遺伝的アルゴリズムに基づいた広域スキャンのフィンガープリント特定技術の提案

田中 智[1,2,a)]　韓 燦洙[1]　高橋 健志[1]　藤澤 克樹[2]

概要：インターネット上の到達可能かつ未使用の IP アドレス空間（ダークネット）を利用し，新興のマルウェア活動を検知することは，迅速なサイバーセキュリティ対策を行うために必要不可欠である．しかし，巧妙な攻撃者による分散スキャンと調査目的スキャンを区別することは非常に難しい．既存研究では，スキャン対象のポートや送信元ホストの分布に着目することで，攻撃者によるスキャン活動の検知を試みているが，緻密に組織化されたスキャン活動の特定には至っていない．一方，スキャンパケットには他の通信と区別するための特徴（フィンガープリント）が埋め込まれていることが既存研究で知られている．本稿ではフィンガープリントを論理式で表現し，遺伝的アルゴリズムを応用することで，複雑な特徴（論理式）を捉える手法を初めて提案する．ダークネットトラフィックを用いた実験では，既存及び未知の論理式の特定に成功した．論理式を満たすパケットを分析することで，複数の脆弱性を狙った複数ホストによるスキャン活動を確認するとともに，それらは中規模以下のスキャナ郡によって行われることを確認した．

キーワード：ダークネットトラフィック，遺伝的アルゴリズム，フィンガープリント

Proposing a Genetic Algorithm Approach for Unveiling Fingerprint of Internet-Wide Scanner

Akira Tanaka[1,2,a)]　Chansu Han[1]　Takeshi Takahashi[1]　Katsuki Fujisawa[2]

Abstract: Detection of malware activities using darknet traffic is essential to perform prompt cybersecurity measures. However, distributed malware scans are indistinguishable from scan activities for investigative purposes. On the other hand, existing research has revealed that scan packets have their identifier to specify their scan packets from other traffic data. Therefore, this paper represents an identifier as a boolean formula and specifies the identifier based on the genetic algorithm, which is the first research to the best of our knowledge. Numerical experiments using darknet traffic revealed both existing and unknown boolean formulas. We also confirmed some middle- or low-rate port scans targeting multiple vulnerabilities by analyzing packets satisfying the boolean formulas.

Keywords: darknet traffic, genetic algorithm, fingerprint

1. はじめに

ポートスキャンはネットワークに接続しているサーバー上で開いているポートを確認する作業であり，様々な用途に使用される．インターネットに接続された機器の検索エンジンである Shodan や Censys[*1] はポートスキャンを実行することで，脆弱性が見つかった機器の把握や脆弱性が悪用された際の被害推定に役立つ．システムにセキュリティ上の脆弱性が存在するかどうかを確認するペネトレーションテストでは，最初の工程でポートスキャンが実行され，

[1] 国立研究開発法人情報通信研究機構
National Institute of Information and Communications Technology
[2] 九州大学
Kyushu University
[a)] tanaka.akira@nict.go.jp

[*1] https://www.shodan.io/, https://censys.io/

© 2021 Information Processing Society of Japan

開放されているポートで実行されているサービスを調べる．

そうした正当な使い方とは対照的に，稼働サービスの調査及びその脆弱性を悪用する目的でポートスキャンが実行される場合がある．侵入検知システム（IDS）やファイアウォールは特定の IP アドレスもしくはネットワーククラスから到達するパケット数を監視することで，攻撃を検知する．しかし，ホストを分散させて低レートでスキャンを行う狡猾な攻撃者は検知を逃れる．Durumeric ら [4] はポートスキャンの実態調査を行い，分散型のボットネットによるスキャン活動を確認した．それらの悪意あるスキャンは，正常な通信や調査目的のスキャン，バックスキャッタ [3] に紛れ込んでしまい，特定することが難しい．Robertson ら [12] は連携したホストは同じサブネットに属するという仮定をおいて，悪意のある分散型スキャンの検知を試みた．宛先ポート番号や宛先 IP アドレスに着目した Yegneswara [15] らは不正アクセスの大部分は少数のホストによって行われていることを明らかにした．Blaise ら [2] は統計的手法を用いてポートの変化検知を行うことで，新興のボットネットを早期検知した．

他方，スキャンパケットとバックスキャッタを区別するための特徴（フィンガープリント）がパケットに見られることが既存研究 [8] で明らかになった．送信元ホストごとにフィンガープリントを作成するのは非効率であるため，攻撃者は同じフィンガープリントを用いてスキャン活動を行っていると考えられる．多くの IDS や Snort, Suricata[*2]は既知のフィンガープリントと照合することで攻撃を検知する．フィンガープリントは表現能力が高く解釈が容易である反面，手動作成のコストや未知の脅威を検知できないという大きな問題を抱えている．この問題を解決するために，Griffioen ら [8] は TCP 及び IP ヘッダのフィールドからフィンガープリントを特定する手法を提案した．分散型スキャンや低レートスキャンを想定した実験によって，彼らの手法が様々な状況下においても有効であることが確かめられた．

しかし，既存手法の表現能力に乏しいフィンガープリントでは，複雑なフィンガープリントを持ったマルウェア活動を検知することができない．その問題を解決するために，論理式によって表現された柔軟なフィンガープリントを自動で生成し，真のフィンガープリントを自動で特定する遺伝的アルゴリズムに基づいた手法を提案する．我々の知る限り，柔軟なフィンガープリントを自動で生成するアルゴリズムを提案した初めての論文である．生成されたフィンガープリントは，脆弱性の悪用を企むスキャン活動の特定や分析に用いられる．まとめると，本稿の貢献は以下の 3 点である．

(1) IPv4/TCP ヘッダのフィールドに埋め込まれたフィンガープリントを特定する手法を遺伝的アルゴリズムに基づき提案した．

(2) ダークネットトラフィックを用いた実験によって，既存のフィンガープリントだけでなく未知のフィンガープリントの特定に成功した．

(3) フィンガープリントを持つパケットを解析することで，複数の脆弱性を狙ったスキャン活動を確認すると同時に，それらのスキャン活動は中規模以下のスキャナを用いて行われていることを確認した．

[*2] https://www.snort.org/, https://suricata.io/

2. 提案手法

提案手法では，遺伝的アルゴリズムに基づきフィンガープリント（論理式）の候補を初めに生成する．フィンガープリントの候補は IPv4 ヘッダと TCP ヘッダのフィールド上の演算によって定義される．その後，生成された候補から真のフィンガープリントを特定する．後節ではフィールドの説明を 2.1 節で行い，既知のスキャナの論理式を節 2.2 で紹介し，提案アルゴリズムを 2.3 節で説明する．

2.1 TCP/IPv4 ヘッダのフィールド

図 1 に示す通り，IPV4 と TCP ヘッダのフィールドは（1）ユーザーが任意に修正可能なフィールド（黄色）と（2）修正できないフィールド（白色）に大別できる．修正できないフィールドとは値の変更によって IPv4 や TCP の正常動作が妨げられるフィールドである．例えば，IPv4 のバージョンを変更すると，宛先ホストでパケットのパースが正しく行われない．そのため，パケットに埋め込まれるフィンガープリントは修正可能なフィールドのみから生成されていると考えることができる．

2.2 既知のフィンガープリント

TCP パケット全体の集合を $\mathscr{P}$，バイナリ全体の集合を $\mathscr{B}$ で表記する．TCP パケットの入力に対し，バイナリの出力する関数 $f : \mathscr{P} \to \mathscr{B}$ を TCP 写像と定義し，TCP 写像全体の集合を $\mathscr{F}$ と表記する．例えば，TCP パケット p を入力し，パケットの送信元 IP アドレスを出力する関数は TCP 写像であり，$f(p) = \text{ip.srcaddr}$ と表記する．TCP 写像 f とバイナリ b のペア (f, b) をサインと定義する．TCP パケット p が $f(p) = b$ を満たす時，TCP パケット p はサイン (f, b) を持つと言う．また，TCP パケット p がサイン (f, b) を持つか否かを表す定数 $x_{p,(f,b)}$ を以下で定義する．

$$x_{p,(f,b)} = \begin{cases} 1 & \text{パケット } p \text{ がサイン } (f,b) \text{ を持つ} \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

この時，調査目的やマルウェアのスキャン活動を表す論理式は $x_{p,(f,b)}$ を用いて与えられる．既存研究 [9], [10] によると，「ウィンドウサイズが 14600」かつ「シーケンス番号

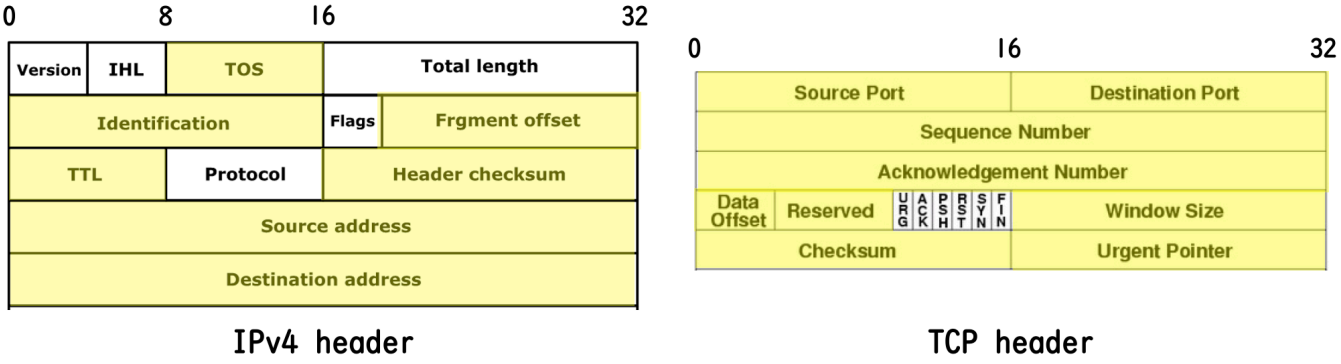

図 1 IPv4 と TCP ヘッダのフィールド．黄色はユーザーが任意に修正可能なフィールドで白色は修正できないフィールド．

の上位もしくは下位の 1 バイトが 0」を満たす SYN パケットはマルウェア Hajime [11] によるスキャン活動だと考えられ，対応する論理式は以下で与えられる．

$$x_{p,(f_1,b_1)} \wedge \left(x_{p,(f_2,0)} \vee x_{p,(f_3,0)}\right) = 1 \qquad (2)$$

ここで，f_1, f_2, f_3, b_1 の定義とその他のスキャン活動の論理式は表 1 にまとめる．

2.3 スキャナの論理式の特定

本節では，スキャナの論理式を特定する方法について述べる．手順全体の流れを以下で示し，擬似コードを Algorithm 1 に記載する．

(1) TCP 写像の生成
遺伝的アルゴリズムに基づき，現状の TCP 写像から新しい TCP 写像を生成する操作を繰り返す．2.3.1 節で TCP 写像の初期化について説明し，TCP 写像の生成手順については 2.3.2 節，遺伝的アルゴリズムとの比較については 2.3.3 節で扱う．

(2) 有効サインの特定
サイン (f, b) を持つパケットが頻繁に観測されるとき，$x_{p,(f,b)}$ はスキャナの論理式に含まれていると考えられる．そうしたサイン (f, b) を有効サインと定義し，有効サインの特定方法について 2.3.4 節で説明する．

(3) スキャナの論理式の特定
有効サイン同士の共起度（パケットが同時に複数の有効サインを持つ頻度）を計算する．共起度が大きい有効サインの集約及び整理を行い，スキャナの論理式を作成する．

観測されるパケットからこれまで特定した論理式を満たすパケットを除き，上記（2），（3）の手順を再実行することで，新たなスキャナの論理式を獲得する．

2.3.1 TCP 写像の初期化

提案手法では，現状の TCP 写像の集合から新たな TCP 写像を作成するため，TCP 写像の集合の初期化についてこの節で説明する．ユーザーが任意に修正可能なヘッダの

Algorithm 1: Scanner's Boolean Formula Finder

Input : n_TCPfunction：TCP 写像の生成数
$P = \{p\}_p$：観測された SYN パケット
$F_{\text{init}} \subseteq \mathscr{F}$：TCP 写像の集合の初期値
Output: BF：論理式の集合

```
Function boolean_formula_finder(n_candidate, P, F_init):
    // (1)TCP 写像の生成
    F ← F_init // TCP 写像の初期化（2.3.1 節）
    τ_cnt(f) (∀f ∈ F) を定義（2.3.2 節）
    for i ← 1 to n_TCPfunction do
        // Algorithm 2 を参照
        f ← generate_TCPfunction(F, τ_cnt)
        F.add(f)
    // (2) 有効サインの特定（2.3.4 節）
    E ← ∅
    for f ∈ F do
        E' ← find_effective_sign(f,P) // Algorithm 3
        E ← E ∪ E'
    // (3) スキャナの論理式の特定
    BF ← 有効サインの集約及び整理を行い，論理式を特定
    return BF
```

フィールド（図 1 上の黄色）を出力する TCP 写像を TCP 写像素と定義し，TCP 写像素の集合を F_{origin} で表す．例えば，パケットを入力した時にシーケンス番号を出力する関数は TCP 写像素である．ヘッダのフィールドのいくつかは大多数のパケットに対して，同じバイナリ値をとる．例えば，TCP ヘッダの緊急ポイントは 99.97%が 0 である．そうしたフィールドはスキャナの論理式の構成要素として不適切であるため，F_{origin} から予め除外する．最後に，フィールドの特定の位置にあるバイト列を出力する TCP 写像を F_{origin} に追加することで，TCP 写像の集合の初期値 F_{init} を獲得する．

$$F_{\text{init}} \coloneqq \bigcup_{f \in F_{\text{origin}}} \{f, f_{\text{F2B}} \circ f, f_{\text{L2B}} \circ f\} \qquad (3)$$

表 1 有名なオープンソースの調査目的スキャンである Masscan と ZMap 及びマルウェアの Mirai と Hajime の論理式 [4], [9]．表中の $f_{\mathrm{FB}}(\cdot)$ と $f_{\mathrm{LB}}(\cdot)$ は入力の上位と下位の 1 バイトを出力する関数である．同様に $f_{\mathrm{L2B}}(\cdot)$ は入力の下位 2 バイトを出力する関数である．$\oplus$ はビット毎の排他的論理和を表す．

Name	論理式	TCP 写像 f	バイナリ b
Hajime [11]	$x_{p,(f_1,b_1)} \wedge \left(x_{p,(f_2,0)} \vee x_{p,(f_3,0)}\right)$	f_1 =tcp.window $f_2 = f_{\mathrm{FB}}$(tcp.seq) $f_3 = f_{\mathrm{LB}}$(tcp.seq)	$b_1 = 14600$
Massscan [7]	$x_{p,(f_4,0)}$	f_4 = ip.id $\oplus f_{\mathrm{L2B}}$(ip.dstaddr) $\oplus$ tcp.dstport $\oplus$ f_{L2B}(tcp.seq)	
ZMap [5]	$x_{p,(f_5,b_2)}$	f_5 = ip.id	$b_2 = 54321$
Mirai [1]	$x_{p,(f_6,0)}$	f_6 = ip.seq $\oplus$ ip.dstaddr	

ここで $f \circ g$ は f と g の合成写像を表し，$f_{\mathrm{F2B}}(\cdot)$ と $f_{\mathrm{L2B}}(\cdot)$ は入力の上位と下位の 2 バイトを出力する関数を表す．また，F_{init} の要素を初期 TCP 写像と定義する．

2.3.2 TCP 写像の生成

提案手法では，現状の TCP 写像の集合 F に対して，確率 r で (a) 特徴抽出，確率 $1-r$ で (b) 二項演算を適用することで，新たな TCP 写像を生成する．ここで $r \in [0,1]$ は (a) と (b) の操作の優先度を決定するハイパーパラメータである．(a) 特徴抽出は以下の一連の操作からなる．

(1) TCP 写像 $f \in F$ を選択
(2) （事前に用意した）バイナリが入出力となる関数の集合 K からランダムに関数 $k : \mathscr{B} \to \mathscr{B}$ を選択
(3) 新規 TCP 写像を $k \circ f$ で定義

例えば，TCP 写像 f = ip.dstaddr と $k = f_{L2B}$ が選択された場合， 新規 TCP 写像 $k \circ f = f_{L2B} \circ$ ip.dstaddr は宛先 IP アドレスの下位 2 バイトを出力する関数である．一方，(b) 二項演算の操作手順は以下である．

(1) 2 つ TCP 写像 $f, g \in F$ を選択
(2) (事前に用意した)$\mathcal{F}$ 上の二項演算の集合 Ψ からランダムに関数 $\psi : \mathscr{F} \times \mathscr{F} \to \mathscr{F}$ を選択
(3) 新規 TCP 写像を $\psi(f, g)$ で定義

例えば，f = ip.seq，g = ip.dstaddr，ψ の出力が 2 つの TCP 写像による像の排他的論理和である場合，新規 TCP 写像 $\psi(f, g)$ = ip.seq $\oplus$ ip.dstaddr はシーケンス番号と宛先 IP アドレスのビット毎の排他的論理和を返す関数となる．生成過程から明らかなように，任意の TCP 写像 f は初期 TCP 写像に複数回別の写像を作用させた関数であり，$\tau_{\mathrm{cnt}} : \mathcal{F} \to \mathbb{N}$（$\mathbb{N}$ は自然数の集合）を以下で定義する．

$$\tau_{\mathrm{cnt}}(f) = \text{特徴抽出の回数+二項演算の回数} + 1 \quad (4)$$

例えば， τ_{cnt} (ip.seq $\oplus$ ip.dstaddr) $= 2$ である．また，(a) 特徴抽出と (b) 二項演算において TCP 写像を選択する際には，Algorithm 2 の select_TCP_function にある通り，$\tau_{\mathrm{cnt}}(f)$ の値が小さい TCP 写像 f が選ばれるようにする．そうすることで，より単純な TCP 写像を優先して生成することができる．TCP 写像の生成方法の詳細を Algorithm 2 に示す．

Algorithm 2: Generate TCP function

Input : $F \subseteq \mathscr{F}$: TCP 写像の集合
$\tau_{\mathrm{cnt}} : \mathscr{F} \to \mathbb{N}$: 式 (4) で定義
K : 入出力がバイナリとなる関数の集合
（つまり，$k \in K \Rightarrow k : \mathscr{B} \to \mathscr{B}$）
Ψ : $\mathscr{F}$ 上の二項演算
（つまり，$\psi \in \Psi \Rightarrow \psi : \mathscr{F} \times \mathscr{F} \to \mathscr{F}$）
$r \in [0,1]$: 特徴抽出と二項演算の優先度
Output: $E = \{(f, b)\}$: 有効サイン

```
Function select_TCP_function(F, τ_cnt):
    f ← 確率 (1/τ_cnt(f))^2 / Σ_{f∈F}(1/τ_cnt(f))^2 で f ∈ F を選択
    return f
Function feature_extraction(F, τ_cnt):
    f ← select_TCP_function(F, τ_cnt)
    k ← k ∈ K を一様分布に従って取得
    return k ∘ f
Function binary_operation(F, τ_cnt):
    f ← select_TCP_function(F, τ_cnt)
    g ← select_TCP_function(F, τ_cnt)
    ψ ← ψ ∈ Ψ を一様分布に従って取得
    return ψ(f, g)
Function generate_TCPfunction(F, τ_cnt):
    f ← f ∈ F を一様分布に従って取得
    while f ∈ F do
        x ∼ Uniform(0,1) // [0,1] 上の一様乱数
        if x ≤ r then
            f ← feature_extraction(F, τ_cnt)
        else
            f ← binary_operation(F, τ_cnt)
    τ_cnt(f) を定義
    return f
```

2.3.3 TCP 写像の生成と遺伝的アルゴリズムの比較

メタヒューリスティックの 1 つである遺伝的アルゴリズムは，エネルギー配分・素材産業・進化的計算など諸分野で直面する最適化問題に対して，優れた解を求めるために広く利用される [14]．遺伝的アルゴリズムでは個体と呼ばれる解の集合に対して，選択・交叉・突然変異といった生物の適応進化を模した操作を繰り返し適用することで，より良い解を探索する．各反復では個体の適応度を計算し，適応度が高い個体に対して交叉や突然変異を適用することで，新しい個体を生成する．提案モデルでは，TCP 写像が個体に対応し，適応度は τ_{cnt} によって評価される．より単純な TCP 写像の生成を目指すため，τ_{cnt} の値が小さい TCP 写像がより選ばれるようにしている．突然変異と交叉に対応する操作は (a) 特徴抽出と (b) 二項演算であり，個体の特徴を受け継いだ新しい個体を生成する．

2.3.4 有効サインの特定

この節では，生成された TCP 写像 f の出力分布に着目し，論理式の構成要素である有効サイン (f, b) を特定する方法について述べる．

f を TCP 写像，$P = \{p\}_p$ を観測された SYN パケットの集合とする．任意のバイナリ $b \in \mathscr{B}$ に対して，出現頻度 $r(b)$ を以下で定義する．

$$r(b) \coloneqq \frac{\#\{p \in P \mid f(p) = b\}}{\#P} \tag{5}$$

ここで，$\#A$ は集合 A の要素数を表す．また，f による $P = \{p\}_p$ の像 $B \coloneqq f(P)$ は以下で定義される．

$$B \coloneqq f(P) = \{f(p) \mid p \in P\} \subseteq \mathscr{B} \tag{6}$$

多重集合 $R \coloneqq \{\{r\left(f(p)\right)\}\}_{p \in P}$ を定義する．任意の実数 α に対して，$R_{<\alpha} \subseteq R$ $(R_{\leq\alpha} \subseteq R)$ を $r < \alpha$ $(r \leq \alpha)$ を満たす $r \in R$ の全てから構成される集合と定義する．また，多重集合 A の母分散を σ_A^2 で表す．この時，TCP 写像 f に対するバイナリ $b \in B$ の有効指標 $e_f(b)$ を以下で定義する．

$$e_f(b) \coloneqq \begin{cases} \sigma^2_{R_{\leq r(b)}} / \sigma^2_{R_{<r(b)}} & (\text{if } \sigma^2_{R_{<r(b)}} > 0) \\ \text{Not defined} & (\text{otherwise}) \end{cases} \tag{7}$$

有効指標 $e_f(b)$ はバイナリ b が出現頻度の分散に与える影響を数値化した関数で，$e_f(b)$ の値が大きい場合は (f, b) が有効サインであることを示唆している．有効サイン特定の擬似コードを Algorithm 3 に記載する．

Algorithm 3: Find Effective Sign

Input : f : TCP 写像
$P = \{p\}_p$: 観測された SYN パケット
max_sign : TCP 写像あたりの最大有効サインの数
sign_thres : 有効サインの閾値
Output: $E = \{(f, b)\}$: 有効サインの集合

```
Function find_effective_sign(f, P):
    B ← {f(p) | p ∈ P}
    sorted_B ← r(b) に対する降順で B を並び替え
    max_idx ← NULL
    for i ← 0 to max_sign −1 do
        b ← B[i]
        if e_f(b) > sign_thres then
            max_idx ← i
    E ← ∅
    if max_idx ≠ NULL then
        for i ← 0 to max_idx do
            E.add((f, B[i]))
    return E
```

3. 実験結果と評価

提案手法の有用性を示すために，ダークネットトラフィックに適用した．実験で用いたデータセットについて 3.1 節で説明し，3.2 節でパラメータ設定について述べ，3.3 節で論理式を満たすパケットの分析を行う．

表 2 実験で使用したダークネットのトラフィックデータ

	IP 数 §	観測期間 (2018 年)*	パケット数
提案手法の適用	4,096	10/22 – 10/24	572,289†
パケットの解析	4,096	10/22 – 10/28	117.5×10^6

* 活発なマルウェア活動が観測された期間 [9], [10] で検証した．
§ 観測 IP アドレス数
† 計算時間削減のため，1 時間分に相当するパケットをランダムに選んだ．（期間中の総パケット数は 41.2×10^6）

3.1 実験データ

実験では NICTER[*3]が運用しているダークネット観測網で収集された SYN パケットを用いた．ダークネットとは，インターネット上で到達可能かつ未使用の IP アドレス空間のことを指しており，正常な通信が観測されないという特性から，不正なスキャン活動の監視が可能である．また，SYN パケットはアクティブなホストや開いているポートを調査する際に用いられる [8]．提案手法を適用した際に用いたデータセット及び論理式を満たすパケットを解析する際に用いたデータセットについて表 2 に記載する．

3.2 パラメータ設定

Algorithm 1 では n_TCPfunction=2000 に設定した．2.3.1 節に現れる F_{origin} は以下で定義した．

[*3] Network Incident analysis Center for Tactical Emergency Response : `https://www.nicter.jp/en`

$$F_{\text{origin}} = \{\text{ip.id}, \text{ip.header\_checksum}, \text{ip.srcaddr}\} \quad (8)$$
$$\cup \{\text{ip.dstaddr}, \text{tcp.srcport}\} \quad (9)$$
$$\cup \{\text{tcp.dstport}, \text{tcp.seq}, \text{tcp.window}\} \quad (10)$$

Algorithm 2 では $r = 0.1$, $K = \{f_{F2B}, f_{L2B}\}$ and $\Psi = \{\psi\}$ を用いた．ただし，$\psi : \mathscr{F} \times \mathscr{F} \to \mathscr{F}$ は任意のパケット p と TCP 写像 $f, g \in \mathscr{F}$ 対して，$\psi(f, g)(p) \mapsto f(p) \oplus g(p)$ を満たす $\mathscr{F}$ 上の二項演算である．Algorithm 3 では max_sign = 10 とし，sign_thres の値は有効サインの数が 20〜50 になるように適切な値を用いた．

Algorithm 1 の計算時間は n_TCPfunction と観測した SYN パケット $P = \{p\}_p$ に比例する．また，ハイパーパラメータである F_{origin}，K，Ψ は TCP 写像の探索範囲を規定する．max_sign や sign_thres は有効サインの判定基準を定めるハイパーパラメータである．なお，上記全てのハイパーパラメータは実験を行いながら経験則に基づいて決定した．

3.3 論理式を持ったパケットの解析

2.3 節の手順に従い，「(2) 有効サインの特定」及び「(3) スキャナの論理式の特定」を繰り返し行うことで 9 個の論理式を獲得した（既知の論理式 3 つ，未知の論理式 6 つ）．表 3 に記載しているように，未知の論理式のいずれもパケット全体に占める割合は 1%未満であり，特定が困難なパケットの特徴を捉えられた．既知の論理式のうち唯一特定できなかったものは Hajime であり，それは Hajime が全パケットの 0.02%しか占めておらず，主要な攻撃と見なされていないからだと考えられる．論理式を分析するために，ポート番号の調査や以下の 2 つの IP アドレスのリストと照合を行い，リストに記載された IP アドレスを送信元としたパケットの数を調べた．

- 調査目的スキャナのリスト
  主にドメイン名を調査することで取得した独自のリストで，674 個の IP アドレスが含まれる．
- 大規模スキャナのリスト
  遠藤らの先行研究 [16] にならい，「1 日の宛先ポート番号のユニーク数が 30 以上」かつ「1 日のパケット数が IP アドレス空間全体の数を上回る」（観測 IP から概算した値）ものを大規模スキャナと呼び，2018 年 10 月の少なくとも 1 日で基準を満たす 325 個の IP アドレス．

特定した論理式は表 3 のように攻撃もしくは調査目的に大別した．ここでは，攻撃目的と調査目的の論理式を満たすパケットの大きな違いについて述べ，個別の論理式の解析については次節以降に委ねる．

- 攻撃目的
  - 脆弱性が存在するポート番号を狙ったパケットが多く見られる．
  - 調査目的・大規模スキャナを送信元とするパケットがほとんど観測されなかった．特に Attack1〜5 では全く一致しなかった．
- 調査目的
  - 宛先ポート番号の偏りはあまり見られない．
  - 調査目的及び大規模スキャナを送信元とするパケットが攻撃目的に比べて多く見られる．

3.3.1 攻撃目的の論理式を持つパケットの解析

表 3 から分かるように，既知のマルウェアである Mirai は提案手法でも特定することができた．Mirai の特徴として定常的にパケットが観測できること・宛先ポート番号に大きな偏りがあることが確認された．例えば，Mirai のパケットの 52.8%が 23 番ポート，6.0%が 2323 番ポートであり，これは Mirai が総当たりで不正ログインを試みる際に使用されるポート番号である．また，6.0%を占める 5555 番ポートは Mirai の亜種がアンドロイド端末を感染させる際に用いるポート番号である．

表 3 の Attack1 は宛先ポート番号が 5431 番, 送信元ポート番号が 6 番，tcp.window=65535，tcp.seq=tcp.ack=0 を満たすパケットから，ZMap のフィンガープリントである ip.id=54321 とスパイク（短期間での大量のパケット観測）を引き起こす送信元 IP アドレス 110.249.212.46 を除いたものである．宛先ポート番号 5431 番はブロードコムの UPnP (Universal Plug and Play) のハイジャックを狙った大規模スキャン活動であることが知られている．

表 3 の Attack2・Attack3 は共に ip.id = 256, tcp.window = 16384，tcp.seq の最後の 2 バイトが 0 という特徴をもったパケットから，スパイクを引き起こす送信元 IP アドレスを除いたものである．Attack2 は送信元ポート番号が 6000 番で固定だが，Attack3 は送信元ポート番号が 6000 番以外の全ての値をとる．宛先ポート番号には共通した偏りが見られ，Attack2 の 22.2%及び Attack3 の 15.9%が 3306 番, Attack2 の 15.2%及び Attack3 の 20.6%が 60001 番，Attack2 の 14.6%及び Attack3 の 18.4%が 1433 番である．3306 番は MySQL の脆弱性を狙っている [6] と考えられ，1433 番は Microsoft's SQL Server のデーターベースの不正操作を目的としたポートスキャン [2] だと考えられる．60001 は Mirai の亜種である fbot ファミリーが Jaws Web Server の脆弱性を狙ったものであると考えられる[*4]．

表 3 の Attack4 は断続的にパケットが観測された論理式であり，送信元ポート番号が全て 80 番，宛先ポート番号は全てダイナミックポート（49152〜65535）である．

Attack5 の論理式中の $\neg x_{(f_{10}, b_{10})}$ は一番大きなスパイクを引き起こす送信元 IP アドレスを除くために追加した．Attack5 も Attack4 と同様に宛先ポート番号が全てダイナ

[*4] https://sect.iij.ad.jp/blog/2020/02/mirai-2019/

表 3 提案手法によって特定した論理式．パケット (%) はパケット全体に占める割合を表す．

用途	名前	パケット (%)	論理式	TCP 写像	バイナリ
攻撃	Mirai	14.31%	$x_{(f_6,0)} \wedge x_{p,(\text{tcp.ack},0)}$	$f_6 = f_{\text{L2B}}(\text{ip.dstaddr}) \oplus \text{tcp.seq}$	
	Attack1	0.37%	$x_{p,(\text{tcp.dstport},5431)} \wedge x_{p,(f_{11},6)}$ $\wedge x_{p,(f_7,b_3)} \wedge x_{p,(\text{tcp.seq},0)} \wedge x_{p,(\text{tcp.ack},0)}$ $\wedge \neg x_{p,(f_{10},b_7)} \wedge \neg x_{p,(\text{ip.id},b_2)}$	$f_7 = \text{tcp.window}, f_{10} = \text{ip.srcaddr}$ $f_{11} = \text{tcp.srcport}$	$b_2 = 54321$ $b_3 = 65535$ $b_7 = 1861866542$ (IP=110.249.212.46)
	Attack2	0.26%	$x_{p,(\text{ip.id},256)} \wedge x_{p,(f_7,16384)}$ $\wedge x_{p,(f_{\text{L2B}}(\text{tcp.seq}),0)} \wedge x_{p,(\text{tcp.ack},0)}$ $\wedge x_{p,(\text{tcp.srcport},b_6)} \wedge \neg x_{p,(f_{10},b_{11})}$	$f_7 = \text{tcp.window}, f_{10} = \text{ip.srcaddr}$	$b_6 = 6000$ $b_{11} = 3722824824$ (IP=221.229.204.120)
	Attack3	0.41%	$x_{p,(\text{ip.id},256)} \wedge x_{p,(f_7,16384)}$ $\wedge x_{p,(f_{\text{L2B}}(\text{tcp.seq}),0)} \wedge x_{p,(\text{tcp.ack},0)}$ $\wedge \neg x_{p,(\text{tcp.srcport},b_6)} \wedge \neg x_{p,(f_{10},b_{12})}$	$f_7 = \text{tcp.window}, f_{10} = \text{ip.srcaddr}$	$b_6 = 6000$ $b_{12} = 1867066988$ (IP=111.73.46.108)
	Attack4	0.36%	$x_{(f_9,0)} \wedge x_{p,(\text{tcp.ack},1)}$ $\wedge x_{p,(\text{ip.id},0)} \wedge x_{p,(f_7,b_8)} \wedge x_{(f_{11},80)}$	$f_9 = f_{\text{L2B}}(\text{ip.dstaddr}) \oplus \text{tcp.dsrport}$ $f_7 = \text{tcp.window}, f_{11} = \text{tcp.srcport}$	$b_8 = 17520$
	Attack5	0.23%	$x_{(f_9,0)} \wedge x_{p,(\text{tcp.ack},1)}$ $\wedge x_{p,(\text{ip.id},b_9)} \wedge \neg x_{(f_{10},b_{10})}$	$f_9 = f_{\text{L2B}}(\text{ip.dstaddr}) \oplus \text{tcp.dsrport}$ $f_{10} = \text{ip.srcaddr}$	$b_9 = 38993$ $b_{10} = 3324836372$ (IP=198.44.250.20)
	Attack6	0.07%	$x_{p,(\text{tcp.window},1300)} \wedge x_{p,(\text{tcp.ack},0)}$		
調査	ZMap	8.04%	$x_{p,(\text{ip.id},b_2)} \wedge x_{p,(\text{tcp.ack},0)} \wedge x_{p,(f_7,b_3)}$	$f_7 = \text{tcp.window}$	$b_2 = 54321$ $b_3 = 65535$
	Massscan	67.14%	$x_{p,(f_4,0)} \wedge x_{p,(\text{tcp.ack},0)}$	$f_4 = \text{ip.id} \oplus f_{\text{L2B}}(\text{ip.dstaddr})$ $\oplus \text{tcp.dstport} \oplus f_{\text{L2B}}(\text{tcp.seq})$	

ミックポート（49152〜65535）である．期間全体を通して，送信元ポート番号に大きな偏りは見られなかったが，3 回のスパイク時には大きな偏りが見られた．1，2 回目のスパイク時には特定の送信元ポート番号のみが使用され，3 回目のスパイク時には 80 番が 98.3%使用された．

表 3 の Attack6 は断続的なパケットが観測され，宛先ポート番号の 31.0%が 80 番，26.9%が 8080 番，26.8%が 85 番，14.9%が 443 番である．NICTER 観測レポート 2018[*5] によると，宛先ポート 80 番，443 番及び 8080 番は GPON のホームルーターを狙った攻撃だと考えられる．

3.3.2 調査目的の論理式を持つパケットの解析

表 3 の ZMap は従来のフィンガープリントである ip.id=54321 にウインドウサイズが 65535 かつ確認応答番号が 0 という条件を追加したものである．元のフィンガープリントを満たすパケットのほとんど全てが追加した条件も満たすため，ZMap の論理式を厳密にしても問題ないと考えられる．調査目的スキャナのリストと，Censys では 64，Binaryedge では 12，Security.ipip.net では 21，Shadowserver では 166 個の IP アドレスで重複があった．また，Attack6 のパケットの 9.42%を調査目的スキャナが占めており，22.16%を大規模スキャナが占めていた．図 2 から見てとれるように，定常的なパケットと 3 回のスパイクが観測された．スパイクが観測された時間帯に絞ってパケットを分析した結果，3 回のスパイク全てが，少数の送信元 IP アドレスによる調査目的の大規模スキャンであった．宛先ポート番号では，81/TCP (8.8%)，22/TCP (4.4%) など多少の偏りはあるものの，脆弱性を狙ったスキャン活動ではないと考えられる．送信元ポート番号に関しては偏りがほとんど見られなかった．

表 3 の Masscan は定常的なパケットが観測されており，調査目的スキャナのリストと Binaryedge では 1，Censys では 2，Onyphe では 1，Security.ipip.net では 3，Shadowserver では 11，Shodan では 6，Suspicious では 1 個の IP アドレスと重複していた．また，Masscan のパケットの 77.38%を大規模スキャナが占めていた．

4. まとめ

本稿では，遺伝的アルゴリズムに基づき，パケットのヘッダ情報に埋め込まれたスキャナの論理式を特定する手法を提案した．ダークネットトラフィックを用いた実験では，既存のスキャナのみならず，これまで知られていなかった論理式の特定に成功した．未知の論理式がパケット全体に占める割合は 1%未満であることから，検知が難しいスキャン活動を特定することができたと考えられる．論理式を満たすパケットを解析したところ，複数の脆弱性を狙ったスキャン活動を発見した．また，調査目的スキャナや大規模スキャナのリストと照合することで，調査目的の論理式を満たすスキャナがリストに含まれているだけでなく，攻撃目的では中規模以下のスキャナが広く用いられていることが分かった．今後は，動的解析によって得られる

[*5] http://www.nict.go.jp/cyber/report/NICTER_report_2018.pdf

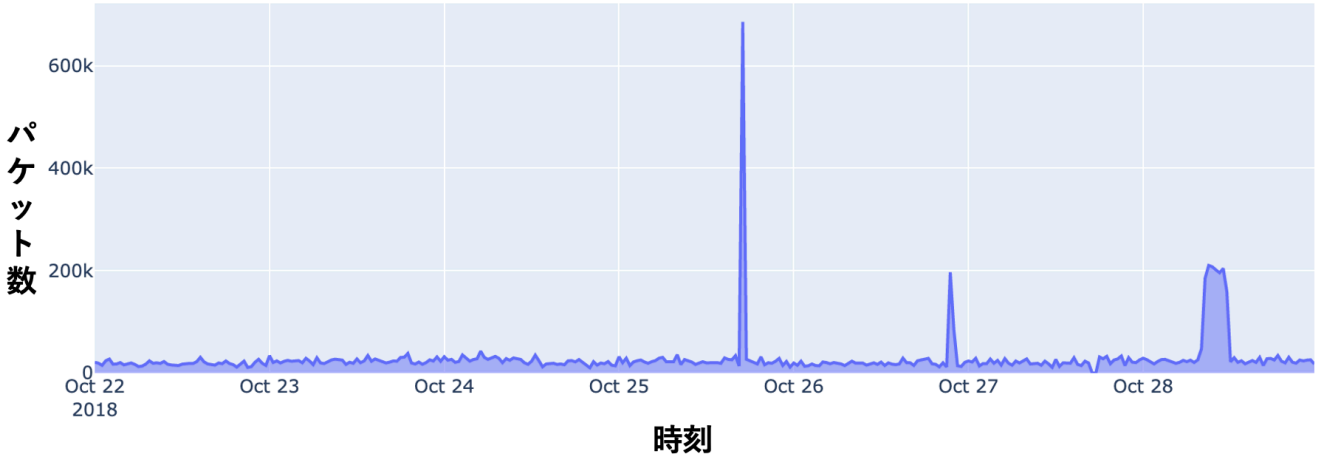

図 2 ZMap の論理式を持つパケット数．縦軸は 30 分毎のパケット数を表す．

マルウェアのパケットから論理式を特定することで，信頼性のある論理式の特定することや，第 3 者の脅威情報等を自動的に関連付ける研究 [13] を進めていきたい．

謝辞 本研究は総務省の「電波資源拡大のための研究開発 (JPJ000254)」における委託研究「電波の有効利用のための IoT マルウェア無害化／無機能化技術等に関する研究開発」によって実施した成果を含む．

参考文献

[1] Antonakakis, M., April, T., Bailey, M. and et al.: Understanding the Mirai Botnet, *26th USENIX Secur. Symp.*, pp. 1093–1110 (2017).

[2] Blaise, A., Bouet, M., Conan, V. and Secci, S.: Detection of zero-day attacks: An unsupervised port-based approach, *Comput. Networks*, Vol. 180, No. January (2020).

[3] Blenn, N., Ghiëtte, V. and Doerr, C.: Quantifying the Spectrum of Denial-of-Service Attacks through Internet Backscatter, *Proc. 12th Int. Conf. Availability, Reliab. Secur.*, ARES '17 (2017).

[4] Durumeric, Z., Bailey, M. and Halderman, J. A.: An internet-wide view of internet-wide scanning, *Proc. 23rd USENIX Secur. Symp.*, pp. 65–78 (2014).

[5] Durumeric, Z., Wustrow, E. and Halderman, J. A.: ZMap: Fast Internet-wide Scanning and Its Security Applications, *Proc. USENIX Secur. Symp.*, pp. 605–620 (2013).

[6] Goseva-Popstojanova, K., Miller, B., Pantev, R. and Dimitrijevikj, A.: Empirical analysis of attackers activity on multi-tier web systems, *Proc. Int. Conf. Adv. Inf. Netw. Appl. AINA*, pp. 781–788 (2010).

[7] Graham, R. D.: MASSCAN: Mass IP port scanner, (online), available from ⟨https://github.com/robertdavidgraham/masscan⟩.

[8] Griffioen, H. and Doerr, C.: Discovering Collaboration: Unveiling Slow, Distributed Scanners based on Common Header Field Patterns, *IEEE/IFIP Netw. Oper. Manag. Symp.*, pp. 1–9 (2020).

[9] Han, C., Shimamura, J., Takahashi, T. and et al.: Real-Time Detection of Global Cyberthreat Based on Darknet by Estimating Anomalous Synchronization Using Graphical Lasso, *IEICE Trans. Inf. Syst.*, Vol. 103, No. 10, pp. 2113–2124 (2020).

[10] Han, C., Takeuchi, J., Takahashi, T. and Inoue, D.: Automated Detection of Malware Activities Using Non-negative Matrix Factorization, *20th IEEE International Conference On Trust, Security And Privacy In Computing And Communications (TrustCom)* (2021).

[11] Herwig, S., Harvey, K., Hughey, G. and et al.: Measurement and analysis of Hajime, a peer-to-peer IoT botnet, *Netw. Distrib. Syst. Secur. Symp.* (2019).

[12] Robertson, S., Siegel, E. V., Miller, M. and Stolfo, S. J.: Surveillance detection in high bandwidth environments, *Proc. DARPA Inf. Surviv. Conf. Expo. DISCEX*, Vol. 1, pp. 130–138 (2003).

[13] Takahashi, T., Umemura, Y., Han, C., Ban, T., Furumoto, K., Nakamura, O., Yoshioka, K., Takeuchi, J., Murata, N. and Shiraishi, Y.: Designing Comprehensive Cyber Threat Analysis Platform: Can We Orchestrate Analysis Engines?, *2021 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops)*, IEEE (2021).

[14] Wang, Z. Z. and Sobey, A.: A comparative review between Genetic Algorithm use in composite optimisation and the state-of-the-art in evolutionary computation, *Compos. Struct.*, Vol. 233 (2020).

[15] Yegneswaran, V., Barford, P. and Ullrich, J.: Internet intrusions: Global characteristics and prevalence, *Perform. Eval. Rev.*, Vol. 31, No. 1, pp. 138–147 (2003).

[16] 遠藤由紀子， 森好樹，島村隼平，久保正樹：ダークネット観測における大規模スキャナの判定指標の提案，信学技報，ICSS2019-80, Vol. 119, No. 437, 沖縄，pp. 73–78 (2020).