動的なホワイトリストを活用したDoS攻撃検知における誤検知低減手法の提案

片野 諒子[1,a)]　小林 良太郎[1]

概要：近年, 大量のデータを送りつけてサービスを妨害する DoS 攻撃が深刻化しており, 企業や個人にとって重大な脅威の一つとなっている. その対策として, Network-Based Intrusion Prevention System（以下 NIPS）があげられる. 本研究では, 様々なサイバー攻撃に対応するため, 機械学習を用いた NIPS の実装を行う. しかし, 機械学習による NIPS には, 正常な通信を誤って遮断してしまう偽陽性の問題が存在する. 先行研究では, サーバへの負荷に応じて NIDS と NIPS の動作を切り替える動的なフィルタリングシステムが提案されている. 本研究はこの先行手法を拡張し, 同一セッション内で双方向通信が確認された IP アドレスを正常と判断し, サーバへの負荷が低い間にホワイトリストとして登録する. さらに, 負荷が高まった際には, 登録された IP アドレスに対して判定や遮断を行わないことで, 偽陽性の発生を抑制する手法を提案する.

キーワード：ネットワークセキュリティ, NIPS, Dos

A Proposal for Reducing False Positives in DoS Attack Detection Using a Dynamic Whitelist

Ryoko Katano[1,a)]　Ryotaro Kobayashi[1]

Abstract: In recent years, DoS attacks that flood services with massive amounts of data have become increasingly severe, posing a significant threat to both organizations and individuals. The Network-Based Intrusion Prevention System (NIPS) is one of the primary countermeasures against such attacks. This study implements an NIPS using machine learning to address various types of cyberattacks. However, NIPS using machine learning is prone to false positives, which can result in the mistaken blocking of normal communications. Previous studies have proposed a dynamic filtering system that switches between NIDS and NIPS based on network load. Building on this prior approach, our study extends the system by identifying IP addresses involved in bidirectional communication within the same session, considering them normal, and adding them to a whitelist when the network load is low. Furthermore, when the network load increases, the system refrains from inspecting or blocking communications from whitelisted IP addresses, thereby reducing the occurrence of false positives.

Keywords: Network Security, NIPS, DoS

1. はじめに

近年, 単一の発信元から大量の通信を送ることで, サービスを妨害する DoS 攻撃は, その規模と巧妙さが増し続けており, インターネットを基盤とする各種サービスの可用性を深刻に脅かしている. また, その構造的に拡張された形態である DDoS 攻撃は IPA が発表した 2025 年における情報セキュリティ 10 大脅威に挙げられており, これにより DoS 攻撃は企業及び個人 全体で重大な脅威の 1 つとなっている. それらの攻撃を防ぐため, ネットワーク機器に到着するパケットが正常通信であるか悪性通信であるかを判別し

[1] 工学院大学
Kogakuin University, Shinjuku, Tokyo 163-8677, Japan
[a)] j122092@ns.kogakuin.ac.jp

遮断する NIPS は必要不可欠である. 近年では, 機械学習を用いた DDoS 攻撃検知が数多く研究され, 高い有効性が示されている [1],[2]. また, 機械学習を用いた NIDS の手法は多く提案されており, 未知の攻撃手法にも柔軟に対応可能である [3],[4]. この流れを受けて, 機械学習を用いた NIPS が注目されている. しかし, 現状の機械学習を用いた NIDS を NIPS として実装すると, 正常な通信を誤って攻撃と判定されてしまう偽陽性の発生が課題となる. 偽陽性が生じた場合, DDoS 攻撃下ではない平常時であっても通信が不必要に遮断され, ユーザの可用性が損なわれる可能性がある. 本研究はこの課題に着目し, 伊藤氏が提案したネットワークの負荷に応じて動的にフィルタリングする NIPS の先行研究を基盤としている [5]. この研究では, ネットワーク内のトラフィック量を監視し, 一定期間に観測されるパケット数が設定した閾値を下回る場合には NIDS として動作し, 閾値以上の場合には NIPS として動作するよう切り替えるシステムが提案されている. 本研究ではこれを発展させ, 同一セッション内で双方向通信が確認された IP アドレスを正常通信とみなし, ネットワークの負荷が低い間はそれらの IP アドレスをホワイトリストとして追加する. ネットワークの負荷が高まった際には, リストに登録された IP アドレスの判定及び遮断処理は行わない機能を追加することで誤検知を抑制する手法を提案する.

本研究の最終目標は，新しい NIPS の提案・実装・評価であるが，本稿ではその途中経過として主に提案と実装を報告する．本論文の構成は以下の通りである．まず, 第 2 章では関連研究について述べる. 第 3 章では提案システム, 第 4 章では提案システムの実装, 第 5 章において提案システムの 検証内容及びその結果について述べる. それらを基に第 6 章ではまとめについて述べる.

2. 関連研究

2.1 先行研究

本研究では, 先行研究として伊藤らが提案した 2 つの NIPS を基盤とし, それらを拡張する形で新たな防御機構を構築している [5]. 1 つ目の NIPS は, パケット群における悪性パケットの割合が許容閾値を超えた場合にのみ遮断を行う手法が提案されている. 2 つ目の NIPS は, トラフィック超過時に一部のパケットを間引いて選択的に判定を行う手法が提案されている. いずれの手法も, 偽陽性の発生を抑制することを目的としている. 以下では, それぞれの手法の構成と特徴について詳述する.

2.1.1 許容割合を変化させる NIPS

この NIPS では, ネットワークのトラフィック量に応じて, 一定パケット数の中で許容できる悪性パケットの割合を動的に設定する点が特徴である. 具体的に, 一定数のパケットを 1 つの塊として扱い, その中で悪性と判定されたパケットの割合が許容範囲内であれば, 遮断処理を行わない. 一方で, その割合が許容値を超えた場合には, 悪性と判定されたパケットを遮断するという手法である. 低トラフィック量時は高トラフィック量時に比べ, 悪性と判定した割合が大きくとも遮断しにくくなる. これにより, 低トラフィック量時の偽陽性を減少させることができる.

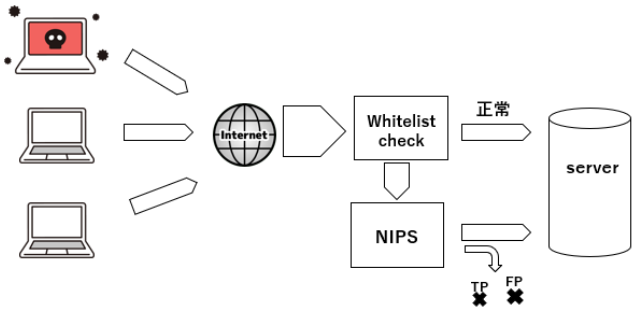

図 1: 提案システム
Fig. 1: Proposed System

2.1.2 判定パケット数を変化させる NIPS

この NIPS では, サーバの耐久性能を考慮しつつ, ネットワークトラフィックの量に応じて判定対象とするパケット数を動的に調整する方式を採用している. トラフィック量が増加した場合, 数パケットに 1 パケットのみを判定対象とし, 判定頻度を抑えることでサーバの耐久性能の限界近くまでパケットを通過させる. このようにして, サーバが処理可能な範囲内のトラフィックについては通信をそのまま許可し, 処理能力の限界を超過した場合に一部のパケットに対して判定を行うことで, 不要な遮断を回避し, 偽陽性を極力減少させることができる.

2.2 本論文の位置づけ

本論文は, 機械学習を用いた NIPS における偽陽性の課題に着目し, その抑制を目的とした手法を提案する. 伊藤らによる攻撃トラフィック量に動的に対応する NIPS の先行研究を基盤に, 正常通信の特徴として同一セッションにおける双方向通信を採用し, 機械学習で正常と判定されたうえで当該条件を満たす相手 IP アドレスをホワイトリストに段階的に登録する仕組みを導入する. これにより, 誤検知による不要な通信遮断を低減することを目指す.

3. 提案システム

本研究では, ネットワーク上のトラフィック量に応じて段階的に検知及び防御方針を切り替える防御システムを提案する. 図 1 に提案システムの概要図を示す. 本システムは, 平常時から高負荷時までを 3 つのフェーズに分割し, それぞれのフェーズにおいて判定対象, 使用する機械学習のモデル, ホワイトリストの管理方針, 遮断の有無を切り替えることで, 機械学習モデルによる誤検知を抑制することを目的とする.

3.1 双方向通信の定義

本研究における双方向性は次のとおり定義する．セッションは, 送信元 IP・送信元ポート・宛先 IP・宛先ポート・プロトコルの 5 要素で区別し, 一定時間通信が途切れた場合は新たなセッションとして扱う．ある方向のパケットを観測した後, 同じ 5 要素を反転させた逆方向のパケットが, 設定した期間内に一度でも観測されたとき, そのセッションで双方向通信が成立したとみなす．なお本システムでは, 逆方向が時間内に観測できたかどうかのみで判定する．

3.2 ホワイトリストの登録条件

登録対象は通信相手の IP アドレスとする．登録は, 判定が正常である, 双方向通信が確認できる, システムがフェーズ 1 または 2 のとき, の三つをすべて満たした場合に実施する．登録の単位は IP アドレスで, 同じ相手との複数セッションは 1 件にまとめる．

3.3 判定機の設定

分類器にはランダムフォレストを用いる．推論ではパケット・セッションの特徴から攻撃である確率を計算し, その値が一定以上なら攻撃, 未満なら正常と判断する．実装では学習時と同じ特徴量の並びと型に揃え, 欠損は 0 で補完する．

3.4 フェーズの切り替え

本研究では, フェーズ切替に用いる負荷指標として指数移動平均 (以下 EMA) を採用する．処理は一定期間に到着パケット数を集計し, その値を重みをつけで滑らかに更新する．初回のみ最新値をそのまま採用する．以降のフェーズ判定は, この滑らかな指標値に基づいて行う．

3.5 フェーズ 1: 平常時

フェーズ 1 は, サーバへの攻撃がなく, サーバの応答速度に異常が見られない平常時の運用状況を想定している．この段階は, 正常とみなせる通信相手の情報を収集し, ホワイトリストの構築を目的とする．そのため, 可能な限り多くの通信を対象とし, すべてのパケットに対して判定を実施する．本段階における判定には, パケット単位の機械学習モデルを用いる．ホワイトリストには, 正常と判定されたうえで同一セッション上における双方向通信が確認された IP アドレスを登録対象とする．本段階はパケットの遮断は行なわず, 観測とデータ取集を優先する．

3.6 フェーズ 2: 軽負荷時

フェーズ 2 は, サーバへの攻撃が観測されているものの, 応答遅延や高負荷といったサーバ側に明確な影響が観測されていない段階を想定している．この段階は, ホワイトリストを用いて正常通信を除外することで, 判定対象を限定し, システム全体の処理負荷を抑えながら, ホワイトリストの構築を目的とする．本段階では, 判定対象をホワイトリストに登録されていない IP 通信に限定する．判定には, 判定精度の向上を目的としてパケット単位及びセッション単位の機械学習モデルを併用する．ホワイトリストには, フェーズ 1 と同様に, 正常と判定されたうえで同一セッション上における双方向通信が確認された IP アドレスを登録対象とする．本段階はフェーズ 1 と同様に, パケットの遮断は行なわず, 観測とデータ取集を優先する．

3.7 フェーズ 3: 高負荷時

フェーズ 3 は, サーバへの攻撃が観測され, 応答遅延といった運用上の支障が観測できる段階を想定している．この段階は, リソース制約下においても最小限の安全性を確保し, 攻撃の影響を抑制することを目的とする．本段階では, 判定対象をホワイトリストに登録されていない IP 通信に限定する。判定には, フェーズ 2 と同様にパケット単位の機械学習モデルを使用する．ホワイトリストの新規登録はこの段階で停止し, フェーズ 1 及び 2 で構築された既存のホワイトリスト情報のみを用いて運用を継続する．また, 先行研究に基づく動的制御方式に対応し，運用時には判定するパケット数または許容割合を変化させる NIPS のいずれか一方を選択して適用する。通信の遮断については, 攻撃と判定された通信に対して実施する．

4. 提案システムの実装

本研究では, サーバに対し DoS 攻撃等の悪性通信のパケットを検知及び遮断する NIPS を提案する．本研究における NIPS は変化し続ける様々なサイバー攻撃に対応できるようにするため, 機械学習を用いて検知を行う．機械学習では偽陽性が発生してしまう．その対処法として, 動的なホワイトリスト機構と, トラフィック量に応じてフィルタリング処理を動的に制御する NIPS を組み合わせた手法を提案する．

表 1: トラフィック量とフェーズ数の関係

Fig. 1: Relationship Between the Amount of Traffic and the Number of Phases

パケット数	判定フェーズ
32000 以下	フェーズ 1
38000 以下	フェーズ 2
38001 以上	フェーズ 3

4.1 NIPS の実装

動的なホワイトリスト機構の各処理と 2 つの NIPS についてそれぞれ述べる．

表 2: トラフィック量と許容する悪性通信の割合の関係
Table 2: Relationship between traffic volume and percentage of malicious traffic allowed

パケット数	許容割合
40000 以下	50%
45000 以下	60%
50000 以下	70%
50001 以上	80%

表 3: トラフィック量と判定するパケット数の関係
Table 3: Relationship Between the Amount of Traffic and the Number of Packets to Be Judged

パケット数	judgeStatus	判定パケット数
40000 以下	0	判定しない
45000 以下	1	1/3 パケット
50000 以下	2	2/3 パケット
50001 以上	3	全パケット

4.1.1 フェーズ処理

本実装では, 0.1 秒間隔で到着パケット数 x_t を集計し, EMA で平滑化した値 ema_t を負荷指標として用いる. 更新式は

$$\mathrm{ema}_t = \alpha\, x_t + (1-\alpha)\,\mathrm{ema}_{t-1} \qquad (1)$$

であり, 平滑化係数は α =0.3 とした. 初回は $ema_0 = x_0$ とする. 実装では, 0.1 秒間ごとのパケット数を PacketCount とし, 平滑化後の値を ema とした. ema の値に基づいて CurrentPhase を決定する. CurrentPhase はシステムの動作フェーズを表す変数である. トラフィック量に応じて変化させるフェーズを表 1 に示す. フェーズが 1 のときは全てのパケットに判定を行い, 正常と判定したパケットの送信元 IP アドレスはホワイトリストに登録を行う. フェーズが 2 のときは, 送信元 IP アドレスがホワイトリストにないパケットのみ判定を行う. 正常と判定したパケットの送信元 IP アドレスはホワイトリストに登録を行う. フェーズが 3 のとき, 送信元 IP アドレスがホワイトリストにないパケットのみ NIPS で処理を行う.

4.1.2 双方向通信とホワイトリスト登録の扱い

本研究では, ホワイトリストへの誤登録を避けるため, 同一セッション上で逆方向通信が短時間内に観測されたかを機械的に確認する. セッションは送信元 IP アドレス, 送信元ポート番号, 送信先 IP アドレス, 送信先ポート番号, プロトコルの 5 要素である 5 タプルで識別し, 各タプルの最終観測時刻を保持する. 逆方向は送信先 IP アドレス, 送信先ポート番号, 送信元 IP アドレス, 送信元ポート番号, プロトコルの並びで定義する. 双方向成立の判定には 10 秒間の観測区間を用い, 現在時刻と逆方向の最終観測時刻の差が 10 秒以内であれば bidir_ok = True とする. ホワイトリストへの登録はフェーズ 1 か 2 のときに限り実施し, 機械学習の判定が正常かつ bidir_ok = True の場合に, 送信元 IP アドレスを登録する. 同一相手との複数セッションは 1 件に集約される. フェーズ 3 では新規登録を行わない.

4.1.3 許容割合を変化させる NIPS

PacketCount100 とは 100 パケットをカウントするカウンタである. MalciousCount とは機械学習により悪性と判定されたパケット数をカウントするカウンタである. BlockState とは 100 パケット中の悪性パケットの割合が許容割合を超えたことを表すフラグである. トラフィック量の決定方法は上記のフェーズ処理と同様である. トラフィック量に応じて悪性パケットの遮断処理を開始する 100 パケット中の悪性通信の割合（許容割合）を決定する. 本研究では動作確認としてトラフィック量と許容割合の関係を表 2 とする. 100 パケット毎に悪性通信と判定した割合を計算し, 100 パケットにおいて悪性通信と判定したパケット数が表 1 より多い場合は, 次の 100 パケット間で悪性と判

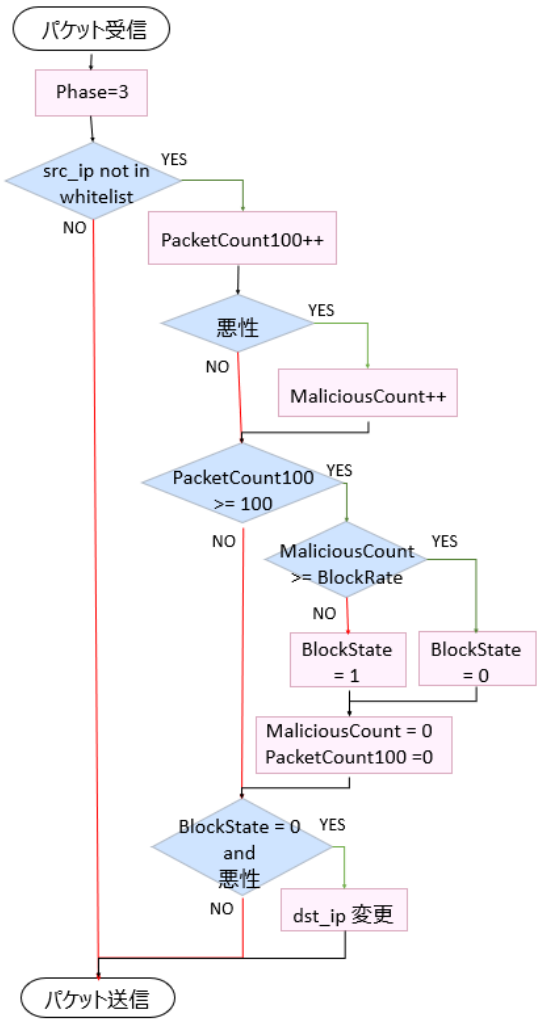

図 2: 許容割合を変化させる NIPS のフローチャート
Fig. 2: Flowchart of NIPS that changes the acceptable ratio

定したパケットを遮断する．遮断する方法として iptables を操作して，遮断を行う．これにより，OS レベルで ip アドレスの遮断ができる．本研究では観測用 PC にて観測できるようにするため，宛先 IP アドレスを変更し，観測用 PC に送信するようにする．これにより，観測用 PC では宛先 IP アドレスを調べることで破棄されたパケットか否かがわかる．

4.1.4 判定パケット数を変化させる NIPS

JudgeStatus とは判定を行うパケット数を表すフラグである．トラフィック量の決定方法は上記のフェーズ制御と同様である．トラフィック量に応じて変化させる判定を行うパケット数を表 3 に示す．本研究では，サーバの耐久能力を 0.1 秒あたり 4 万パケットと仮定する．また，DDoS 攻撃下においては悪性通信が全体の約 7 割を占める状況を想定し，悪性・正常パケットの比率を 7:1 と設定した．さらに，機械学習モデルの判定精度は悪性・正常ともに 8 割と仮定した．これらの値は，先行研究 [5] における条件設定を参考にしている．遮断方法については上記の許容割合を変化させる NIPS と同様である．

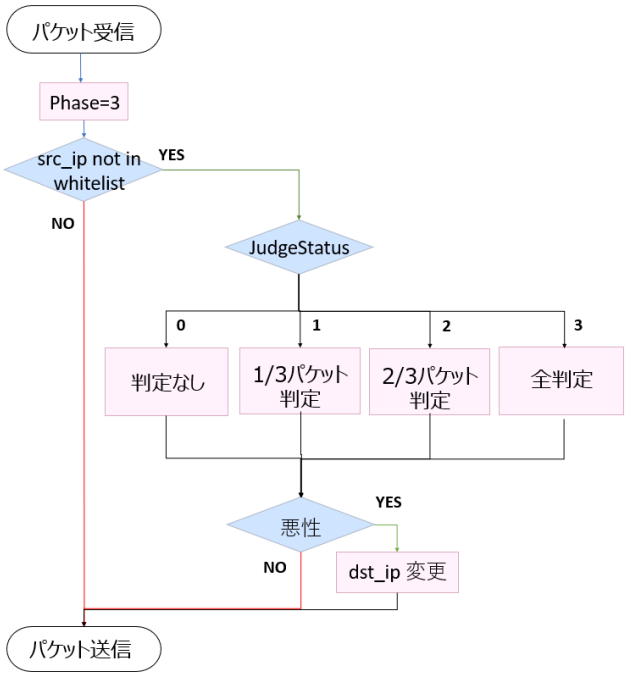

図 3: 判定パケット数を変化させる NIPS のフローチャート

Fig. 3: Flowchart of NIPS that changes the number of packets to be judged

5. 提案システムの検証

本研究では，本システム全体を評価するのではなく，途中経過として，ホワイトリストの登録および攻撃通信の検出・遮断の検証を行い，それ以外は今後の課題とした．

表 4: 実験環境のアドレス割当

機器	インタフェース	アドレス
HostOnly ネット	vboxnet0	192.168.56.1/24
	vboxnet1	192.168.57.1/24
Client (VM)	eth0	192.168.56.101
NIPS (VM)	eth0	192.168.56.102
	eth1	192.168.57.102
Server (VM)	eth0	192.168.57.103

5.1 評価環境

本研究では，提案システムの検証を行うために，仮想環境を構築した．検証環境は，VirtualBox 上に 3 台の仮想マシンを立ち上げ，それぞれに Client, NIPS, Server を配置する．Client から NIPS を経由して Server へ通信を行う．NIPS は Client と Server の間に配置され，Client からのパケットを受信し，機械学習モデルによる判定を行い，Server へパケットを転送する．表 4 は各 VM のネットワーク設定である．

5.2 学習データの概要

学習用データは，外部ネットワーク及びローカル Web サーバへのアクセスから収集した正常通信と，DDoS 系トラフィック生成ツールで生成・取得した攻撃通信から構成する．

5.3 システムの検証方法

実験ではフェーズ境界を低値に設定し機能確認のみ実施した．パケットを 1 分間送信し，検証した．送信元は 192.168.56.111〜192.168.56.120 の 10 個の仮想 IP とし，UDP/53 の DNS には dnsmasq と dig を用い，TCP/80 の HTTP には Apache と curl を用いてアクセスを発生させた．これを正常通信とした．悪性通信は hping3 を用いて作成した．dnsmasq は，軽量な DNS フォワーダおよび DHCP サーバとして利用できるツールである [6]．curl は，HTTP をはじめとする各種プロトコルでデータ転送を行うコマンドラインツールである [7]．hping3 は，任意のパケットを作成し送信できるネットワークツールである [8]．複数送信元の再現には，1 台のクライアント VM 上で Linux network namespace と macvlan を用いた．親 IF 上に macvlan 子 IF を複数生成し，各子 IF を独立の namespace に割り当て，それぞれに異なる送信元 IP を付与した．全通信は NIPS を経由させた．これにより少ないリソースで多数の独立送信元を模擬でき，ホワイトリスト登録の検証に用いた．

5.4 検証項目

本研究では，提案システムの検証を行うために，以下の項目を設定した．

- フェーズ 1 における正常通信のホワイトリスト登録

- フェーズ 1 における安全性確認
- フェーズ 3 におけるホワイトリスト登録抑制・悪性通信の遮断

フェーズ 1 での検証ができれば, フェーズ 2 でも動作することが分かるため, フェーズ 2 での確認は実施していない.

5.4.1 フェーズ 1 における正常通信のホワイトリスト登録

正常通信のみを送信して検証した. TCP は送信元 IP の 10/10 が双方向成立かつ正常判定となり, 全てホワイトリスト登録された. 遮断動作は無し. UDP は送信元 IP の 6/10 がホワイトリスト登録された. 未登録の 4/10 は, 登録条件である双方向成立が観測窓内で満たされなかったためである. 機械学習の判定は TCP/UDP とも全て正常だった. 上記により, フェーズ 1 で正常通信のパケットがホワイトリストに登録されることを確認した.

5.4.2 フェーズ 1 における安全性確認

正常通信と悪性通信を送信して検証した. TCP は正常通信の送信元 IP の 10/10 が双方向成立かつ正常判定となり, 全てホワイトリスト登録された. 悪性通信の送信元 IP は登録されていなかった. UDP は正常通信の送信元 IP すべてがホワイトリストに登録されなかった. 未登録は, 登録条件である双方向成立が観測窓内で満たされなかったためである. 機械学習の判定は全て正常だった. 悪性通信の送信元 IP は登録されていなかった. TCP/UDP ともに遮断処理は行われていなかった. 上記により, フェーズ 1 で悪性通信がホワイトリストに登録されないことと遮断処理が行われないことを確認した.

5.4.3 フェーズ 3 におけるホワイトリスト登録抑制・悪性通信の遮断

正常通信と悪性通信を送信して検証した. ホワイトリスト登録は行われなかった. 遮断処理は行われていた. 上記により, フェーズ 3 でホワイトリスト登録が行われず, 遮断処理が行われることを確認した.

5.5 考察

フェーズ 1 では, TCP の送信元 IP の 10/10 が正常判定かつ双方向通信が成立し, すべてホワイトリスト登録された. 一方, UDP の送信元 IP の 6/10 が登録され, 残りは観測期間内で双方向成立が確認できず未登録となった. フェーズ 1 では遮断処理は行われず, 挙動が保たれている. フェーズ 3 ではホワイトリスト登録は停止され, 攻撃トラフィックを検出・遮断できた. 平時はホワイトリストの構築に集中し, 攻撃時は防御へ切り替えるという本システムの基本設計は確認できた. UDP で未登録が生じた要因は, 登録条件の 1 つである双方向成立が観測期間内に満たされなかった点にある. DNS は要求応答が非常に短く, キャッシュやリトライの影響を受けやすいため, 受信側で応答パケットが一時的に観測されないことがある. そのため, モデルの判定は正常でも双方向条件を満たせず未登録になる場合が生じる. DNS などに対して, 反転 5 タプルではなく IP ペアでの双方向確認を用いる, プロトコルに応じて, 観測期間を変えるなどをする必要があると考える. また, 現状ホワイトリストは一度登録すると登録を解除されることはないため, 誤登録があった場合には修正できない. これに対して, ホワイトリストの登録期間を設定する, 定期的な再判定などを導入し, 誤登録リスクを低減する必要があると考える.

6. まとめ

本論文では, 近年において脅威になっている DoS 攻撃に対して検知と遮断することを目的とし, 動的なホワイトリストとトラフィック量に応じて学習・防御方針を自動で切り替える機械学習ベースの NIPS を提案する. 平常時にはホワイトリストの構築を優先し, 機械学習による正常判定かつ一定期間内の双方向成立を満たす通信相手のみホワイトリストに登録する. 一方, 高負荷・攻撃時はホワイトリスト更新を停止し, ホワイトリスト外の通信を判定・遮断することで, 偽陽性の低減を図る設計とした. ホワイトリスト登録に関して, 正常通信であるのに未登録となる事例があった. その多くは UDP であった. UDP にいて観測期間内で双方向成立が満たされなかったことが要因である. 反転 5 タプルによる判定は誤登録抑止に有効である. 一方, 要求応答が非常に短くキャッシュや再送の影響を受けやすい DNS では, 応答パケットの観測漏れなどが生じやすく, モデル判定が正常でも未登録になる場合が生じている. 改善策としては, 特定のプロトコルに限って IP ペアの双方向確認を導入する, またはプロトコルに応じて観測期間を調整することが有効と考えられる. 併せて, 現行のホワイトリストは一度登録されると失効しないため, 登録期間や定期的な再判定といった運用を導入し, 誤登録への体制を高める必要がある.

謝辞 本研究の一部は，JSPS 科研費 23H03396 の支援により行った.

参考文献

[1] F. Laiq, F. Al-Obeidat, A. Amin and F. Moreira, “*DDoS Attack Detection in Edge-IIoT Using Ensemble Learning*,” Proceedings of the 7th Cyber Security in Networking Conference (CSNet), pp. 204–207, 2023.

[2] S. Abiramasundari and V. Ramaswamy, “*Distributed Denial-of-Service (DDoS) Attack Detection Using Supervised Machine Learning Algorithms*,” Scientific Reports, Vol. 15, Article 13098, 2025.

[3] Kunal and M. Dua, “*Machine Learning Approach to IDS: A Comprehensive Review*,” Proceedings of the 3rd International Conference on Electronics, Communication and Aerospace Technology (ICECA), pp. 117–121, 2019.

[4] T. Saranya, S. Sridevi, C. Deisy, T.D. Chung and M.K.A.A. Khan, “*Performance Analysis of Machine Learning Algorithms in Intrusion Detection System: A Review*,” Procedia Computer Science, Vol. 171, pp.

1251–1260, 2020.
[5] 伊藤祐真, 小林良太郎, “攻撃トラフィック量に動的に対応する機械学習ベースの *FPGA* を用いた *NIPS*,” コンピュータセキュリティシンポジウム 2024 論文集, pp. 1594–1601, 2024.
[6] S. Kelly, “*dnsmasq*,” https://thekelleys.org.uk/dnsmasq/doc.html (Accessed 2025-8-8).
[7] curl, “*curl*,” https://github.com/curl/curl (Accessed 2025-8-8).
[8] antirez, “*hping*,” https://github.com/antirez/hping (Accessed 2025-8-8).