5ZB-07

gRPC 通信に対する Low-rate DoS 攻撃の試み

久末　瑠紅[†]　稲村　浩[†]　石田　繁巳[†]　中村　嘉隆[‡]
[†]公立はこだて未来大学　[‡]京都橘大学

1. はじめに

TCP に対する新たな攻撃手法として，低レート DoS（LDoS: Low-rate DoS）攻撃が議論されている．LDoS 攻撃とは，サービス拒否攻撃（DoS: Denial of Service）攻撃の一種であり，パルス形状の攻撃トラフィックを用いることで，平均攻撃通信量を低く抑えている．この特徴により，検知されにくい品質低下（RoQ: Reduction of Quality）攻撃が可能である[1].

LDoS 攻撃に関するこれまでの研究では，FTP などを用いた大容量データの送信時に発生する長時間転送を攻撃対象としていた．これに対し，我々は短時間転送を行う遠隔手続き呼び出し（RPC: Remote Procedure Call）に着目する．RPC は，クライアント側で呼び出した関数の処理をサーバー側で実行する仕組みを実現する．本研究では，TCP を用いた RPC の実装である gRPC[2]を取り上げる．gRPC は，マイクロサービス間の通信や，異なるプラットフォーム間のプロセス通信を実現するうえで重要な構成要素となっている[3]．RPC では転送するデータは関数実行に必要となる引数のみであるため，トランザクションサイズが小さい．このような，対話型トランザクションで発生する短時間の転送について，LDoS 攻撃の実現性を考慮している文献は我々の知る限り存在しない．

本研究の目的は，短時間転送に対する LDoS 攻撃の実現性を示すことである．目的達成に向け，gRPC 通信で発生する短時間転送に対する LDoS 攻撃の攻撃効果を報告する．

実現性の評価には，転送データ量を変化させ，4.2 で定義する攻撃効果を用いる．

2. Low-rate Shrew 攻撃手法

本研究では，LDoS 攻撃の中で，RTO 再送を悪用する Low-rate Shrew 攻撃手法[4]の適用を検討する．

2.1. TCP の RTO 再送

パケットロスが発生した際に，TCP はデータを再送信する再送制御を行う．再送制御には，再送タイムアウト（RTO: Re-Transmission Timeout）処理が組み込まれている．高速再転送が失敗した後は RTO によって再送を行う．RTO による再送タイミングは指数バックオフに従う．RTO 動作を規定する RFC6298 で，n 回目の再送開始までの時間は $RTO_n = 2 \cdot RTO_{n-1}$, $RTO_1 = minRTO$ と定義されている．$minRTO$ の推奨値は 1 秒とされている．

2.2. 攻撃手法の原理

Low-rate Shrew 攻撃は，再送タイマーの初期値 $minRTO$ [秒]間隔で攻撃パルスを送信し，経路上のボトルネックルーターのキューを埋め尽くす．これにより，RTO 再送のタイミングでキューに空きがなくなり，パケットロスが発生する．タイムアウト待ちの間は新たな送達確認がない限り，TCP は送信動作を行わず待機する．つまり，待機時間を発生させることでデータ送信の抑制が可能となる．攻撃パルスを繰り返し送信することで攻撃対象の TCP コネクションにおいて送信を連続的に抑制する．

3. アプローチ

我々は，これまであまり議論されていない，転送時間がごく短い TCP コネクションへの LDoS 攻撃を確実に成功させるため，以下に述べる検討を行った．攻撃の成功には，攻撃対象である gRPC 通信の転送開始直後に RTO を確実に発生させることが望ましい．よって本研究では，初期攻撃パルス幅拡大手法を提案する．

2.2 で述べた従来手法では，短いパルス状の攻撃トラフィックを $minRTO$ [秒]間隔で送信していた．しかし，例えば転送時間が $minRTO$ [秒]未満である場合，わずかなタイミングのずれによって RTO が発生しない場合がある．

そこで，提案手法では対象 gRPC 通信の転送開始時点に向けて，最初の攻撃トラフィックのパルス幅のみを拡大させ，確実に RTO を発生させた上で LDoS 攻撃パルスを送信する．具体的には，

An Evaluation of Low-rate DoS Attack to gRPC Communications
Ryuku HISASUE[†], Hiroshi INAMURA[†], Shigemi ISHIDA[†], and Yoshitaka NAKAMURA[‡]
[†]Future University Hakodate
[‡]Kyoto Tachibana University

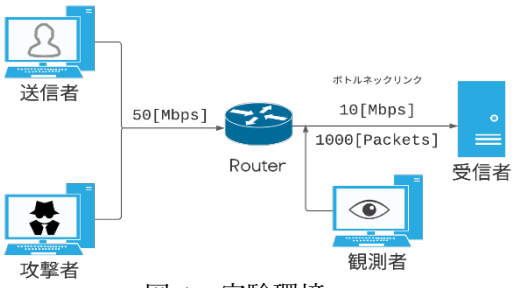

図 1　実験環境

minRTO の値である 1 秒間まで初期攻撃パルス幅を拡大させ，その後一般的な LDoS 攻撃方法に従い *minRTO* の 1 秒間隔で短い攻撃トラフィックを送信し続ける．本手法の有無と，攻撃対象の TCP コネクションにおいて転送データ量を変化させ，短時間転送に対する LDoS 攻撃の実現性を検証する．

4.　実験と評価

4.1. 実験方法

gRPC 通信に対する LDoS 攻撃の実現可能性検証を目的に，図 1 に示す実験環境を構築した．ファイル転送は，gRPC のサーバーストリーム形式で 2 台の Raspberry Pi Model 3B+（Raspberry Pi OS）間で行った．転送には 1–10MB のデータを 1MB 単位で用意した．攻撃ノードについても Raspberry Pi Model 3B+（Raspberry Pi OS）を使用した．構築した環境を用いて，送信者はルーターを経由して受信者にデータを転送した．

Shrew 攻撃の実現には，ルーターにパケットがキューイングされる必要がある．そのため，ルーターの先にあるネットワークがボトルネックリンクとなっており，送信者が送信したパケットがルーターにキューイングされる必要がある．ルーターキューのサイズは 1000 パケットに制限した．

4.2. 評価に用いる攻撃効果の定義

攻撃トラフィックを送信しなかった場合の平均スループット T_{normal} と，送信した場合の平均スループット $T_{onAttack}$ を用いて，攻撃効果 E を

$$E = 1 - \frac{T_{onAttack}}{T_{normal}}$$

とする．

LDoS 攻撃が有効となる有効攻撃効果を $E > 0.4$，理想的な目標攻撃効果を $E > 0.7$ と定義した．

4.3. 実験結果

図 1 の実験環境にて，1–10 MB のデータを転送させ，提案手法と従来手法によって攻撃を行い，転送サイズに対する攻撃効果を図 2 にまとめた．各 200 回転送し，転送データ量ごとに攻撃下でのスループットの平均値を算出した．

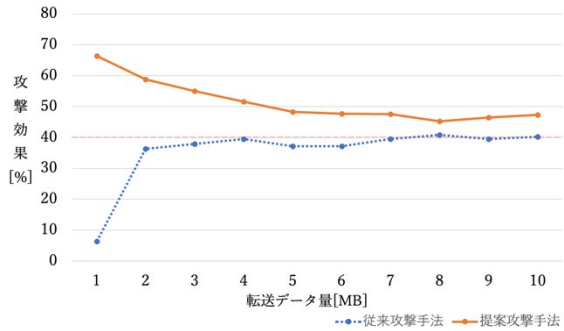

図 2　平均スループットを用いた攻撃効果の推移

2.2 で述べた従来手法では，1–2MB の少量の転送サイズにおいて，十分な攻撃効果が得られなかった．これに対し，提案手法では，転送時間が 1 秒未満である転送データ量 1MB の時点から有効攻撃効果を満した．

転送データ量 1–10MB の範囲において，目標攻撃効果を満たす割合は，従来手法が 4.4%であったが，提案手法は 8.2%と目標達成数が増加した．

以上のことから，提案手法は特に短時間転送に対して有効であり，転送量を増加させると従来手法と同等の振る舞いとなることがわかった．

5.　おわりに

提案手法を用いることで，短時間転送を行う gRPC を用いたファイル転送に対して Low-rate Shrew DoS 攻撃が有効であることを示した．今後の展望として，より多くの攻撃可能性を突き止め，攻撃手法に対する検知・抑止機構の考案を目指す．

6.　参考文献

[1] 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).

[2] gRPC Author.: Introduction to gRPC, https://grpc.io/docs/what-is-grpc/introduction/ (閲覧日：2021/12/22).

[3] 森 直.: gRPC Internal - gRPC の設計と内部実装から見えてくる世界,Wantedly Engineer Blog, https://www.wantedly.com/companies/wantedly/post_articles/219429 (閲覧日：2021/12/22).

[4] 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, SIGCOMM '03, Association for Computing Machinery, pp. 75–86 (2003).