンクにて通信されているセグメントが，3WHS のみであるセグメントを攻撃対象トラフィックとして識別する．実ネットワーク上では様々な通信がされており，通信の周期性を測定するには攻撃対象トラフィックの識別を行う必要がある．TCP を用いた死活監視機能では，多くは 3WHS 処理の成否によりサーバの異常を判断している [3]．監視用トラフィック以外で 3WHS のみ行う通信は少ないと考え，リンク内に存在する 3WHS のみの通信を攻撃対象トラフィックとする．

次に，識別した攻撃対象トラフィックの通信間隔の測定と，測定結果に含まれる通信タイミング誤差を計測する．3WHS を用いた死活監視では，周期的に監視トラフィックを送信することによって，監視対象の環境が正常動作しているか判別する．この特性に基づき，識別した攻撃対象となる監視トラフィックを複数サンプル観測し，攻撃対象トラフィックの送信周期を特定する．識別した攻撃対象トラフィックの通信間隔には，ネットワークのジッタや RTT，外乱トラフィックによる混雑状態などの要因により，通信タイミングに誤差が生じることが考えられるため，この通信タイミング誤差の分散を計測する．計測回数は先行研究 [8] に基づき 50 回とする．

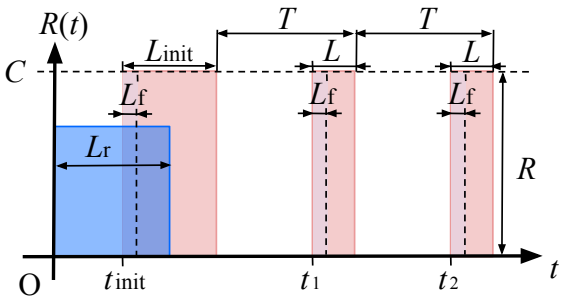

図 1 Fawe-Shrew 手法のトラフィックモデル [6]

3.3.2 攻撃パラメータ設定機能

攻撃パラメータ決定機能では，攻撃で必要となるパラメータを導出し，そのパラメータを用いて攻撃トラフィックのモデルを形成する．

提案する RiPTL 手法は，先行研究 [6] にて示した Fawe-Shrew 手法の攻撃トラフィックモデルおよび攻撃成功条件を基づき攻撃トラフィックを形成する．Fawe-Shrew 手法の攻撃モデルを**図 1** に示す．このモデルは**表 1** に示したパラメータを用いて形成される．

識別した攻撃対象トラフィックの通信間隔を測定することで観測した攻撃対象トラフィックの次の攻撃対象トラフィックの通信タイミングを予測し，通信間隔の分布に基づき初期パルス幅の拡大を行う．初期パルスにより RTO が 1 回以上発生する条件が $-L_{\mathrm{init}} \leq P \leq L_{\mathrm{r}} - L_{\mathrm{f}}$ である [6] ことから，L_{r} を 0 としたとき，1 回以上の RTO を発生させるために必要な初期パルス幅は $L_{\mathrm{init}} + L_{\mathrm{f}}$ となる．分布が $[I_{\mathrm{min}}, I_{\mathrm{max}}]$ となるとき，$L_{\mathrm{init}} = I_{\mathrm{max}} - I_{\mathrm{min}}$ となる．

分布に外れ値などが含まれる場合を想定し，分布のうち 85%が含まれる区間の最大値と最小値の差が最も小さくなる区間を $[I_{\mathrm{min}}, I_{\mathrm{max}}]$ とする．攻撃トラフィックの送信タイミングの誤差とバッファを満たすまでの時間を考慮し，RiPTL 手法における初期パルス幅 L_{init} を $I_{\mathrm{max}} - I_{\mathrm{min}} + L_{\mathrm{f}}$ とする．

スループットは単位時間あたりの通信量で算出することから，初期パルスの拡大が大きすぎる場合に DoS 攻撃検知機構により検知される可能性が高まるため，初期パルス幅には最大値を設け $L_{\mathrm{init}} < 0.9$ とする．

攻撃効果を最大化するために, 観測した通信間隔の分布から最も初期パルスと衝突する範囲が大きくなるよう攻撃タイミングにオフセットをかける．観測された通信間隔を I とし，I が $[I_{\mathrm{min}}, I_{\mathrm{max}}]$ の範囲で分布するとき，攻

表 1 図 1 で用いたパラメータ [6]

パラメータ	記号
攻撃開始時刻	t_{init} [秒]
i 番目のパルスによる攻撃開始時刻	t_i [秒]
攻撃対象トラフィックの転送開始時刻	O [秒]
攻撃なし状態における正規トラフィックの転送時間	L_{r} [秒]
初期パルス幅	L_{init} [秒]
バッファを埋める時間	L_{f} [秒]
後続パルス幅	L [秒]
パルス周期	T [秒]
攻撃パルスレート	R [Mbps]
ボトルネックリンク帯域幅	C [Mbps]

撃タイミングは，RTO が 1 回以上発生する条件である $-L_{\text{init}} \leq P \leq L_{\text{r}} - L_{\text{f}}$ の範囲に入っている必要がある．標的とするトラフィックが 3WHS であることから，通信時間である L_{r} は小さく，攻撃トラフィックによりボトルネックリンクのルータバッファを満たすまでの時間である L_{f} は観測することが難しい．よって，L_{r} および L_{f} を 0 とする．このとき $[I_{\min}, I_{\max}]$ 内で，L_{init} 秒間に収まる I が最も多くなるような区間を求め，観測した攻撃対象トラフィックの送信タイミングからその区間の最小値分待機する．

Attacker は，攻撃対象トラフィックと他アプリケーションの通信を区別するため，特定のポートのパケットを監視し Sender と Receiver 間の通信の SYN フラグを検知する．検知後，Router に対してパルス形状になるように攻撃トラフィックを送信し，1 秒周期で一定時間 Router のキューを占有した状態にする．攻撃効果を最大化するため，観測した通信間隔の分布から最も初期パルスと衝突する範囲が大きくなるよう攻撃タイミングにオフセットをかけ攻撃する．

4. 提案手法による攻撃効果と緩和手法の評価

本稿では，RiPTL 手法の実現可能性とその緩和策について，TCP による死活監視を想定した異常判定の実験的評価を行う．提案シナリオにおける実現可能性について，

1) 外乱トラフィックが混じっている中で 3WHS セグメントを観測し，RiPTL 手法が可能な精度の送信間隔が導出可能か
2) 初期パルス幅拡大のアプローチにより，異常判定回数が増加する要因は何か

の 2 点を評価・分析する．1) について，無作為にデータ転送を行う外乱トラフィックがある中で，3WHS のセグメントを送信し，それに基づく送信間隔の導出を行う．2) について，初期パルス幅拡大のアプローチによる異常判定回数の変化を評価し分析する．その後，異常判定回数の削減に向けた緩和策について述べる．

4.1 実験環境

図 2 に実験で使用した実機によるテストベッドネットワークのトポロジーを示す．Sender と Attacker を Router に接続し，Router からボトルネックリンクで Receiver に接続している．各エンティティで用いた機材を**表 2** に示す．各エンティティの通信プロトコルとして Sender と Receiver 間のデータ転送には TCP，Attacker の攻撃パルスの形成には UDP を用いた．Sender と Receiver 間の往復遅延は約 0.43 ミリ秒であった．minRTO の値は RFC6298 [9] の推奨値である 1 秒に設定した．

Router は Sender が送信したデータを Receiver に向け転送する．Attacker は攻撃トラフィックを送信する．

ボトルネックリンクには，リンク内で通信されるトラフィックを監視するため Observer を設置している．Observer は Linux tcpdump コマンドを用いて，パケットキャプチャ (PCAP) データを取得し評価に用いる．

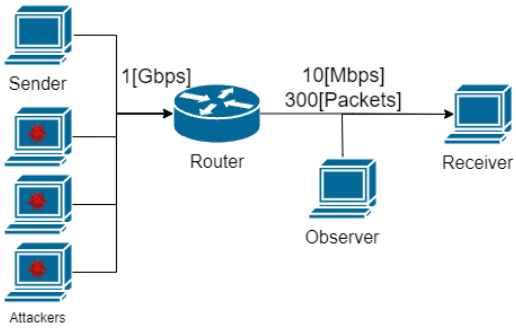

図 2 実験用ネットワークのトポロジー

表 2 各エンティティで用いた機材

エンティティ	OS	CPU
Sender	Raspberry Pi OS	ARM Cortex-A72
Receiver	Raspberry Pi OS	ARM Cortex-A72
Router	OpenWRT	Intel(R) N100
Attacker	Raspberry Pi OS	ARM Cortex-A72
Observer	Ubuntu	Intel(R) N100

ボトルネックリンクを作るため，Linux tc コマンドを用いて仮想的な帯域制限をかけている．帯域幅を 10 Mbps，ルータのキューサイズを 300 パケットに設定した．

4.2 外乱トラフィック混在下における監視トラフィック送信間隔の導出

外乱トラフィックが発生しているときの監視トラフィック分析機能による監視トラフィックの送信間隔が導出可能であるか評価した．

Sender と Receiver の 2 ノードにて，疑似外乱トラフィックを発生させたリンクにて 5 秒周期で監視トラフィックを送信し，リンク内で取得した PCAP データから通信周期の導出を行った．疑似外乱トラフィックとして，Attacker ノード 3 台は Receiver に対し，コネクション確立と 1 MB のデータ転送を 0.1 秒間隔で連続送信させた．

評価の結果，導出した通信間隔が，設定値から 0.5 秒程度の誤差が発生することを確認した ($N = 4000$)．この誤差は，RiPTL における初期パルスの上限値である 0.9 よりも小さいため，提案手法は適用可能であるといえる．

4.3 異常判定回数の変化の要因

提案する RiPTL 手法を用いて監視トラフィックの分析および攻撃トラフィックの形成を行い，攻撃の実験的評価を行った．

4.3.1 評価方法

死活監視機能によりサーバが異常と判定された回数を用いて攻撃効果の評価を行う．

表 3 攻撃パラメータ決定機能により導出された初期パルス幅とトラフィック衝突範発生範囲

分散	初期パルス幅	衝突発生範囲
0.10	0.540	$[-0.540, 0.000]$
0.15	0.811	$[-0.811, 0.000]$
0.20	0.843	$[-0.843, 0.000]$

後続パルス幅は，実験環境においてルータのバッファを満たし，RTO 再送処理によるデータの再送信を引き起こすために十分な 0.3 秒に設定した．

Sender は 5 秒ごとに Receiver に向けて 100 kB のファイル送信を行い，Attacker はそれを観測する．Attacker は Sender の通信を 50 回観測し周期性を特定する．その後，Sender の通信に対し特定した周期を用いて Sender に向けて攻撃パルスを送信する．攻撃開始から Sender の通信が 100 回行われる間に Observer が観測したデータを評価に用いる．この処理を合計 10 試行実施し，各条件において 1000 回の監視トラフィックを送信する．

提案する RiPTL 手法は，監視トラフィックの分析結果に基づき，攻撃パラメータ決定機能により初期パルスの拡大を行う．初期パルス決定のメカニズムが攻撃効果向上に寄与しているか評価するため，Sender の送信周期へ正規分布に従うように誤差を設ける．分散 σ が 0.10，0.15，0.20 の 3 パターンで実験を行う．

タイムアウト時間を連続で超過した回数が閾値を越えたときを異常判定，通信が連続で成功した回数が閾値を超えたときを正常判定と判定する．本実験では，1 回の通信において異常と判定される連続タイムアウト回数の閾値を 2 回，正常と判定される連続 3WHS 成功回数の閾値を 2 回とする．

4.3.2 結果

表 3 に，各条件における攻撃パラメータ決定機能により導出された初期パルス幅とトラフィック衝突範発生範囲を示す．この表から，分散が大きくなるほど，初期パルス幅が拡大されていることがわかる．

各条件における異常判定の発生回数を**表 4** に示す．攻撃パラメータ決定機能により初期パルス幅を拡大することにより，分散 0.10 のとき 2.7 倍，分散 0.15 のとき 2.2 倍，分散 0.20 のとき 2.1 倍まで異常判定回数が増加していることがわかる．

以上の結果から，RiPTL 手法の攻撃パラメータ決定機能は，攻撃対象の分散の大きさに応じて 3.3.2 に示したメカニズムにより初期パルス幅を拡大し，異常判定回数を増加させることがわかる．

これに加えて，いくつかの場合で推定タイミングの値が広く分布していたことが確認された．推定タイミングは，Attacker ノードが観測する正規トラフィック到着時のタイムスタンプに基づき算出されるが，このタイムスタンプはネットワークデバイスの動作状況によって実際の受信時刻と誤差が生じる場合がある [14]．よって，攻撃処理により Attacker ノードの負荷が増加し，攻撃中にネットワーク上のパケット数が増えたことから誤差が出たと考えられる．このことから，Attacker ノードの処理能力が向上するほど，攻撃タイミングが向上する可能性が考えられる．

表 4 各条件における異常判定回数

分散	攻撃パラメータ決定機能	異常判定回数
0.10	不使用	39
	使用 (初期パルス拡大)	105
0.15	不使用	95
	使用 (初期パルス拡大)	211
0.20	不使用	104
	使用 (初期パルス拡大)	217

表 5 監視トラフィックの周期性を排除した際の異常判定回数

攻撃パラメータ決定機能	異常判定回数
不使用	57
使用 (初期パルス拡大)	90

4.4 疑似乱数を用いたランダマイズによる攻撃緩和効果

本シナリオにおいては，攻撃対象である監視トラフィックに周期性があり，攻撃タイミングが予測し易いことが攻撃成功の要因となっている．このことから，周期性を排除することにより提案手法に対する耐性を付与することが可能であると考えられる．そこで，監視トラフィックの送信間隔を 1 秒から 10 秒の範囲で疑似乱数を用いてランダマイズしたうえで，RiPTL 手法による攻撃で異常判定回数に変化が現れるか評価した．

評価結果を**表 5** に示す．分散が 0.10 であり，攻撃パラメータ決定機能を用いていない場合のみを除き，異常判定回数が減少したことが示された．特に，監視トラフィックの分散が 0.2 であるときと比較すると，攻撃パラメータ決定機能を使用した場合には 58.5%，使用していない場合においても 45.2%異常判定回数が削減した．

5. おわりに

本研究では，推定を用いた輻輳による攻撃効果の最大化を実現する予測型 LDoS 攻撃として RiPTL 手法を示した．RiPTL 攻撃は，監視トラフィック分析機能と攻撃パラメータ決定機能の 2 機能で構成されている．外乱トラフィックがある中で監視トラフィック分析機能による周期推定が可能であるか評価したところ，評価に用いたパラメータにおいて設定値からの誤差は 0.5 秒程度であり，攻撃の実現可能性があることを示した．攻撃対象である監視トラフィックの周期性に誤差が含まれる場合に，攻撃パラメータ決定機能による初期パルス幅拡大を実施したところ，誤差が大きくなるほど初期パルス幅を拡大するとともに，異常判定

回数を増加させることがわかった．加えて，死活監視タイミングをランダマイズすることで，提案シナリオに対し緩和効果を得られることを示した．

参考文献

[1] 総務省：平成 28 年版 情報通信白書 第 1 部 特集 IoT・ビッグデータ・AI〜ネットワークとデータが創造する新たな価値〜 (2016). https://www.soumu.go.jp/johotsusintokei/whitepaper/ja/h28/html/nc122320.html.

[2] Mont, M. C., McCorry, K., Papanikolaou, N. and Pearson, S.: Security and privacy governance in cloud computing via SLAS and a policy orchestration service, *Security Governance and SLAs in Cloud Computing*, Vol. 2, SCITEPRESS, pp. 670–674 (2012).

[3] Google: ヘルスチェックの概要 (2023). https://cloud.google.com/load-balancing/docs/health-check-concepts(Accessed on 2024/05/10).

[4] Feng, Z., Bai, B., Zhao, B. and Su, J.: Shrew attack in cloud data center networks, *Proc. the 7th International Conference on Mobile Ad-hoc and Sensor Networks*, pp. 441–445 (2011).

[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] 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).

[7] Baitha, A. K. and Vinod, S.: Session hijacking and prevention technique, *Int. J. Eng. Technol*, Vol. 7, No. 2.6, pp. 193–198 (2018).

[8] 野上竜杜，久末瑠紅，稲村　浩，石田繁巳：攻撃タイミング推定による短時間転送向け LDoS 攻撃手法の提案，第 86 回全国大会講演論文集，Vol. 2024, pp. 3:127–3:128 (2024).

[9] Paxson, V., Allman, M., Chu, J. and Sargent, M.: Computing TCP's retransmission timer, Request for Comments RFC 6298, Internet Engineering Task Force (2011).

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

[11] Duncan, A., Creese, S., Goldsmith, M. and Quinton, J. S.: Cloud Computing: Insider Attacks on Virtual Machines during Migration, *2013 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications*, pp. 493–500 (online), DOI: 10.1109/TrustCom.2013.62 (2013).

[12] Ladisa, P., Ponta, S. E., Sabetta, A., Martinez, M. and Barais, O.: Journey to the Center of Software Supply Chain Attacks, *arXiv preprint arXiv:2304.05200* (2023).

[13] Shieh, A., Kandula, S., Greenberg, A. and Kim, C.: Seawall: Performance Isolation for Cloud Datacenter Networks, *Proceedings of the 2nd USENIX Conference on Hot Topics in Cloud Computing*, HotCloud'10, USA, USENIX Association, p. 1 (2010).

[14] Group, T. T.: PCAP-TSTAMP(7) MAN PAGE (2020). https://www.tcpdump.org/manpages/pcap-tstamp.7.html (Accessed on 2024/05/10).