

2017年度 修士論文

パケット到着間隔に基づく  
DRDoS攻撃の検知法

提出日：2018年1月30日

指導：後藤滋樹教授

早稲田大学 基幹理工学研究科 情報理工・情報通信専攻  
学籍番号：5116F067-2

野口 大貴

# 目次

<b>第1章 序論</b>	<b>5</b>
1.1 研究の背景	5
1.2 研究の目的	6
1.3 本論文の構成	6
<b>第2章 DRDoS 攻撃</b>	<b>8</b>
2.1 NTP Amp 攻撃	8
2.2 SNMP Amp 攻撃	9
2.3 mDNS Amp 攻撃	10
<b>第3章 ChangeFinder</b>	<b>11</b>
<b>第4章 クラスタリング</b>	<b>13</b>
4.1 クラスタリングの概要	13
4.2 k-means 法	13
4.3 Affinity Propagation 法	14
<b>第5章 関連研究</b>	<b>16</b>
5.1 パケット連続到着時間に基づく攻撃の判別	16
5.2 到着間隔のクラスタリングによる攻撃の判別	18
<b>第6章 提案手法</b>	<b>21</b>
6.1 提案手法の概要	21
6.2 STEP1: 外れ値の除去	22
6.3 STEP2: 到着間隔の分類	23
6.4 STEP3: トラフィックパターンの分類	24

---

<b>第7章 評価実験</b>	<b>28</b>
7.1 実験の概要 . . . . .	28
7.2 実験に使用するデータ . . . . .	28
7.3 実験の結果 . . . . .	33
7.3.1 提案手法を適用した場合の実験結果 . . . . .	33
7.3.2 既存手法の精度評価 . . . . .	37
7.3.3 k-means 法との比較 . . . . .	39
<b>第8章 結論</b>	<b>41</b>
8.1 まとめ . . . . .	41
8.2 今後の課題 . . . . .	42
<b>参考文献</b>	<b>44</b>

# 図一覧

2.1	NTP Amp 攻撃	9
3.1	SDAR アルゴリズムによる ChangeFinder の二段階学習の手順	12
5.1	連続到着時間の定義	16
5.2	二段階検知方式による ISP 網へのセキュリティ装置の接続方式	17
5.3	提案手法における連続到着時間を用いた判定基準	17
5.4	到着間隔のグルーピング (概念図)	18
5.5	提案手法による k-means 法を用いた到着間隔のクラスタリングの例	19
5.6	良性通信及び悪性通信における連続到着時間の抽出	19
6.1	フローにおける到着間隔への ChangeFinder の適用例	22
6.2	STEP2 の詳細図	23
6.3	先頭 $N$ パケットの到着間隔のラベリング	24
6.4	連続到着時間の決定方法	26
7.1	実トラヒックの観測環境	29
7.2	フローにおける宛先 IP アドレス数の内訳	31
7.3	攻撃ホストとリフレクター間の通信の相関図	31
7.4	良性通信における到着間隔の分布	32
7.5	悪性通信における到着間隔の分布	32
7.6	先頭パケット数 $N$ と AUC の関係	34
7.7	先頭パケット数 $N = 16$ における ROC 曲線	35
7.8	高レート型に分類されたフローの $d$ の CDF による分布	36
7.9	低レート型に分類されたフローの $p$ の CDF による分布	36
7.10	高レート型に分類されたフローの $d$ の CDF による分布	38
7.11	Affinity Propagation 法による最適なクラスタ数 $k$ の出現数	40

# 表一覧

6.1	到着間隔のラベリングルール . . . . .	25
6.2	文字列から抽出すべき要素 . . . . .	25
6.3	推定結果と真の結果の関係 . . . . .	26
7.1	パケットヘッダ内情報のフィンガープリンティング条件 . . . . .	30
7.2	観測されたフローの種類別統計 . . . . .	30
7.3	ChangeFinder のパラメータチューニング内容 . . . . .	33
7.4	Affinity Propagation 法のパラメータチューニング内容 . . . . .	34
7.5	ChangeFinder により先頭パケットで外れ値除外が発生したフロー数 . . . . .	35
7.6	高レート型及び低レート型において分類されたフロー数 . . . . .	37
7.7	高レート型及び低レート型の分類による評価結果 . . . . .	37
7.8	高レート型及び低レート型の分類による評価結果 . . . . .	37
7.9	クラスタ数 $k$ を変化させた場合の k-means 法による精度比較 . . . . .	39
7.10	Affinity Propagation 法による最適なクラスタ数 $k$ の出現数 . . . . .	39

# 第 1 章

## 序論

### 1.1 研究の背景

DRDoS (Distributed Reflection Denial of Service) 攻撃による被害が増加の傾向にある。DRDoS 攻撃 (第 2 章) は, DDoS (Distributed Denial of Service) 攻撃の中でもリフレクターと称されるサーバを攻撃者が利用するものである。リフレクターとして選ばれるサーバは UDP プロトコル上で動作するサービスを提供する外部に対してオープンなサーバであり, 受信したクエリ情報に比べて大きなデータサイズで返答する機能を実装している特徴がある。攻撃者がこの機能を悪用し, 送信元を攻撃対象の IP アドレスに詐称したクエリを大量に送信することによる攻撃対象のネットワーク機器への膨大な負荷が問題となっている。

被害ホスト付近のネットワーク帯域が埋まる規模の DRDoS 攻撃が発生した場合, エンドユーザー側では攻撃トラフィックの総量を把握できない。従って, 適切に対策を講じることが難しく上位 ISP (Internet Service Provider) にて大量のトラフィックを発生させる要因となる DRDoS クエリの流入を抑えることが重要である [1, 2]。ISP では, 攻撃トラフィックの検知を目的として監視対象の通信フローをセキュリティ装置で観測することが一般的であるが, 単位通信量に対する観測コストが高いことから事前に疑いのあるフローのみを抽出してから装置に引き込む二段階検知方式が提唱されている [3]。ISP のようなバックボーンネットワークでは常に大量の通信が発生しており, 一次検知の段階では lightweight な検知手法を採用することで通信遅延などの影響を増加させないことが望ましい。

## 1.2 研究の目的

本研究では、バックボーンネットワークで観測された良性通信及び DRDoS クエリの到着間隔を分析し、後者を検知するための時間間隔によるシンプルな閾値を決定する手法を提案する。不審な送信元ポート番号と宛先ポート番号の組み合わせと DPI (Deep Packet Inspection) によって検知可能であるのは、攻撃手法が自明である NTP Amp 攻撃や SNMP Amp 攻撃などに限られ、従来の IDS を始めとしたシグネチャマッチングによる悪性通信検知方式はポート番号の組み合わせが正常であるかが不明な未知の攻撃を捕らえることが困難である [4]。近年新たな DRDoS 攻撃手法が報告されており [5], DRDoS 攻撃の種類を問わずして共通の特徴による検知手法が早急に求められている。

一方, DRDoS 攻撃を含めた DDoS 攻撃を利用者が課金を行うことで代行する Booter と呼ばれるサービスが存在する [6]。個人による DDoS 攻撃よりもリスクが低く, DRDoS 攻撃に用いるプロトコルの種類や攻撃の規模の選択が可能である Booter は利用されやすい。このようなサービスには ISP やエンドポイントでのトラフィックレートによる検知を回避する目的で高レートでのパケットの送信を行わないものがある。この性質に着目しパケット到着間隔のパターンを分析することで DRDoS 攻撃の種類に依存せず, かつ低レート型の攻撃の特徴を抽出できる。また, バックボーンネットワークにおける既存の DDoS 攻撃対策の欠点としてネットワーク機器に対する負荷増大が報告されており, 複雑な閾値を用いて検知を行うことは回避すべきである [7]。よって, パケットの到着間隔のみを利用する lightweight な手法を提案する。

## 1.3 本論文の構成

本論文は以下の章により構成される。

### 第 1 章 序論

本研究の概要について述べる。

### 第 2 章 DRDoS 攻撃

DRDoS 攻撃の概要とその種類について紹介する。

### 第 3 章 ChangeFinder

ChangeFinder アルゴリズムを解説する。

#### 第 4 章 クラスタリング

クラスタリングアルゴリズムを解説する.

#### 第 5 章 関連研究

本研究に関連する研究について述べる.

#### 第 6 章 提案手法

本研究の提案手法を説明する.

#### 第 7 章 評価実験

提案手法の有効性を示すために行った実験, 及びその結果と考察を示す.

#### 第 8 章 結論

本研究の結論を述べ, 残された課題を示す.

## 第 2 章

# DRDoS 攻撃

DRDoS 攻撃は DDoS (Distributed Denial of Service) 攻撃の一種であり, 攻撃者がリフレクターを利用して攻撃対象に増幅されたパケットを送信することで帯域負荷を増大させるものである. リフレクターは世界中に点在し, それらはクエリパケットに比べて大きなデータサイズで返答する機能を持つサーバであることが多い [8]. 攻撃者から送信元を攻撃対象の IP アドレスに詐称したクエリパケットを受信した場合, リフレクターからは悪意のあるパケットかどうかの判別が難しく, 良質な通信とみなし返答してしまい, 気づかずに攻撃プロセスに加担してしまうという問題がある. さらに, クエリに対し多くの情報で応答を行うリフレクターの特性上, 攻撃者が送信するパケットサイズは最小限に留めても有効であることから, 攻撃者の負担すべきリソースが TCP SYN Flood 攻撃などの DDoS 攻撃に比べ節約できるため, 小規模の攻撃者グループでも深刻な被害を及ぼしてしまう [9].

本章では, 現在問題となっている DRDoS 攻撃として, NTP Amp 攻撃 (第 2.1 節), SNMP Amp 攻撃 (第 2.2 節), mDNS Amp 攻撃 (第 2.3 節) の UDP を用いた 3 種類の攻撃を説明する. また, 本研究ではこれら DRDoS 攻撃をデータセットを用いて分析し, クエリパケットにどのような特徴が存在するかを把握した上で, 攻撃への防御手法を提案する.

### 2.1 NTP Amp 攻撃

NTP はシステム上の時刻情報を同期させるプロトコルであり, NTP Amp 攻撃は NTP の機能を利用して攻撃対象に増幅されたパケットを送りつける攻撃のことである. NTP では時刻の正確さを判定し, また規定値より外れていれば補正するために NTP サーバ同士で UDP 通信を行う. そのため, NTP サーバ自身は外部に公開されており, いかなる外部からの通信も許可

する設定であることが多い。攻撃者はこの仕様を利用し, monlist コマンドを用いて増幅攻撃を仕掛ける [10]. monlist コマンドはサーバが過去に通信を行った端末の最大 600 台分の履歴を要求する機能であり, 増幅率としては 200 から 1000 倍ほどの応答パケットを見込めるため, 甚大な被害が発生する可能性がある。また, 外部に公開されている NTP サーバは容易に見つけ出すことが可能であり, 効率的にリフレクターとなる NTP サーバを収集できてしまう [9, 11].

NTP Amp 攻撃による被害は拡大し続け, 2014 年には 400Gbps もの大規模な攻撃が報告されており, 被害者側での対策は困難な状況となっている [12]. 対策としては, 公開されている NTP サーバを探索し研究用データを収集するプロジェクトが行われている他に [13], 警視庁の注意喚起における対策として, 不必要に NTP サーバを外部ネットワークに公開しないことや, 公開する場合は攻撃に使われる monlist コマンドの機能を無効化することが推奨されている [14]. しかし, 正規のクライアントが monlist コマンドを利用する場合, 詐称された送信元 IP によって悪性判定を行えないためフィルタリングは難しい。

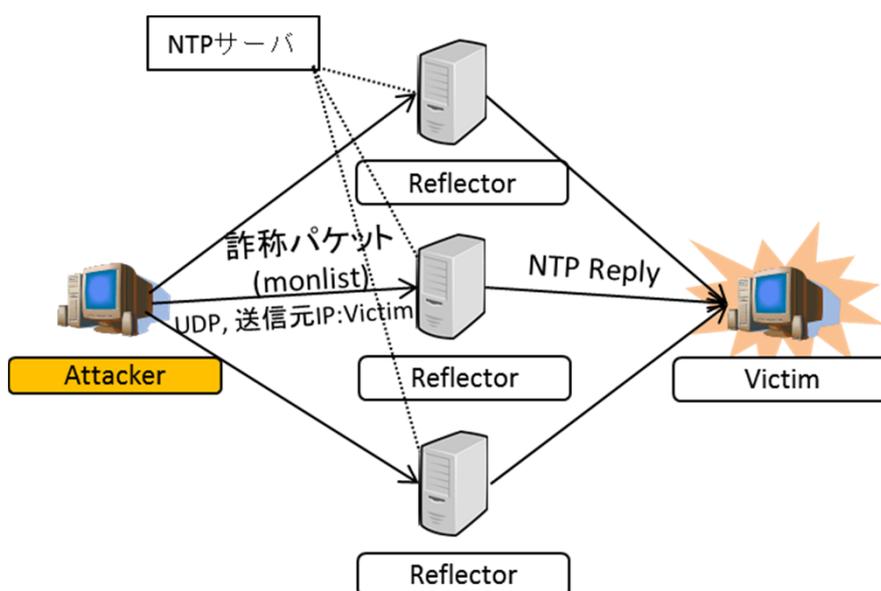


図 2.1: NTP Amp 攻撃

## 2.2 SNMP Amp 攻撃

SNMP はシステム上のルータ、スイッチ等のネットワーク機器やサーバの状態監視及び管理のための UDP プロトコルであり, SNMP Amp 攻撃は SNMP の機能を利用して攻撃対象に増幅されたパケットを送りつける攻撃のことである。SNMP メッセージは SNMP マネージャと

SNMP エージェントが同一のコミュニティというネットワークシステム空間に所属している場合に送受信される。コミュニティ名はデフォルトで public に設定されることがあり、変更のない状態で外部ネットワークに対してオープンにしている SNMP 機器が存在する [15]。攻撃者はこの仕様を利用し、GetBulk Request メッセージを用いて増幅攻撃を仕掛ける。GetBulk Request とは SNMPv2 で新たに追加された SNMP メッセージであり、SNMP エージェントが管理するシステムの機器情報の集合体である MIB (Management Information Base) から複数の管理情報を一括取得するメッセージである。そのデータ取得処理の繰り返し回数をオプションで設定できることから、増幅率は 650 倍ほどの応答パケットを見込める。

## 2.3 mDNS Amp 攻撃

mDNS (Multicast DNS) はローカルリンクネットワーク上に存在するサービスや機器を発見するプロトコルである。mDNS Amp 攻撃は mDNS の機能を利用して攻撃対象に増幅されたパケットを送りつける攻撃のことである。

mDNS に対応している機器の中では、外部ネットワークから送信されたユニキャストクエリに対して応答をする実装が存在する。応答にはネットワークに接続された機器に関する情報（機器の種類、モデル番号、オペレーティングシステム等）が含まれ、応答パケットサイズが増える。結果として、増幅率の高い攻撃を実現する [16]。

## 第 3 章

# ChangeFinder

ChangeFinder [17] は時系列モデルにおける変化点検出を行う二段階学習アルゴリズムである。時系列データの急激な変化を外れ値スコアという値で自動的に検出する。アルゴリズムの計算量がデータ数の線形オーダーであることから、ネットワーク監視等のオンライン処理に適性がある [18]。ChangeFinder では AR (Autoregressive) モデルでモデル化する。AR モデルは以下の式 (3.1) で示される。

$$y_t = \sum_{i=1}^o a_i * y_{t-i} + w \quad (3.1)$$

ここで、 $y_t$  は時系列データ、 $a_i$  は AR 係数、 $o$  はオーダー、 $w$  は平均がゼロ、分散が  $\sigma^2$  の正規分布に従うホワイトノイズである。AR モデルは時系列モデルの定常性を仮定しており、非定常なデータに対する適性がない。また、バッチ学習方式であり、過去のデータから未来のデータを予測することで計算時間が膨大になるという欠点がある。そこで、忘却型学習アルゴリズムである SDAR (Sequentially Discounting AR) アルゴリズムを適用することにより、計算量を抑えつつ非定常なデータを対象とすることが可能になる。SDAR アルゴリズムは以下の式 (3.2) で示される。

$$I = \sum_{i=1}^t (1-r)^{t-i} \log P(x_i | x^{i-1}, A_1, \dots, A_k, \mu, \Sigma) \quad (3.2)$$

ここで、 $r$  は忘却パラメータである。 $p(x_i | x^{i-1})$  を  $x_i$  の条件付き密度関数であり、 $x^{i-1}$  は  $(x_1, x_2, \dots, x_{i-1})$  を示す。式 (3.2) において  $I$  を最大化するように SDAR パラメータ  $A_i, \mu, \Sigma$  が決定される。SDAR アルゴリズムは逐次学習方式であり、直前の処理において決定された SDAR パラメータと最新のデータを用いて未来のデータを予測することから、計算量を減らすことが可能である [19]。

具体的な SDAR アルゴリズムを用いた二段階学習の手順を図 3.1 に示す。

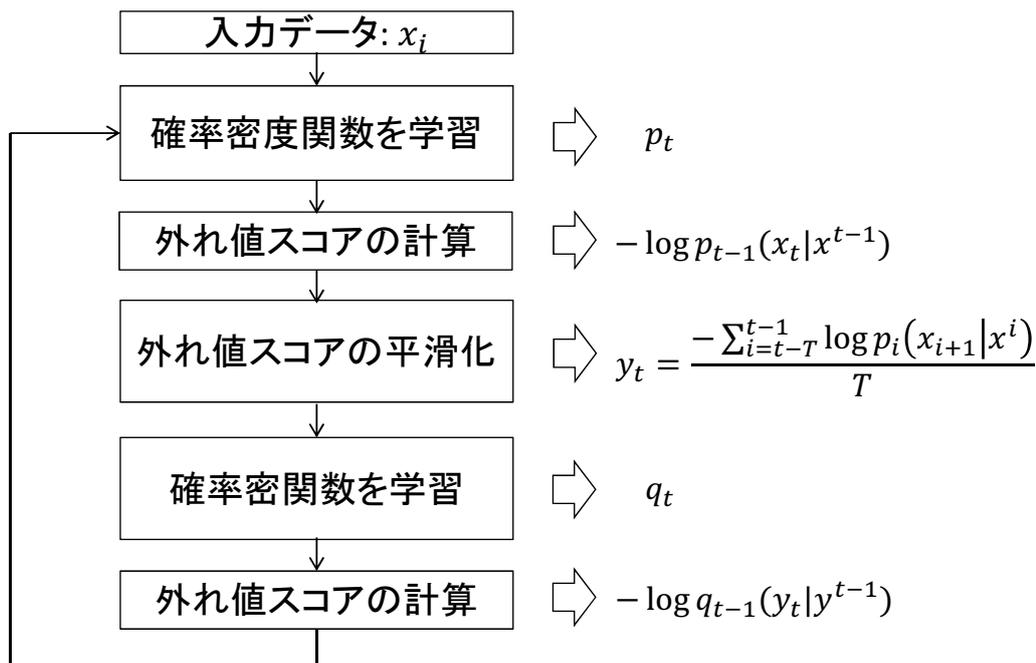


図 3.1: SDAR アルゴリズムによる ChangeFinder の二段階学習の手順

外れ値スコアは AR モデルで予測したデータに対して実際のデータ  $x_t$  の乖離度を示す。平滑化パラメータ  $T$  は、オーダー  $o$ 、忘却パラメータ  $r$  と共に ChangeFinder を適用する場面においてチューニングするべき値である。

## 第 4 章

# クラスタリング

本章では、本研究に用いるクラスタリングについて説明する。

### 4.1 クラスタリングの概要

クラスタリングとは、教師なし学習アルゴリズムの一つであり、対象となるデータを類似度（または非類似度）によって複数のクラスタ（グループ）に分類するものである。クラスタリング手法としては様々なアルゴリズムが提唱されており、それらは階層的であるか、あるいは非階層的かで大きく分類される。

本研究では非階層的クラスタリング手法である Affinity Propagation 法を用いたため、これに類する技術を説明する。

### 4.2 k-means 法

k-means 法は非階層的クラスタリング手法である。前提として予め分割するクラスタの数である  $k$  を手動で決定する必要があり、その条件下で以下の処理を実行し最適な分割となる解を導出する。

1.  $k$  個の各クラスタに対する代表点  $c_1, c_2, \dots, c_k$  をデータの中からランダムに選択する。
2. 各クラスタにおける重心を更新する。
3. 各データは最も距離が近い代表点を持つクラスタに割り当てられる。

4. 2に戻り, 各データの割り当てと各クラスターの重心の更新が収束するまで繰り返す.

上記の処理は, 以下の評価関数である式 (4.1) を最小化することと同値である.

$$\sum_{i=1}^n \sum_{x \in C_i} \|\mathbf{x} - \mu_i\|^2 \quad (4.1)$$

$C_i$  ( $1 \leq i \leq n$ ) は各クラスターを示す.  $n$  はクラスターの総数であり,  $\mathbf{x}$  はクラスターの要素である.  $\mu_i$  はクラスターにおける重心を示す. 式 (4.1) の値を最小化するために, クラスターごとに一番最適であると推定される  $\mu_i$  を選択する. k-means 法は計算コストが小さいことから実用的なクラスタリング手法であるが, 留意すべき事項としてクラスタリングの結果が初期の代表点に依存する性質がある. 初期値の選択方法により最終的に決定される各クラスターの重心が変動するため, 複数の結果が得られることがあるが, それらを評価する明確な指標がないため最適解を求めることは容易ではない [20]. また, クラスター数  $k$  の決定が不適切であると本来分割されるべきデータ群が同一クラスターの要素として分類されてしまう不具合が生じる.

### 4.3 Affinity Propagation 法

Affinity Propagation 法 [21] は非階層的クラスタリング手法であり, 二種類のメッセージと呼ばれる評価値を各データ間で交換することで exemplar と呼ばれる各クラスターの中心を全データの中から選出する. メッセージは各データ間における類似度 (similarity) によって計算される. データ  $i$  とデータ  $j$  の類似度  $s(i, j)$  が大きいほど双方の類似性が高いことを示す. 類似度の成分は preference と呼ばれるが, Affinity Propagation 法では preference の決定に関する制約はなく, 通常は類似度の中央値を用いる. この類似度を用いて responsibility と availability という二種類のメッセージが計算される [22]. responsibility は式 (4.2), availability は式 (4.3), 式 (4.4) として各データ間で計算が行われる.

$$r(i, k) \leftarrow s(i, k) - \max_{k' \text{ s.t. } k' \neq k} \{a(i, k') + s(i, k')\} \quad (4.2)$$

$$a(i, k) \leftarrow \min\{0, r(k, k) + \sum_{i' \text{ s.t. } i' \notin i, k} \max\{0, r(i', k)\}\} \quad (4.3)$$

$$a(k, k) \leftarrow \sum_{i' \text{ s.t. } i' \neq k} \max\{0, r(i', k)\} \quad (4.4)$$

responsibility はデータ  $k$  がデータ  $i$  に対する exemplar としての適合度を示す. availability はデータ  $k$  が exemplar となる時の, データ  $i$  のクラスタ要素としての適合度を示し, responsibility においてデータ  $k$  以外の exemplar 候補に対するデータ  $i$  の availability が高い時,  $r(i, k)$  は減少する. 二種類のメッセージは計算開始時にゼロで初期化され, 収束するまで再帰的に計算される. この時, 二種類のメッセージの振動を防ぐ目的で damping factor と呼ばれる値を考慮する [24]. 最終的に, 候補とされる exemplar とそれ以外の各データとの類似度が最大化される. Affinity Propagation 法ではこの操作によりクラスタリング結果が初期値に依存せず, クラスタ数も自動的に最適な数値に決定される. また, 前述の k-means 法に対して精度が向上していることが報告されている [24].

# 第 5 章

## 関連研究

本章では、パケットの時間間隔を用いた関連研究について述べる。

### 5.1 パケット連続到着時間に基づく攻撃の判別

パケットの連続到着時間による DDoS 攻撃の検知に関する研究が行われている。林 [3] は DDoS 攻撃の一種である HTTP GET Flood 攻撃の判定を行うために、良性通信において連続的にパケットが到着する時間が短いという特徴を利用して、図 5.1 に示される連続到着時間を各フローで計算を行い、連続到着時間が設定した閾値よりも長い通信を悪性通信であると判定している。ここで、フローとは  $(src\_ip, dst\_ip, dst\_port, protocol)$  の粒度で分類されたパケット列のことを意味する。



図 5.1: 連続到着時間の定義

時間間隔を特徴として用いる理由として、HTTP GET クエリはパケットサイズが小さく、従来型のトラフィック量で判別することが難しいことが説明されている。前提として、彼らの提案する手法は ISP 内に実装されるものであり、図 5.2 で示されるような二段階検知方式における

一次検知に適用される。DDoS 軽減装置は単位トラフィック量に対する検知コストが高く、引き込むべき通信を可能な限り減らすことで装置の負荷を軽減することが目的である。林は提案手法を定性的に評価を行っているが、実トラフィックデータを用いた定量的な評価は行っていない。図 5.3 に提案手法における連続到着時間を用いた判定基準を示す。

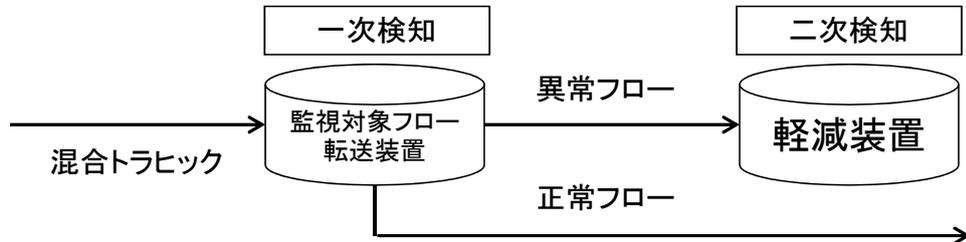
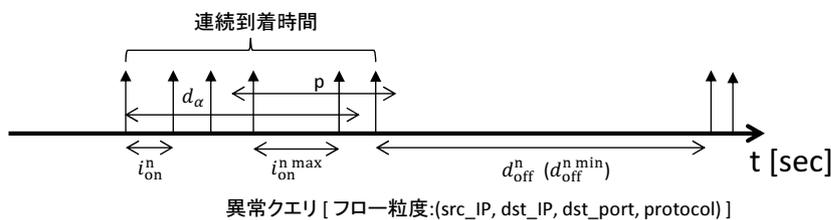


図 5.2: 二段階検知方式による ISP 網へのセキュリティ装置の接続方式



記号	記号の意味
$d_{off}^n$	正常時の非連続到着時間
$d_{off}^{n \min}$	$d_{off}^n$ の最小値
$i_{on}^n$	正常時のパケット到着間隔
$i_{on}^{n \max}$	$i_{on}^n$ の最大値
$i_{on}^a$	異常時のパケット到着間隔
$i_{on}^{a \max}$	$i_{on}^a$ の最大値
$p$	間隔判定閾値
$d_\alpha$	異常判定閾値

図 5.3: 提案手法における連続到着時間を用いた判定基準

図 5.3 における閾値  $p$  は隣り合うパケット群が連続到着かどうかを判定するための閾値であり、式 (5.1) を満たすものである。閾値  $p$  時間以内に連続して到着したパケットにおける到着時間の合計である連続到着時間が閾値  $d_\alpha$  よりも大きい時、該当フローを異常であると判定する。

$$\max\{i_{on}^{n \max}, i_{on}^{a \max}\} < p < d_{off}^{n \min} \tag{5.1}$$

林は閾値  $p$  の具体的な決定方法を議論しておらず、およその数値の仮定に議論を留めてい

る。また、連続到着時間における閾値  $d_\alpha$  は良性通信の連続到着時間の  $\alpha$  パーセンタイルにより決定されるが、連続到着時間の計算に用いる到着間隔の閾値  $T_{th}$  は 0.01 秒から 10 秒まで変化した時の結果を議論しており、明白な  $T_{th}$  の決定方法については触れられていない。この問題を解決するため、次に説明する筆者らの先行研究では二種類の閾値に関する決定方法を議論すると共に、実トラフィックデータを用いて DRDoS 攻撃を検知可能かを検証した。

## 5.2 到着間隔のクラスタリングによる攻撃の判別

筆者の先行研究 [25] では k-means 法によってパケットの到着間隔をクラスタリングし、連続的なパケットの到着間隔を抽出している。図 5.4 にフローに含まれる到着間隔のスカラー値が近いもの同士をグルーピングする概念図を示す。

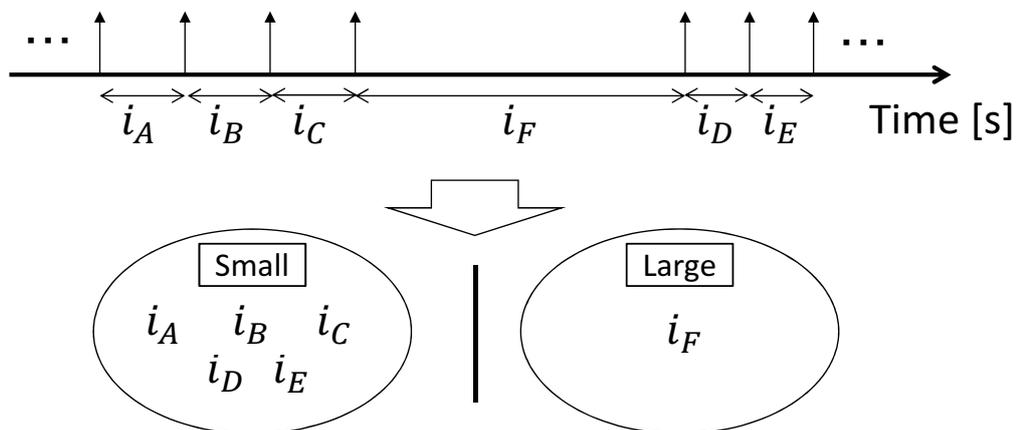


図 5.4: 到着間隔のグルーピング (概念図)

野口 [25] では、図 5.5 にあるように、各フロー ( $src\_ip, dst\_ip, dst\_port, protocol$ ) において連続到着間隔が含まれるクラスター  $C_{succ}$  を特定し、隣り合うパケットが連続しているかを判定する閾値  $T_{th}$  を決定している。各クラスターにおいて、クラスター内の同一の要素の個数の最大値  $Q$  にクラスター内の要素数  $R$  を掛け合わせた  $U$  を求める。最大の  $U$  を持つクラスターを  $C_{succ}$  とし、その中で最大の到着間隔を  $T_{th}$  とし、間隔が  $T_{th}$  以下のものをすべて連続到着間隔に分類する。

その後、図 5.6 のように  $T_{th}$  を用いて連続到着時間を計算し、その分布を良性通信及び悪性通信で比較することにより、最終的に通信の悪性判定を行うための閾値  $D_{th}$  を求めている。この時、悪性通信はフローの中で抽出された連続到着時間のうち、最も長いものを採用している。

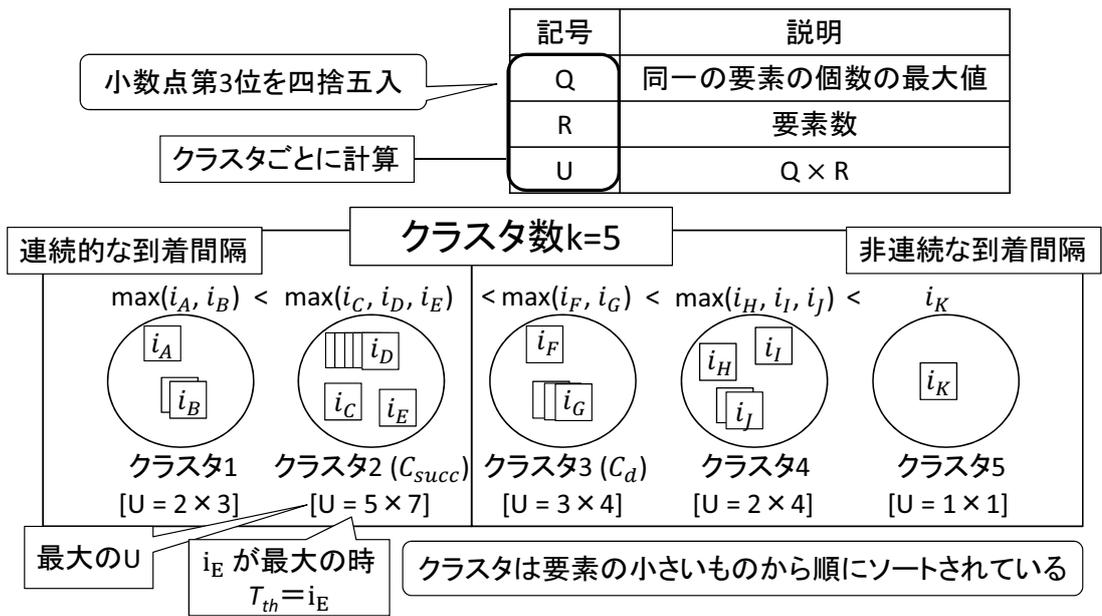


図 5.5: 提案手法による k-means 法を用いた到着間隔のクラスタリングの例

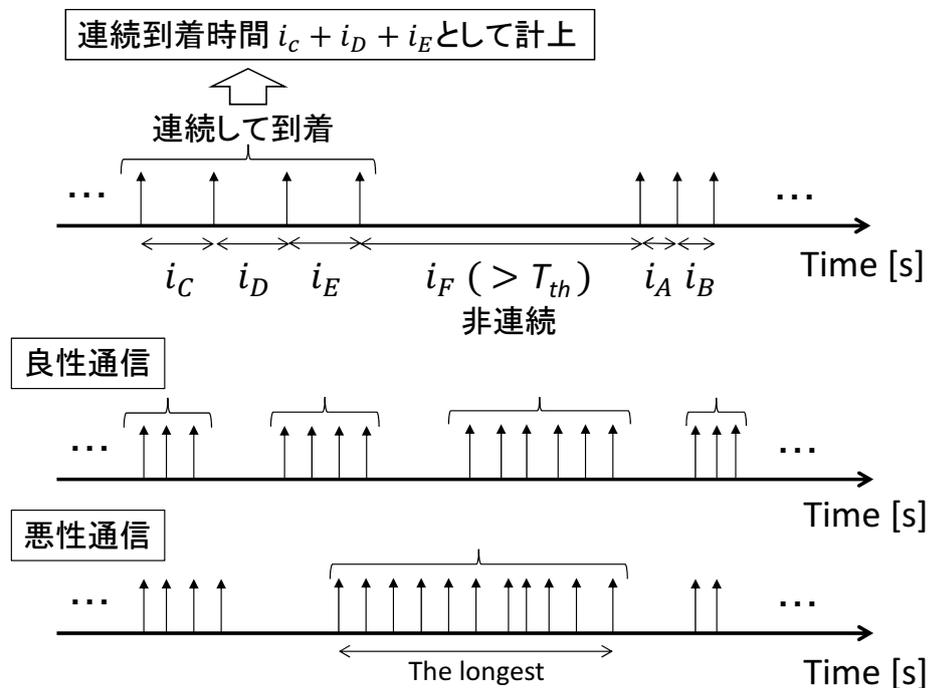


図 5.6: 良性通信及び悪性通信における連続到着時間の抽出

評価実験では, 二種類の閾値を利用してデータセンターの境界ルータで実際に観測されたトラフィックから, 良性通信の誤検知率を低減しながらも DRDoS 攻撃に関わる通信を高い精度で検知している. DRDoS 攻撃者は攻撃トラフィックの増幅率を高める傾向があり, 林 [3] の研究で扱われた HTTP GET クエリと同様に DRDoS クエリのパケットサイズは小さい. 従来のトラフィック量による検知は難しいが, 連続到着時間による検知は有効であることを先行研究では示している. ただし, 連続到着時間が長い攻撃は高レート型攻撃の特徴であり, 非連続到着時間を挟みながらパケットの送信を繰り返す低レート型の攻撃に対しては効果的ではないと考えられる.

# 第 6 章

## 提案手法

### 6.1 提案手法の概要

本研究における提案手法は3段階のステップに分けることが出来る. STEP1ではChangeFinderを用いて,トラフィックデータにおけるフローに含まれる著しく大きなパケット到着間隔を抽出し,一時的に除去する. STEP2ではフロー内のパケット到着間隔をクラスタリングによって分類する. STEP3ではクラスタリングによって得られた連続到着間隔および非連続到着間隔にラベリングを行い,到着間隔のパターンマッチングにより悪性通信を検知するための閾値  $D_{th}$  及び  $P_{th}$  を決定する. ここで,本手法で扱うフローの定義を以下に示す.

$$(src\_ip, src\_port, dst\_port, protocol)$$

DRDoS クエリを検知する場合,送信元 IP アドレスは詐称されており被害ホストの IP アドレスとなっている. 一方,宛先 IP アドレスはリフレクターの IP アドレスである. 第 5.2 節で述べた先行研究 [25] では,フローの条件に宛先 IP アドレスを含めており,単独のリフレクターへの DRDoS クエリを対象としている. DRDoS 攻撃は単独の被害ホストに対して複数のリフレクターを経由する報告があり [26], 単独のリフレクターを対象とする場合,攻撃フローの全体を捕らえることが出来ない問題がある. そこで,本手法では図 2.1 で示されるような複数のリフレクターを用いた攻撃に対応するため,宛先 IP アドレスは Any とする. また,本手法の STEP2, 3 を各フローにおける先頭  $N$  パケットに対し適用することで,  $D_{th}$  及び  $P_{th}$  を検知に用いる際にフロー情報を長期間保有する必要がないという利点がある. 早期に攻撃を検知できるので,占有リソース量の面で比較的低負荷な手法といえる. パラメータ  $N$  は本手法の適用時に最適に設

定する必要がある。その方法については実トラフィックデータを用いて第 7.2 節で説明する。

STEP1, 2, 3 の詳細を以下に述べる。

## 6.2 STEP1: 外れ値の除去

最初に、各フローに含まれるパケット到着間隔について ChangeFinder を適用し、著しく大きなものを外れ値として除外する。ChangeFinder を適用した例を図 6.1 として示す。

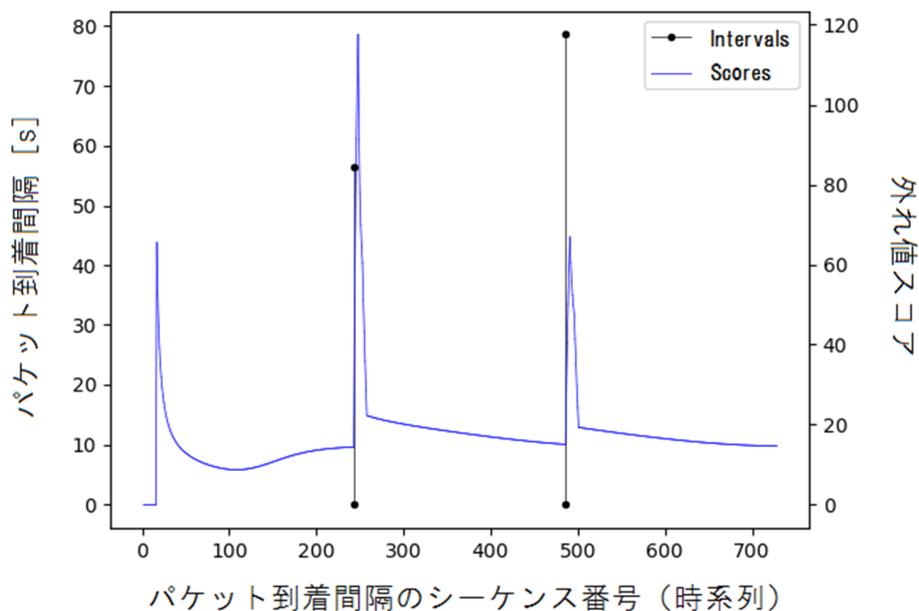


図 6.1: フローにおける到着間隔への ChangeFinder の適用例

端に黒点が打たれた直線は到着間隔であり、時系列順に左から並べた場合の各間隔の大きさを示している。実線は ChangeFinder の到着間隔に対する外れ値スコアを示し、スコアが著しく高いものは外れ値として抽出されていることがわかる。外れ値スコアの閾値は実装環境によってチューニングを行う。

各フローにおいて、ChangeFinder による閾値を超えたスパイク (急激な波形の上下) の数だけ以下の処理を実行する。

1. フローにおける到着間隔の最大値を求める。
2. 最大値を一時的に除外する。

上記の操作により, ChangeFinder がスパイクを発生させた到着間隔のうち, それらの最小値を  $ch_{min}$  とすると, フローに含まれる到着間隔の中で  $ch_{min}$  を超える間隔はすべて外れ値として除外される. そのため, 初期学習段階で外れ値が存在する場合スパイクが正常に観測されないが, 結果的に除外できる. 外れ値スコアは正規化されており, 到着間隔の範囲が異なる場合でも, 外れ値スコアは同一の範囲を取る. 図 6.1 では外れ値として抽出されているもの以外は微小な間隔がほとんどを占めており, 到着間隔を示す直線が描画されていないように見えている. なお, 最初の数パケットは初期学習に用いられているためスコアはゼロとなり, 最初のスコアの立ち上がりは初期学習段階から逐次学習段階への遷移に反応したものであり, 考慮に入れない.

### 6.3 STEP2: 到着間隔の分類

STEP1 の処理を適用した各フローの先頭  $N$  パケットの到着間隔に対し, Affinity Propagation 法によるクラスタリングを行う. この時, 事前に damping factor 及び preference 値を実装環境に合わせチューニングをする. Affinity Propagation 法ではフローにおける最適なクラスタ数  $k$  を自動的に決定するため, 事前に考慮する必要はない. 以下に STEP2 の詳細図を示す.

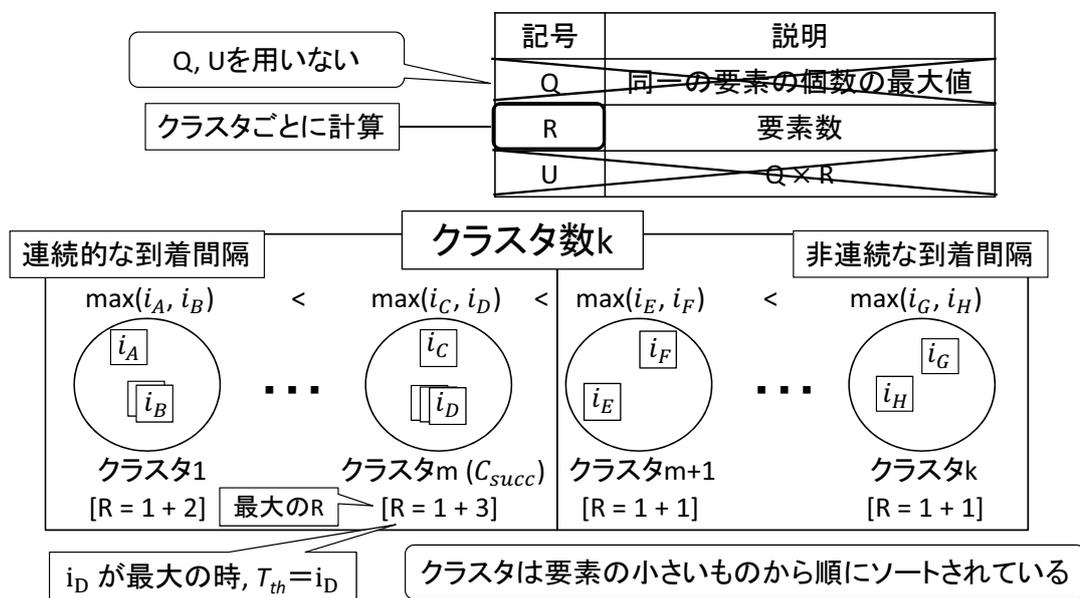


図 6.2: STEP2 の詳細図

第 5.2 節で述べた先行研究では  $C_{succ}$  の決定に  $Q$  及び  $R$  を用いていたが,  $Q$  を計算する際に到着間隔を小数点第二位で丸める必要があり情報が欠落してしまう可能性があるため, 本手法

では  $R$  のみで決定する. 先行研究と同様に, 最大の  $R$  を持つクラスタを  $C_{succ}$  とし, その中で最大の到着間隔を  $T_{th}$  とし, 間隔が  $T_{th}$  以下のものをすべて連続到着間隔に分類する. STEP1 でフローの先頭  $N$  パケット以内に外れ値が含まれていた場合, 除外された状態でクラスタリングを行うことにより, より精度の高いクラスタリング結果が見込める. 外れ値を含む状態でクラスタリングを行ってしまうと, 図 6.2 のクラスタ  $k$  は外れ値により占有される. これに伴い, 本来区別されるべき到着間隔が同一クラスタに収まってしまう不具合が発生すると考えられる.

## 6.4 STEP3: トラフィックパターンの分類

STEP2 により各フローにおいてクラスタリングされた到着間隔から, フローの先頭  $N$  パケットすべてに対して以下の図 6.3 のようにラベリングを行う. ラベリングルールについては, 表 6.1 に示すように,  $T_{th}$  以下の連続到着間隔を a,  $T_{th}$  を超える到着間隔を b とする.

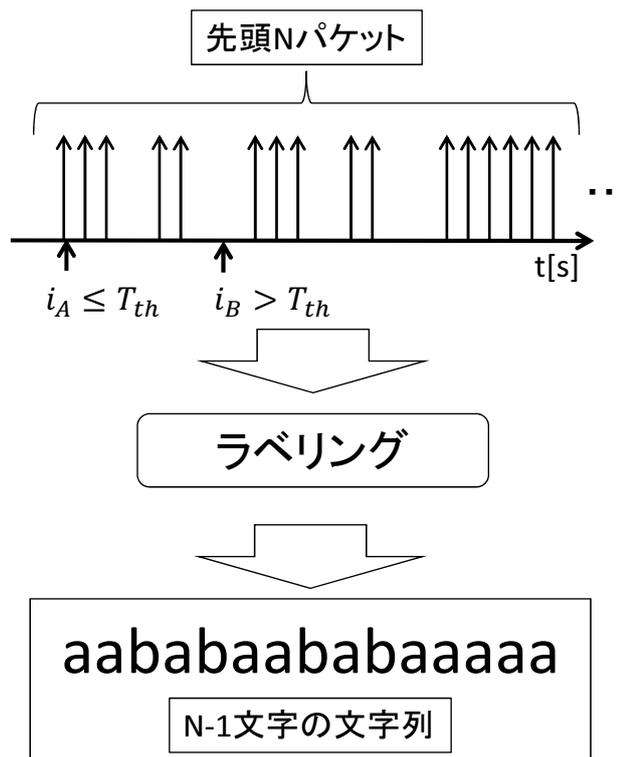


図 6.3: 先頭  $N$  パケットの到着間隔のラベリング

ラベリングされた到着間隔は  $N - 1$  文字の文字列と見なすことができる. この文字列から, 以下の表 6.2 に示す要素を抽出し要素の個数を計算する.  $lar_s$  が短時間に多く存在するほど,  $T_{th}$

表 6.1: 到着間隔のラベリングルール

到着間隔の種類	ラベリング結果
各フローにおける $T_{th}$ 以下の到着間隔	a
各フローにおける $T_{th}$ を超える到着間隔	b

以内で到着するパケットの位置は先頭  $N$  パケットの中で分散する。その結果、待ち時間を挟みながらごく少数のパケットを送信するようなトラヒックパターンである可能性がある。一方、 $sm_{succ}$  と比較して短い  $sm_m$  が多く存在する場合、前述したケースと比較してバースト的にパケットを送信するが、 $sm_{succ}$  よりも短いために、適度に待ち時間を挟むトラヒックパターンである可能性がある。なお、 $sm_{succ}$  の長さ  $(N - 1)/2$  を計算する際に小数点以下は切り捨てる。

表 6.2: 文字列から抽出すべき要素

呼称	抽出要素	例
$lar_s$	両隣が b 以外である文字 b	ba , aba, ab
$lar_m$	両隣が b 以外である文字列 b ... b	b ... ba, ab ... ba, ab ... b
$sm_m$	両隣が a 以外である文字列 a ... a	a ... ab, ba ... ab, ba ... a
$sm_{succ}$	長さが $(N - 1)/2$ である文字列 a ... a	aaaabbab, abaaaaaabb

次に、フローが高レートであるかを以下の条件で分類を行う。

- $lar_s + lar_m$  の個数が 2 以下であるか。
- $sm_{succ}$  が含まれるか。

$sm_{succ}$  が存在する場合は、先頭パケット数の半数のパケットがバースト的に連続到着していることから、高レート型フローであることを推定できる。また、 $lar_s + lar_m$  の個数が小さい場合も、非連続到着間隔が多く存在しないことから、先頭  $N$  パケットはバースト的に到着している。よって、同様に分類できる。

条件に合致する高レート型フローについては、第 5.2 節で述べた既存手法 [25] における連続到着時間の計算手法を STEP2 で得られた  $T_{th}$  を用いて適用する。ただし本手法では、図 6.4 に示すように良性通信及び悪性通信ともにフロー中に含まれる連続到着時間の最大値  $d$  を採用することにより、既存手法と比較して精密に分類を行う。

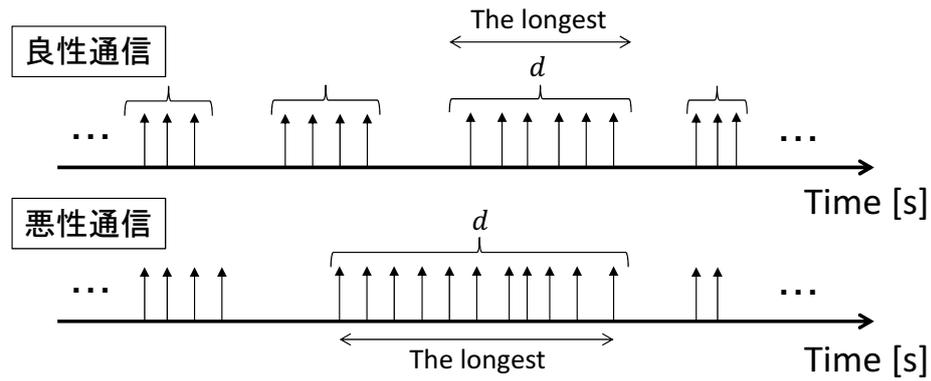


図 6.4: 連続到着時間の決定方法

条件に合致しないフローについては低レート型であると見なし, 以下の式 (6.1) に基づいて  $p$  を求める.

$$p = \{(lar_s \text{の合計}) + (sm_m \text{の合計})\} \quad (6.1)$$

最後に, 各フローのうち高レート型における  $d$ , 低レート型における  $p$  の分布から閾値  $D_{th}$ ,  $P_{th}$  を決定する. 具体的には, フローごとに計算されたすべての  $d$ ,  $p$  が閾値  $D_{th}$ ,  $P_{th}$  の候補であるので, 各々を閾値  $D'_{th}$ ,  $P'_{th}$  とした場合において以下の表 6.3 に示す評価指標を求める.

表 6.3: 推定結果と真の結果の関係

		真の結果	
		悪性通信	良性通信
推定結果	悪性通信	True Positive (TP)	False Positive (FP)
	良性通信	False Negative (FN)	True Negative (TN)

表 6.3 の中身はそれぞれの結果に当てはまるフロー数である. 閾値  $D'_{th}$  に関しては, すべてのフローが同一パケット数で比較されており, かつ  $sm_{succ}$  が含まれていることからフローの先頭  $N$  パケットの少なくとも半分以上が短時間に連続して到着するパケット列であるので,  $d$  が小さいフローは高レート型 DRDoS クエリであると推定することが可能である. よって,  $d$  が  $D'_{th}$  以下であるフローは悪性通信と推定し, 真の結果が同一であれば True Positive として計上する. また, 閾値  $P'_{th}$  に関しては,  $p$  が  $P'_{th}$  以下であれば低レート型 DRDoS クエリのトラヒックパターンに合致しないと判断し, 良性通信であると推定する. 真の結果が同一であれば True Negative として計上する.

得られた評価値から, Precision, Recall, F 値を以下の式に基づいて計算する. Precision は悪性通信であると推定したフローのうち, 正しく悪性通信であると推定できたものの割合である. Recall は実際に悪性通信であるもののうち, 正しく悪性通信であると推定できたものの割合である. F 値は Precision と Recall の調和平均により算出される評価尺度である.

$$Precision = \frac{TP}{TP + FP} \quad (6.2)$$

$$Recall = \frac{TP}{TP + FN} \quad (6.3)$$

$$F \text{ 値} = \frac{2Recall \cdot Precision}{Recall + Precision} \quad (6.4)$$

閾値  $D'_{th}$ ,  $P'_{th}$  ごとに、得られた F 値を比較し、最大である F 値における閾値  $D'_{th}$ ,  $P'_{th}$  を  $D_{th}$ ,  $P_{th}$  と決定する.

# 第 7 章

## 評価実験

### 7.1 実験の概要

本研究の提案手法の有効性を示すため、評価実験を行う。バックボーンネットワークで観測された実トラフィックデータに対し、提案手法を適用しどの程度の分類が可能であるかを測定する。

最初に、実験 1 として本研究の提案手法を適用し、精度評価を行う。特に、高レート型及び低レート型 DRDoS 攻撃の各々に対する検知精度を評価する。次に、実験 2 で第 5.2 節で述べた既存手法 [25] を実トラフィックデータに適用した際の精度評価を行う。既存手法では、高レート型 DRDoS 攻撃に対して有効であるが、低レート型 DRDoS 攻撃には効果は薄い。本研究で観測したデータに対して既存手法が有効であるかを評価する。実験 3 では本手法の STEP2 におけるクラスタリングアルゴリズムを k-means 法に置換した場合の精度を評価する。最後に、実験 4 で STEP3 におけるトラフィックパターンでの分類法を単純なものに変更した状態で評価を行い、STEP3 の有効性を検証する。

### 7.2 実験に使用するデータ

今回の実験では、図 7.1 で示すようにトランジットを提供するある日本の特異な学術ネットワークで観測された実トラフィックデータを用いた。

観測期間は、良性通信は 2017 年 11 月 27 日から 2017 年 11 月 28 日までであり、悪性通信については 2017 年 10 月 31 日から 2017 年 11 月 28 日までの約 1 ヶ月間である。良性通信の観測に用いたパケットのキャプチャフィルタを以下に示す。

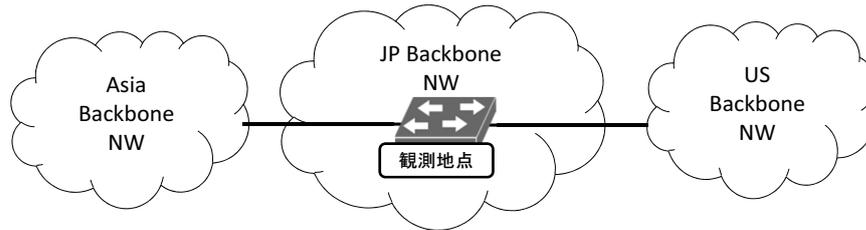


図 7.1: 実トラフィックの観測環境

```
(src_ip, src_port = 53, dst_port > 32767, protocol = udp)
```

良性通信のキャプチャフィルタ

多くの Linux カーネルにおけるエフェメラルポート番号は 32768 から 61000 であることが報告されている [27]. また, ポート番号の上限は 65535 であることを考慮し, 32678 以上の宛先ポート番号が指定されている DNS 応答フローを良性通信と定義している. 良性通信では *src\_port* で待ち受ける単一の DNS サーバが単一あるいは複数のクライアントと一対多の通信を行うことを想定する. また, 悪性通信の観測に用いたパケットのキャプチャフィルタを以下に示す.

```
(src_ip, src_port = [53, 80, 8080], dst_port = [53, 111, 123, 161, 1900, 5353], protocol = udp)
```

悪性通信のキャプチャフィルタ

ここで, 観測するパケットは DRDoS クエリであることから, *src\_port* は被害ホストが待ち受けるポート番号に詐称されていることを考慮し, 一般的に DDoS 攻撃の対象として標的にされる Web サーバ及び DNS サーバの待ち受けポート番号が *src\_port* に指定されていることを想定する. 宛先ポート番号はリフレクターの待ち受けポート番号である. 悪性通信のキャプチャフィルタに当てはまるフローのうち, 以下の表 7.1 の条件に合致するフローを悪性通信として用いた. これらは US-CERT で報告されている条件である [26].

悪性通信は Booter [6] などの DDoS 代行サービスによる攻撃が近年増加しており, 攻撃手段や被害ホストを利用者が選択できるという点と, DRDoS 攻撃がアプリケーションレベルではない点から攻撃トラフィックのパターンは宛先ホストに依存しないことを考慮し, 以上の条件で良性及び悪性通信を定義している.

結果として得られた良性及び悪性通信のフロー数を表 7.2 として示す. また, 図 7.2 に良性, 悪性各々のフローにおける宛先 IP アドレス数の内訳を示す.

表 7.1: パケットヘッダ内情報のフィンガープリンティング条件

<i>src_port</i>	<i>dst_port</i>	条件
80	53 (DNS)	QR=0, RCODE=0, Query class="Any"
	111 (RPC)	"0000000186a000000004" を含む, 異なる <i>dst_port</i> で XID が同一
	123 (NTP)	Monlist command
	161 (SNMP)	GetBulk request
	1900 (SSDP)	M-SEARCH request
	5353 (mDNS)	QR=0, RCODE=0, "000c" を含む

表 7.2: 観測されたフローの種類別統計

フローの種類		フロー数
良性通信		111
悪性通信	NTP	15
	SNMP	37
	mDNS	23

良性通信において全体の約 25.2% のフローは複数の宛先ホストに応答を送信しており, これらはパケット送信間隔が短縮される可能性があることを踏まえると, 比較的高レートと推定でき, 提案手法の STEP3 において高レート型と分類される可能性がある. 一方, 悪性通信は全体の約 49.3% のフローが複数のリフレクターに向けて DRDoS クエリを送信していることが分かる. また, 悪性通信において攻撃ホストとリフレクターの通信を黒線で示したものを図 7.3 として示す.

図 7.3 において三角のノードは攻撃ホストを, 丸のノードはリフレクターを意味する. 図から, 複数のリフレクターを用いている攻撃ホストのうち, 別々の攻撃ホストが同一のリフレクターを用いていることが確認できる箇所がある. このことから, 攻撃ホスト間でリフレクターの IP リストが共有されている可能性もあると推定される.

図 7.4, 7.5 に良性及び悪性通信における到着間隔の分布を示す. ここで, 到着間隔は小数点第 3 位を四捨五入し同一のものを計上している.

良性及び悪性通信において, 0.1 秒から 10 秒の間に多く分布し, この範囲における分布の形

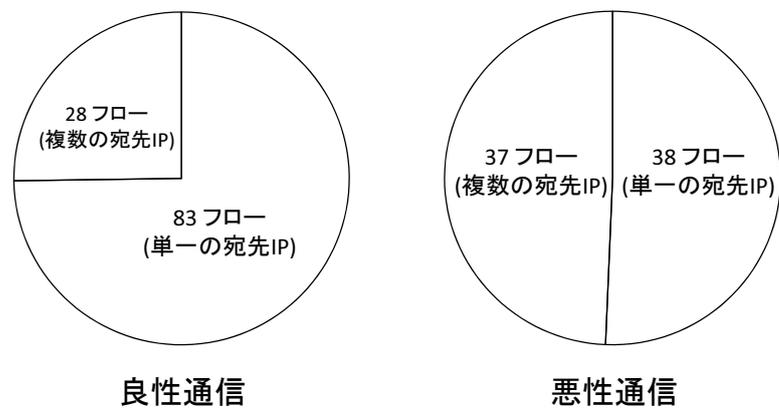


図 7.2: フローにおける宛先 IP アドレス数の内訳

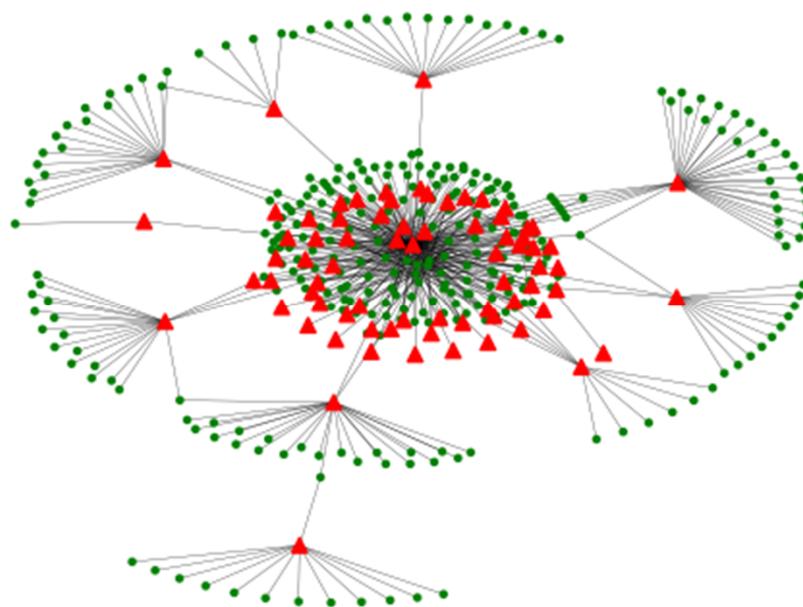


図 7.3: 攻撃ホストとリフレクター間の通信の相関図

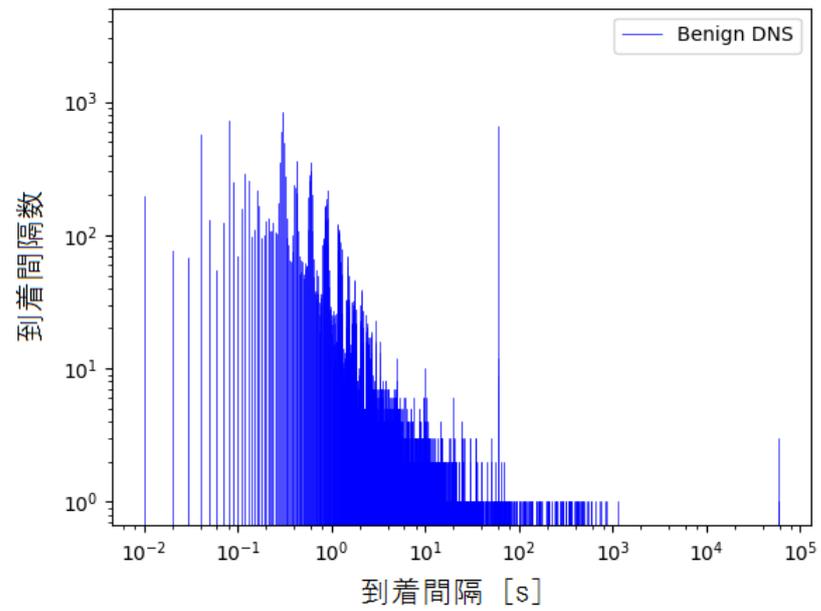


図 7.4: 良性通信における到着間隔の分布

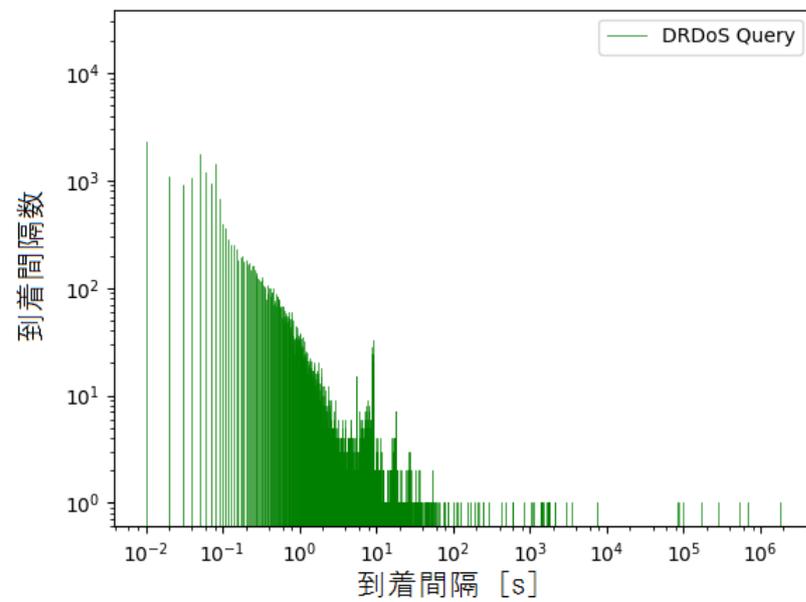


図 7.5: 悪性通信における到着間隔の分布

状がほぼ同一であることから、悪性通信の中には良性通信の packets 送信レートに擬態した低レート型が含まれていると推定される。その一方で、0.01 秒付近では悪性通信における packets 数が良性通信のおよそ 2 倍であることから、高レート型も混在していることが考えられる。

## 7.3 実験の結果

### 7.3.1 提案手法を適用した場合の実験結果

本節では提案手法を実トラフィックデータに適用した結果を説明する。まず、STEP1 における ChangeFinder におけるパラメータは以下のようにチューニングを行った。平滑化パラメータ  $T$  は 5 から 10 の間が標準的である [18]。平滑化パラメータが小さいと局所的な変動に対し過敏に反応する。そこで、著しく大きな外れ値にのみ反応するように  $T$  を 10 とした。

表 7.3: ChangeFinder のパラメータチューニング内容

パラメータ名	値
忘却パラメータ $r$	0.01
オーダー $o$	1
平滑化パラメータ $T$	10

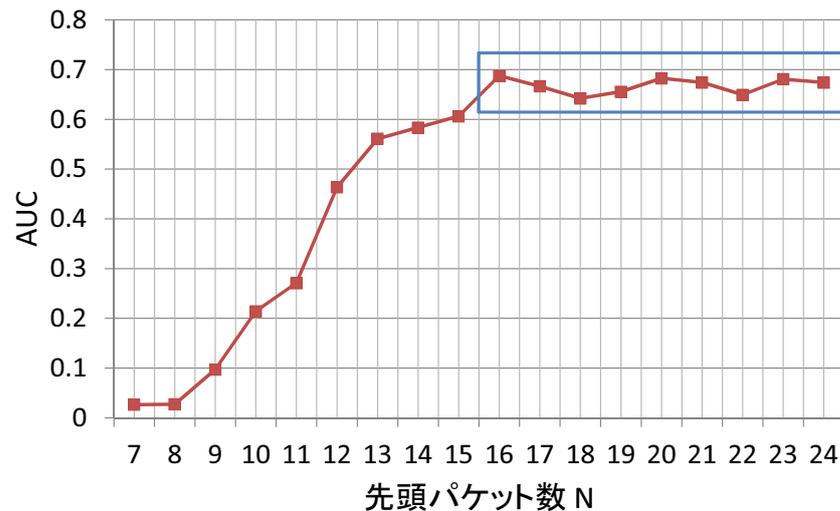
また、図 6.1 における ChangeFinder の適用例は実トラフィックデータに基づいたものであるが、図 6.1 のような外れ値スコアが 25 以上のスパイクが出現しているケースが多く存在していたことを加味し、外れ値スコアの閾値を 25 とした。次に、STEP2 における Affinity Propagation 法におけるパラメータは以下のようにチューニングを行った。杉原 [22] によれば、preference は通常は類似度行列の中央値を用いる。また、Affinity Propagation の実装は python2.7 のパッケージである scikit-learn を用いており、推奨される damping factor が 0.5 としていた。よって、本研究でも同様のチューニングを行った。

STEP2, 3 では各フローの先頭  $N$  packets を扱う。そこで、 $N$  を 16 とした。その根拠を図 7.6 として以下に示す。

図 7.6 は  $N$  を変動させる時、本提案手法の最終的な評価指標である True Positive Rate を縦軸、False Positive Rate を横軸とする ROC 曲線を描いた場合の AUC (Area Under the Curve) を示している。ROC 曲線とは手法の分類性能を示す指標であり、本研究においては、閾値  $D_{th}$  を

表 7.4: Affinity Propagation 法のパラメータチューニング内容

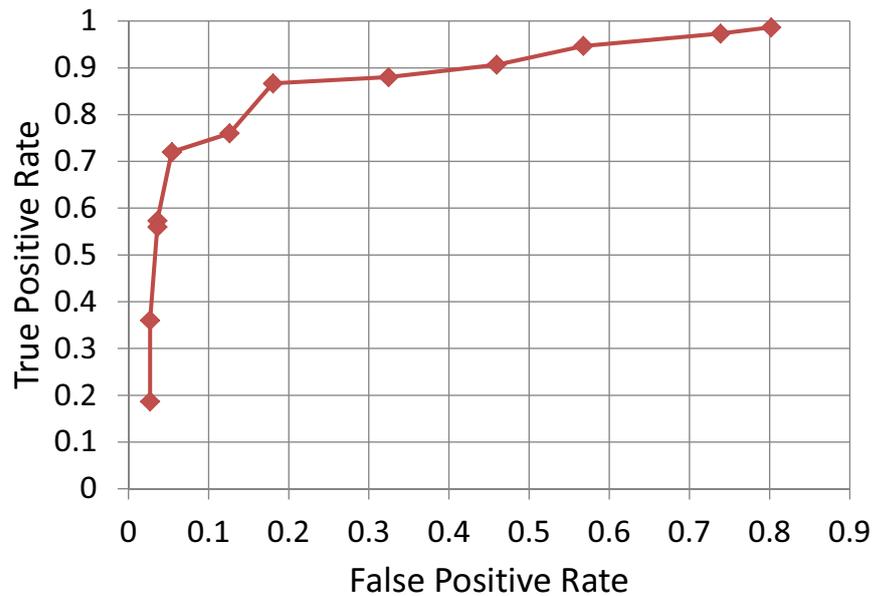
パラメータ名	値
damping factor	0.5
preference	類似度行列の中央値

図 7.6: 先頭パケット数  $N$  と AUC の関係

最適な  $d$  に固定した状態で  $p$  を  $P'_{th}$  としたそれぞれの場合における True Positive Rate と False Positive Rate の関係を示す。ここで、 $N = 16$  の時に AUC は全体における最大値  $AUC_{max}$  をとる。以下の図 7.7 に  $N = 16$  の時の ROC 曲線を示す。

ROC 曲線の下側の面積を AUC と呼ぶ。AUC は 0 から 1 までの範囲の値をとり、1 に近づくほど手法の分類性能が良いことを示す。図 7.6 から、AUC の値は  $N = 16$  までは単調増加していくが、 $N = 16$  からは一定となっていることが分かる。 $N \geq 16$  で  $AUC_{max}$  付近を上下していることを考慮すると、高精度な悪性通信の分類に必要な情報量として最低限必要である先頭パケット数は 16 と断定できる。すなわち、本研究で扱った実トラフィックデータにおいて、悪性通信のトラフィックパターンの高精度な検出には少なくとも先頭 16 パケットを対象にすることができれば可能であることが示されている。

なお、先頭  $N$  パケットに外れ値を含み STEP1 の ChangeFinder で除外されたフロー数を表 7.5 に示す。表 7.2 と比較して、良性通信では約 5%、悪性通信では約 9%のフローが ChangeFinder

図 7.7: 先頭パケット数  $N = 16$  における ROC 曲線

による外れ値除外の対象となっている。

表 7.5: ChangeFinder により先頭パケットで外れ値除外が発生したフロー数

フローの種類	フロー数
良性通信	6
悪性通信	6

次に, STEP3 において高レート型に分類されたフローの  $d$  の分布を CDF で示した図 7.8 を示す. 同様に, 低レート型に分類されたフローの  $p$  の分布を CDF で示した 7.9 を示す.

第 6.4 節において, すべての  $d$  及び  $p$  が  $D_{th}$ ,  $P_{th}$  の候補  $D'_{th}$ ,  $P'_{th}$  になることを説明した. 各図において赤字で示されている値が, 最終的な F 値が最大の時の  $D_{th}$ ,  $P_{th}$  となっており, これらは良性及び悪性通信の CDF の差が最大の時のものと同一である. 高レート型及び低レート型の分類による評価結果を表 7.6, 表 7.8 として示す.

今回観測した実トラフィックデータでは, 悪性通信の 20%が高レート型, 80%が低レート型として分類された. 良性通信では, 約 23%が高レート型, 約 77%が低レート型として分類された. この結果から, 悪性通信は低レートで発生する傾向があるといえる. 高レート型として分類された良性通信は, 複数の正常なクライアントへの DNS 応答が時間的に重なったものと推定され

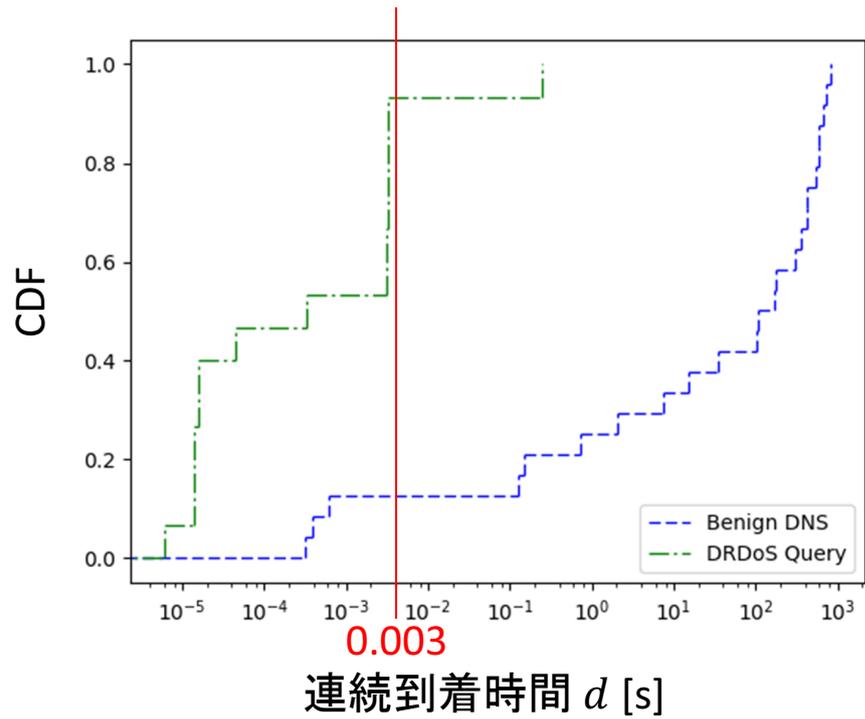
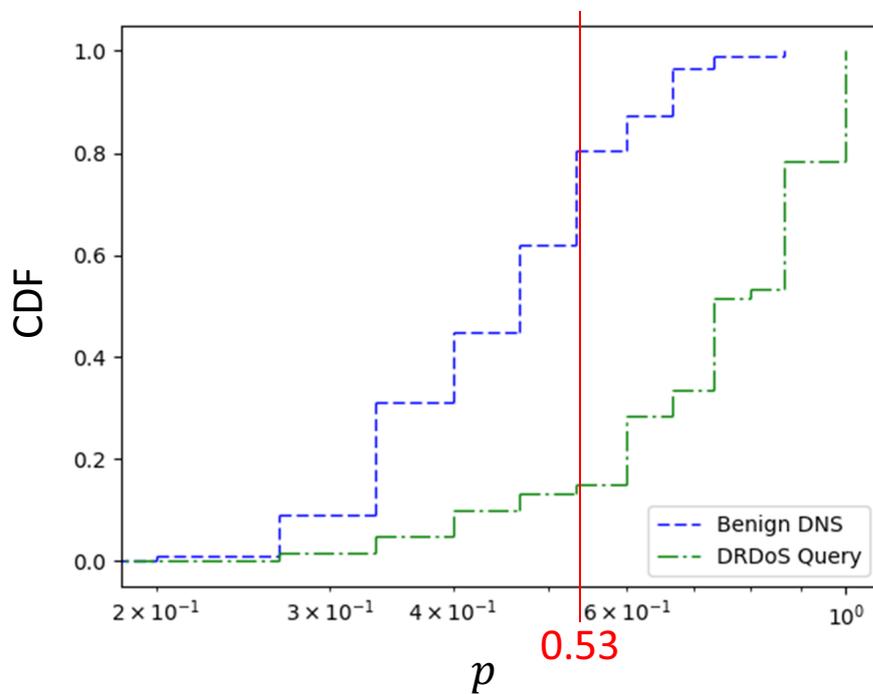
図 7.8: 高レート型に分類されたフローの  $d$  の CDF による分布図 7.9: 低レート型に分類されたフローの  $p$  の CDF による分布

表 7.6: 高レート型及び低レート型において分類されたフロー数

種類	True Positive	False Negative	True Negative	False Positive
高レート型	14	1	21	4
低レート型	51	9	69	17

表 7.7: 高レート型及び低レート型の分類による評価結果

種類	TP Rate (Recall)	FN Rate	TN Rate	FP Rate	Precision	F 値
高レート型	0.933	0.067	0.840	0.160	0.778	0.848
低レート型	0.850	0.150	0.802	0.198	0.750	0.797

る. 高レート型及び低レート型の悪性通信における TPR が共に高く, FPR も同様に低いことから, 各々の分類手法は高精度であり, かつ良性通信の誤検知を低減できていることを確認できた. 以下に, 総合的な評価指標を示す.

表 7.8: 高レート型及び低レート型の分類による評価結果

種類	TP Rate (Recall)	FN Rate	TN Rate	FP Rate	Precision	F 値
高+低レート型	0.866	0.134	0.819	0.181	0.756	0.807

### 7.3.2 既存手法の精度評価

本節では, 第 5.2 節で述べた既存手法を実トラフィックデータに適用した結果を示す. 図 7.10 は, 既存手法による最終的な連続到着時間の分布を示している.

表 7.6 において指摘したように, 今回扱った実データトラフィックにおける悪性通信は低レート型が多くを占める. 結果的に, 図 7.10 における悪性通信は連続到着時間が短く計算されてしまい, 良性通信と区別するための閾値  $D_{th}$  の決定が難しいことを示している.

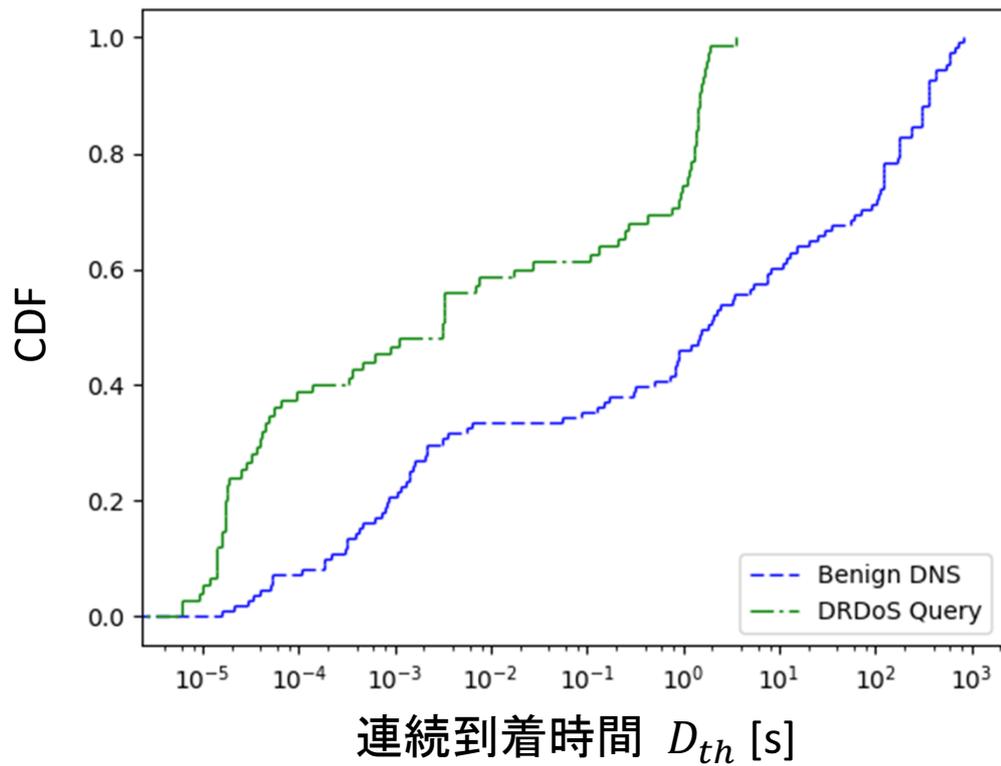


図 7.10: 高レート型に分類されたフローの  $d$  の CDF による分布

### 7.3.3 k-means 法との比較

本節では、STEP2 におけるクラスタリングアルゴリズムを k-means 法とした場合の F 値による評価を行う。k-means 法のクラスタ数  $k$  を変化させた時の最終的な評価指標を以下に示す。

表 7.9: クラスタ数  $k$  を変化させた場合の k-means 法による精度比較

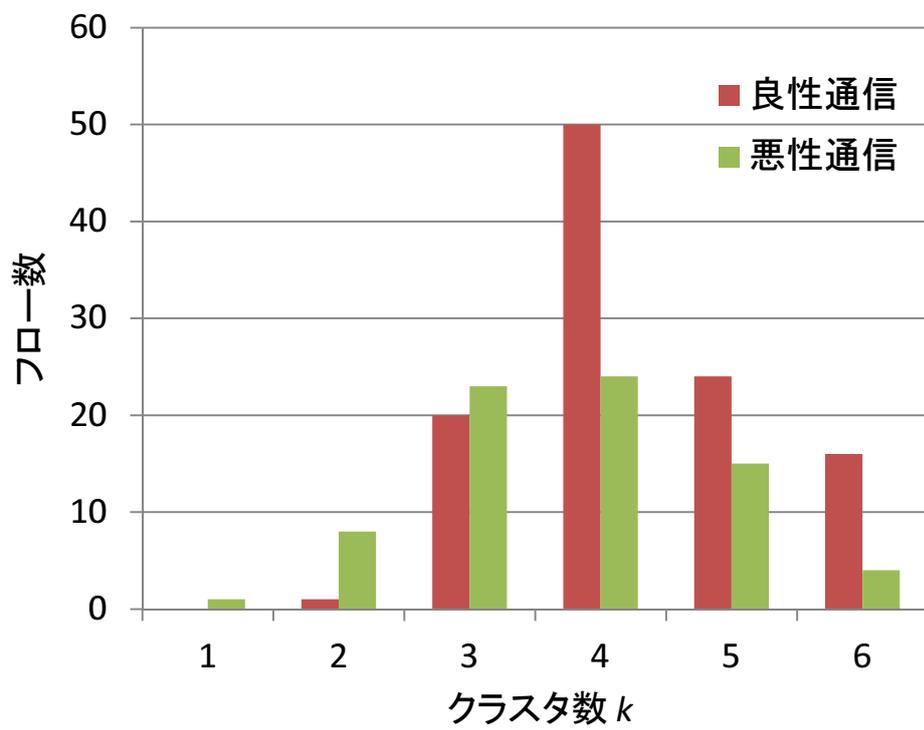
クラスタ数 $k$	TP Rate (Recall)	FN Rate	TN Rate	FP Rate	Precision	F 値
2	0.760	0.240	0.752	0.248	0.679	0.717
3	0.613	0.387	0.847	0.153	0.730	0.666
4	0.760	0.240	0.890	0.110	0.826	0.792
5	0.720	0.280	0.890	0.110	0.818	0.766

表 7.9 から、F 値は提案手法と比較して低い。このことから、k-means 法によるクラスタリングを行った場合、最適にクラスタ数  $k$  が選ばれていないフローが存在することが考えられる。ここで、 $k = 4$  の場合の F 値が k-means 法による評価値の中で最大である。この理由を以下の図 7.11 によって説明する。

表 7.10: Affinity Propagation 法による最適なクラスタ数  $k$  の出現数

フローの種類	$k = 1$	$k = 2$	$k = 3$	$k = 4$	$k = 5$	$k = 6$
良性通信	0	1	20	50	24	16
悪性通信	1	8	23	24	15	4

図 7.11 は、Affinity Propagation 法を適用した場合における、各フローが最適な  $k$  でクラスタリングされた時の  $k$  の出現回数である。特に、良性通信において  $k = 4$  でクラスタリングされたフローが多いことを示している。このことから、今回得られた実トラフィックデータに含まれるフローに対して  $k = 4$  でクラスタリングを行う場合、最適にクラスタリングされるフローの数が最も多くなることで比較的高い精度が出ていることが推定される。

図 7.11: Affinity Propagation 法による最適なクラスタ数  $k$  の出現数

# 第 8 章

## 結論

### 8.1 まとめ

本研究では, 実トラフィックデータを用いて良性通信及び DRDoS クエリの到着間隔を分析し, 後者を検知するための時間間隔によるシンプルな閾値を決定する手法を提案した. 実際に観測された DRDoS クエリは複数の宛先ホストに向けて同一の攻撃ホストから送信されたものが半分を占めており, かつ DRDoS クエリの種類も複数存在した. 実験によりこれらを高精度で分類する閾値を特定できたので, 複数種類の DRDoS 攻撃に対応し, 同時に複数のリフレクターを網羅的に検知可能であることを示した. また, DRDoS クエリ特有の特徴量を用いずに時間間隔を用いたので, 未知の攻撃にも対応できる可能性を示し, かつ DDoS 攻撃の一次検知に用いる閾値に必要な, lightweight であるという特性を満たすことを確認した.

筆者らの先行研究で議論されている既存手法は高レートの DRDoS 攻撃に有効である. この性質を利用し, 低レート型及び高レート型ともに検知可能な閾値を決定できることを確認した. 同時に, Affinity Propagation 法を用いることでクラスタ数  $k$  をフローごとに手動で決定しなければならない問題も解決した.

本研究における成果は ISP などのバックボーンネットワークでの DRDoS 攻撃の一次検知段階に有用であり, 閾値によるフローの検知後は二次検知段階にて厳密に調査を行うことを想定している. 本手法を実装することにより, 観測対象のフロー数を削減することができるため, DDoS 軽減装置等にかかる負荷を軽減できる効果がある.

## 8.2 今後の課題

本研究の今後の課題を以下に述べる.

- DRDoS クエリの収集

本研究で扱った DRDoS クエリのフロー数は 75 であるが, 実トラヒックに即した閾値を決定するためにより多くの DRDoS クエリを収集する必要がある. また, 今回得られた攻撃は三種類であったので, より多くの種類の DRDoS クエリを収集することで共通の特徴を見出し, 提案手法に組み込むことにより分類精度の向上を見込む必要がある.

- クラスタリング手法の最適化

提案手法における STEP2 に組み込んだ Affinity Propagation 法は, 本実験で k-means 法に対して高精度にクラスタリングを行うことを示したが, より一層の精度向上を見込むためにパラメータチューニングを厳密に行う必要がある. また, 藤原 [24] が Affinity Propagation 法の高速度手法を提唱するように, 今後性能向上のために改良される可能性は十分にある. よって, 最新のクラスタリング手法について引き続き調査を行うことが肝要である.

# 謝辞

本修士論文を作成するにあたり、日頃よりご指導を頂いた早稲田大学基幹理工学研究科の後藤滋樹教授に深く感謝いたします。また、本研究のために貴重なデータを提供して頂いた KDDI 株式会社の池田貴俊氏に心より感謝いたします。最後に、本研究を進めるにあたり、多大なるご協力を頂きました後藤研究室の諸氏に感謝いたします。

## 参考文献

- [1] 齋藤衛, “DoS/DDoS 攻撃対策 (1) ～ISP における DDoS 対策の現状と課題～”, 情報処理学会 54-5, pp.468–474, April, 2013.
- [2] 高田美紀, “ISP における DoS/DDoS 攻撃の検知・対策技術”, <https://www.nic.ad.jp/ja/materials/iw/2013/proceedings/s2/s2-takata.pdf>, InternetWeek, 2013.
- [3] 林 裕平, 西山 聡史, 鈴木 昭徳, 阪井 勝彦, 工藤 伊知郎, 神谷 和憲, “パケット連続到着時間を判定基準とした攻撃検知方式の評価”, IEICE technical report 115(488), pp.53–58, March, 2016.
- [4] Cisco, “Defeating DDoS Attacks”, [https://www.cisco.com/c/en/us/products/collateral/security/traffic-anomaly-detector-xt-5600a/prod\\_white\\_paper0900aecd8011e927.html](https://www.cisco.com/c/en/us/products/collateral/security/traffic-anomaly-detector-xt-5600a/prod_white_paper0900aecd8011e927.html), 2004.
- [5] Akamai, “Akamai Security Intelligence Response Team Identifies New Reflection Attack Method”, <https://www.akamai.com/uk/en/about/news/press/2017-press/akamai-security-intelligence-response-team-identifies-new-reflection-attack-method.jsp>, April, 2017.
- [6] JJC de Santanna, Roland M. van Rijswijk, R.J. Hofstede, Anna Sperotto, Mark Wierbosch, Lisandro Zambenedetti Granville, Aiko Pras, “Booters - an analysis of DDoS-as-a-Service attacks”, Integrated Network Management (IM), 2015 IFIP/IEEE International Symposium on. IEEE, 2015.
- [7] Monika Sachdeva, Gurvinder Singh, Krishan Kumar, “Deployment of Distributed Defense against DDoS Attacks in ISP Domain”, IJCA, Volume 15 No.2, pp.25–31, February, 2011.
- [8] Impress, “リフレクション攻撃が増加し、DDoS 攻撃は小型化／2015 年のセキュリティを振り返る”, [http://internet.watch.impress.co.jp/docs/column/security/20160104\\_737474.html](http://internet.watch.impress.co.jp/docs/column/security/20160104_737474.html), 2016.

- [9] キーマンズネット, “国内初の DDoS 犯人摘発例は高校 1 年生! 有名オンラインゲームが標的に”, <http://www.keyman.or.jp/at/30007392/>, 2014.
- [10] TREND MICRO, “NTP を利用する DDoS リフレクション攻撃に対する対策”, <http://blog.trendmicro.co.jp/archives/8437>, 2014.
- [11] Akamai, “An Analysis of DrDoS SNMP/NTP/CHARGEN Reflection Attacks”, <https://www.stateoftheinternet.com/downloads/pdfs/2013-state-of-the-internet-web-security-white-paper-drdo-snmp-ntp-charge-attacks.pdf>, 2014.
- [12] threat post, “NTP Amplification Blamed for 400 Gbps DDoS Attack”, <https://threatpost.com/ntp-amplification-blamed-for-400-gbps-ddos-attack/104201/>, 2014.
- [13] Network Time Foundation, “NTP Scanning Project”, <http://openntpproject.org/>, 2016.
- [14] NPA JAPAN Cyber Force Center, “NTP サーバを踏み台としたリフレクター攻撃 (NTP リフレクター攻撃) に対する注意喚起について”, <https://www.npa.go.jp/cyberpolice/detect/pdf/20140117.pdf>, 2014.
- [15] Akamai, “SNMP Reflection DDoS Attacks”, <https://www.akamai.com/us/en/multimedia/documents/state-of-the-internet/snmp-reflector-attacks-threat-advisory.pdf>, 2014.
- [16] JVN/VU, “マルチキャスト DNS 実装が外部からのユニキャストクエリに応答する問題”, <https://jvn.jp/vu/JVNVU98589419/>, 2015.
- [17] Junichi Takeuchi, Kenji Yamanishi, “A unifying framework for detecting outliers and change points from time series”, TKDE2006, Vol. 18, No.4, pp.482–492, 2006.
- [18] NTT データ数値システム知識工学部, “変化点検出 ChangeFinder”, <http://cl-www.msi.co.jp/reports/changefinder.html>, July, 2013.
- [19] Ryosuke Matsumoto, “SDAR アルゴリズムと統計的手法による時系列からの外れ値と変化点の検出”, [https://speakerdeck.com/matsumoto\\_r/sdararugorizumutotong-ji-de-](https://speakerdeck.com/matsumoto_r/sdararugorizumutotong-ji-de-)

- shou-fa-niyorushi-xi-lie-karafalsewai-rezhi-tobian-hua-dian-falsejian-chu, July, 2013.
- [20] Takashi Onoda, Miho Sakai, Seiji Yamada, “Comparison of Clustering Results for k-means by using different seeding methods”, 27th Fuzzy System Symposium, pp.231–236, September, 2011.
- [21] Brendan J. Frey, Delbert Dueck, “Clustering by Passing Messages Between Data Points”, Science, Volume 315, pp.972–976, February, 2007.
- [22] Takahiko Sugihara, Tsuyoshi Murata, “Community Detection by Affinity Propagation”, JSAI 2011, June, 2011.
- [23] Rong Hu, Brian Mac Namee, Sarah Jane Delany, “Off to a Good Start: Using Clustering to Select the Initial Training Set in Active Learning”, FLAIRS, 2010.
- [24] Yasuhiro Fujiwara, Go Irie, Tomoe Kitahara, “Efficient Approach for Affinity Propagation”, DEIM Forum 2012, March, 2012.
- [25] Daiki Noguchi, Tatsuya Mori, Yota Egusa, Kazuya Suzuki, Shigeki Goto, “Discriminating DRDoS Packets using Time Interval Analysis”, Proceedings of the APAN Research Workshop, 2017.
- [26] US-CERT, “Alert (TA14-017A) UDP-Based Amplification Attacks”, <https://www.us-cert.gov/ncas/alerts/TA14-017A>, December, 2017.
- [27] Team Cymru, “Ephemeral Source Port Selection Strategies”, <https://www.cymru.com/jtk/misc/ephemeralports.html>, August, 2015.