

実行時の通信挙動を用いたマルウェアの分類と 未知検体検出への応用

畑田 充弘^{1,2} 森 達哉¹

概要：膨大な数のマルウェアはネットワーク化された現在の社会において大きな脅威となっている。しかしながら基本的な対策の一つとして広く利用されているアンチウイルスソフトでは検知できないマルウェアも多数ある。脅威の把握のために日々大量に収集するマルウェアの中から、新種を効率的に抽出し、優先度を上げて詳細解析や予防措置といった対策をすることが必要となる。本研究では、マルウェアの動的解析によって得られる通信トラフィックを用いて、マルウェアの通信モデルを提案する。また、提案モデルに基づく分類とともに新種のマルウェアを抽出する手法を提案し、その評価結果を報告する。

キーワード：マルウェア, 動的解析, トラフィック, 分類, 未知検知

Finding New Malware Samples with the Network Behavior Analysis

MITSUHIRO HATADA^{1,2} TATSUYA MORI¹

Abstract: An enormous amount of malware samples pose a major threat to recent networked society. Antivirus software and intrusion detection system are widely implemented to the hosts and network as a basic countermeasure. However, it may fail to detect evasive malware. Setting a high priority to new unknown malware samples is necessary to analyze deeply and take preventive measures. In this paper, we present a traffic model of malware that achieve to classify network behaviors of malware and to find new unknown malware samples. Our model consists of malware specific features and general traffic features that are extracted from packet traces obtained by dynamic analysis of malware. We apply DBSCAN, a kind of unsupervised learning, to generate the classifier and evaluate our proposal by using large-scale live malware samples. The results of experiment demonstrate the effectiveness of find new unknown malware samples.

Keywords: Malware, Dynamic Analysis, Traffic, Classification, Unknown Detection

1. はじめに

膨大な数のマルウェア [1] は依然として大きな問題となっており、Potentially Unwanted Program (PUP) と称される Adware 等も含め、ユーザのプライバシーやコンピュータのセキュリティの脅威となっている [2]。アンチウイルスソフトや侵入検知システム、IP アドレスやドメインによるブラックリストといった感染前の検知に重点をおいた

様々な対策がある。感染ホスト上で自己複製の際にコードを改変するポリモーフィックをはじめ、難読化、ゼロデイ攻撃などこれらの感染予防アプローチを回避することも難しくない。感染後の対策に有用な情報を得るためには、マルウェアを解析して挙動を把握することが重要である。マルウェアの解析手法は静的解析と動的解析に大別され、静的解析はプログラムの実行フローを詳細に解析できる反面、解析者の負担が大きい。一方、動的解析は様々なツールやサービス [3], [4] があり、プロセス情報やファイルあるいはレジストリの読み書き、通信挙動、メモリダンプ、デスクトップのスクリーンショットなどが取得でき、マルウェアの挙動を迅速に把握できる。我々は、マルウェアの

¹ 早稲田大学
Waseda University
{m.hatada, mori}@nsl.cs.waseda.ac.jp
² NTT コミュニケーションズ株式会社
NTT Communications Corporation

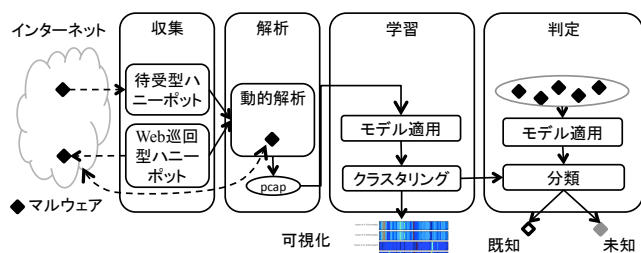


図 1 システム概要

動的解析で得られるデータのうち、解析モジュールの改変等に影響を受けず、マルウェアを実行するホストの外部で取得できる通信データに着目する。

CSIRT (Computer Security Incident Response Team) や SOC (Security Operation Center) にとって、マルウェアの通信挙動は有用な IOC (Indicator Of Compromise) であり、例えば C2 (Command and Control) サーバの IP アドレスやドメイン名が判明すれば同種のマルウェアの感染ホストをネットワーク上で調査できる。また、マルウェアがアクセスする Web サイトやその順番といった通信パターンを SIEM (Security Information and Event Management) のルールとして定義することで、感染ホストを検知できる。近年のマルウェアは通信を行うものが多く、アンチウイルスソフトで検知できない検体 (未知検体) も多い [5]。そのため、未知検体であっても既知検体 (アンチウイルスソフトで検知できる検体) と同様の通信挙動であると判定できれば、多数の未知検体の中でも異なる挙動を示す新種として、詳細解析や対策の優先度を上げることができる。

我々は、マルウェアの通信モデルを定義し、動的解析で得られたマルウェア通信にモデルを適用してクラスタリングを行い、作成されたクラスタによる分類と新種の未知検体を判定する。マルウェアの収集、解析、学習、判定を行うシステム概要を図 1 に示す。本論文の主な貢献は以下の通りである。

- 多数の従来研究とドメイン知識をもとにマルウェア通信特有の特徴量 25 種類と、通信の傾向を把握するための一般的な特徴量 70 種類を組み合わせ、95 種類の特徴量に基づくマルウェアの通信モデルを提案した。
- マルウェア 21,717 検体の動的解析結果をもとに、マルウェアの通信モデルを適用してクラスタリングを行い、その精度を評価して有用性を示した。
- 作成したクラスタをもとに、収集・解析時期の異なる 6,078 検体分のマルウェア通信を分類し、通信挙動のみでマルウェアの分類が可能なことと、効率的に新種の未知検体を発見できることを示した。

本論文の構成は以下の通りである。2 章で提案手法を述べ、3 章で実験に利用するデータセットと実験結果を示す。4 章で関連研究を紹介し、最後に 5 章で結論と今後の課題をまとめる。

2. 提案手法

2.1 マルウェアの通信モデル

マルウェアの通信を表現するために、従来研究 [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15] で利用されてきた有効性が高い特徴量、及びエキスパートによる経験に基づく特徴量からマルウェア特有の通信をモデル化する。また、従来知られていない通信の特徴を持つマルウェアが出現してきた場合にも、その特性が現れやすいような基本的な通信の特徴量もモデルに組み込む。これにより、新種の未知検体の出現を判定した場合には、詳細解析に基づいてマルウェア特有の特徴量として定義を追加し、モデルの更新をしていくことを想定している。なお実装では、動的解析環境で取得した個々のマルウェアのパケットキャプチャファイルから、Wireshark の CLI ツールである TShark [16] を使用し、集計や各種条件との照合を行うことで特徴量を算出する。

2.1.1 マルウェア特有の特徴量

典型的なマルウェアの通信を大別した 9 種類のクラスと、各クラスにおける個々の特徴量を表 1 に示すとともに以下で説明する。

通信開始時間: マルウェアの実装により `Sleep()` や `SleepEx()` の API を使用したり、不要なループ処理を入れることで意図的に処理を遅らせることで、短時間での動的解析を回避する。

インターネット接続確認: インターネットにつながらない解析環境の場合、s その後の動作を止めることで動的解析を回避する。Google や Yahoo のようなよく知られているサイトにアクセスするマルウェアも多い [17] ため、Alexa Top 10K sites [22] を主要な Web サイトのリストとして利用する。また、ホストのグローバル IP アドレスを確認する Web サイトへアクセスするマルウェアも多く、グローバル IP アドレスを確認するサイトとして 10 サイト ([23] 他) を利用する。

情報窃取: 感染ホストのプロファイルを蓄積し、攻撃の用途に応じて悪用したり、解析サービスを特定して回避することを目的に、ホスト名やユーザ名、OS のバージョンなどホストの内部情報を窃取し、HTTP 通信のクエリパラメータとして外部へ送信する [9]。これまでの解析結果をもとに定義したホストの内部情報 16 種類を利用する。

C2 (Command & Control): オンライン状態の確認、各種司令の伝達等を目的とする C2 通信は最もマルウェア特有の挙動であり、従来から様々な研究において C2 通信の検知が取り組まれている。C2 サーバのリスト [24], [25] との一致、CERT 等の対策によりシンクホール [20] されている場合に HTTP レスポンスに付与される "x-sinkhole" ヘッダの有無、C2 に利用されるプロトコル (IRC [18] や P2P [19]) の利用、という 3 種類のアプローチで特徴量を

表 1 マルウェア特有の特徴量 (25 種類)

クラス	特徴量
通信開始時間	マルウェア実行から通信開始までの時間 [秒]
インターネット接続確認	主要な Web サイトへの DNS クエリ数, HTTP リクエスト数 [17], グローバル IP アドレス確認用 Web サイトへの DNS クエリ数, HTTP リクエスト数
情報窃取	ホスト内部情報を含む HTTP リクエスト数 [9]
C2 (Command & Control)	公開ブラックリスト (5 種類) に一致する各セッション数, TCP での IRC セッション数, UDP での IRC セッション数 [18], TCP での P2P セッション数, UDP での P2P セッション数 [19], シンクホールされているホストへのセッション数 [20]
ダウンロード	実行ファイルのダウンロード数
スパムメール送信	MX レコードの DNS クエリ数, SMTP セッション数 [6], [15], [17]
スキャン	内部ホストへの ICMP エコーリクエスト数, 外部ホストへの ICMP エコーリクエスト数 [21]
広告	広告関連のキーワードを含む HTTP リクエスト数
識別	HTTP リクエスト中の異なる User-Agent 数, User-Agent の文字列の長さ [15] の最小値と最大値

抽出する。

ダウンロード: ダウンローダとして本来の機能を有するマルウェアをダウンロードしたり, 機能追加したりするために, 実行ファイルをダウンロードすることはマルウェアの主要な通信挙動と考えられる。HTTP レスポンスの”Content-type”ヘッダが application/exe 等あらかじめ定義した 6 種類を利用する。

スパムメール送信: 感染ホストをスパムメール送信の踏み台として利用する [6], [15], [17]。その際, 主要なドメインの MX レコードを確認し, メールサーバに SMTP 接続を試みる。

スキャン: 同一ネットワーク上のホストの存在確認, あるいは外部の特定のホストへの到達性確認のため, マルウェアは ICMP エコーリクエストを送信する [21]。単一ホストのみで構成する単純な動的解析環境の検知も目的と考えられる。

広告: 広告の表示やクリックによる金銭収入を目的として, Web サイトのコンテンツに広告を埋め込んだりポップアップを表示したりする。これをブロックする Adblock Plus [26] のリストを利用して, 一致する HTTP リクエストを抽出する。

識別: 独自の User-Agent を利用 [15] することで, 攻撃者が用意した Web サーバでマルウェアを識別したり, 頻繁に変更することで検知を回避する意図があると考えられる。

2.1.2 基本的な特徴量

マルウェア特有の通信挙動に限らず, 関連研究 [6], [9], [10], [14], [15], [17], [21] からにも通信パターンを把握する上で有用となる基本的な特徴量を表 2 に示す。多くは一般的な特徴量であるため個々の説明は省略する。

2.2 クラスタリング

マルウェアの通信モデルによって抽出した特徴量をもとに教師なし学習のクラスタリングを行う。教師あり学習

は, 予め各サンプルに出力の正解ラベルを付与する必要がある。多くの関連研究においても, アンチウイルスソフトの検知名を正解ラベルとして利用されることが多い。しかしながら, アンチウイルスソフトで検知できない検体も多く, 妥当なラベル付けを定常的に更新していくこと自体が困難であると考えられるため, 我々は教師なし学習によるクラスタリングを行うことで, 2.1 で示したマルウェアの通信モデルに基づくクラスタを導出する。

前処理として, クラスタリングを行う上で十分とはいえない通信総量のサンプルを除外する。MTU が 46~1,500 バイトであることと, 動的解析環境に依存して観測される NTP サーバの名前解決のための DNS 通信や, ICMPv6 Router Advertisement などがトラフィックに含まれることがあるため, 除外する閾値を 1,000 バイトとした。予備実験により, 各サンプルの通信総量を調査したところ, 約 21%のサンプルが 1,000 バイト以下の通信総量であった。また, 特徴量によって値の差が大きいため, 平均 0, 標準偏差 1 となるよう特徴量毎に標準化を行う。

DBSCAN (Density-Based Spatial Clustering Applications with Noise) [27] は従来よりデータマイニング分野で利用されており, クラスタ数を指定することなく, サンプルの近傍の密度が一定の閾値を超えている場合, クラスタを成長させるため, 任意のクラスタ形状を生成することができる。また, どのクラスタにも属さないサンプルはノイズとしてエラーと扱われる。集合 X のあるサンプル x_p と x_q があり, その間の距離を $D(x_p, x_q)$ とし, 半径 ϵ 以内の近傍に n 個以上のサンプルがある集合を $N_\epsilon(x_p)$ とすると,

$$x_q \in N_\epsilon(x_p)$$

$$|N_\epsilon(x_p)| \geq n$$

$$N_\epsilon(x_p) = \{x_q \in X | D(x_p, x_q) \leq \epsilon\}$$

を満たす時に x_p から x_q へは directly density-reachable であるという。任意のサンプルから directory density-

表 2 基本的な特徴量 (70 種類)

クラス	特徴量
All	通信総量 [bytes]
TCP	セッション数, 宛先ホスト数, 1 セッションの受信サイズの最大値 [bytes] と最小値 [bytes], 1 セッションの送信サイズの最大値 [bytes] と最小値 [bytes]
UDP	セッション数, 宛先ホスト数, 1 セッションの受信サイズの最大値 [bytes] と最小値 [bytes], 1 セッションの送信サイズの最大値 [bytes] と最小値 [bytes]
ICMP	エコーリクエスト先のホスト数, 応答ホスト数, 宛先不達のホスト数
SSL/TLS	セッション数, 宛先ホスト数, 1 セッションの受信サイズの最大値 [bytes] と最小値 [bytes], 1 セッションの送信サイズの最大値 [bytes] と最小値 [bytes]
DNS	TCP でのクエリ数, UDP でのクエリ数, 外部の DNS サーバへのクエリ数, クエリタイプ (A, NS, PTR, MX, TXT, AAAA, SRV) 毎のクエリ数, レスポンスタイプ (CNAME, SOA) 毎のレスポンス数
HTTP	リクエストメソッド (GET, POST, HEAD, M-SEARCH) 毎のリクエスト数, 一致するリクエストメソッドの出現パターン (GET/GET, POST/GET, GET/POST/GET/GET 等 21 種類), ステータスコード (200, 302, 403 等) 毎のレスポンス数

reachable なサンプルの集合で極大のものをクラスタとする。なお、実装では Python ライブラリの scikit-learn [28] を使用する。

2.3 分類

あるマルウェア検体があった時、動的解析を行ってパケットキャプチャファイルを取得し、2.1 で示したマルウェアの通信モデルによって特徴量を抽出する。抽出した特徴量に対して、2.2 で示したクラスタリング時の前処理と同様に、通信総量が 1,000 バイト以下のサンプルは除外し、特徴量毎の標準化を行う。作成した全クラスタのサンプルに対して、分類対象のサンプルのユークリッド距離を計算し、最も近いサンプルが属するクラスタを分類先クラスタとする。その際、どのクラスタのサンプルからも閾値 Th 以上の距離が離れていた場合、既存のクラスタとは異なる通信挙動のサンプルとみなし、新種の未知検体候補とする。

3. 評価実験

3.1 データセット

提案手法の有効性を示すために、表 3 に示す 2.2 のクラスタリングを行う学習用データと、2.3 の分類を行う評価用データを用いる。図 1 に示す待ち受け型ハニーポットと Web 巡回型ハニーポットでインターネット上のマルウェアを収集した。両ハニーポットは Windows ベースの高対話型ハニーポットであり、複数の回線でインターネットに接続しており、巡回型ハニーポットのアクセス先は公開ブラックリストや別途収集しているスパムメール中の URL など多種類に渡る。また両データセット間では、マルウェアの SHA-1 ハッシュ値ベースで重複はない。マルウェアの収集と同時に動的解析を行うが、一定の制限付きでインターネットに接続されており、マルウェアからのリクエスト及びインターネット上の通信先ホストからのレスポ

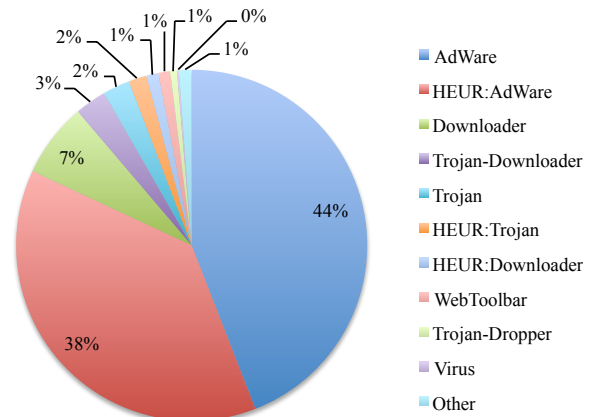


図 2 アンチウイルスソフトによる検知タイプ

スも含む。また、マルウェア実行時にダイアログ等によるユーザ操作を必要とする場合には、自動的にクリックしてマルウェアの実行を進め、最大で 5 分間実行した後、環境はリセットされる。

3.2 クラスタリング

3.2.1 精度

作成したクラスタの正確性、つまり提案手法によって作成したクラスタが、別の方法で作成した分類 (真の分類) とどの程度近い分類ができるかを評価する。そこで、学習用データのマルウェア検体を 2016 年 1 月に、あらためてアンチウイルスソフトで検知した結果を真の分類のラベルとした。収集期間から数ヶ月経過することで、アンチウイルスソフトがなるべく多くかつ正確に検知できることを期待している。図 2 に検知した上位 10 タイプを示すが、21,717 件のうち 18,209 件を検知 (641 種類) し、3,508 件は未検知であった。真の分類のラベルとしては、亜種の識別子まで含む検知名を利用する。

評価指標として、情報検索やソフトウェア類似性の評価 [29] で利用されている P (Purity), iP (Inverse Purity),

表 3 データセットの概要

	学習用データ	評価用データ
収集・解析期間	2014 年 12 月～2015 年 8 月	2015 年 9 月
サンプル数	21,717	6,078
ファイルタイプ	PE32 (97.9%), Archive (2.0%), 他 MsOffice, PE32+(x86-64), MS-DOS	PE32 (56.1%), Archive (43.6%) 他 MsOffice, PE32+(x86-64), MS-DOS
前処理後のサンプル数	17,167	3,680

P と iP の調和平均である F 値を用いる。 N をサンプル数、 L を生成されたクラスタ数、 M を真の分類のクラスタ数、 $n_{i,j}$ を生成された i 番目のクラスタにおいて j 番目の真の分類に属するサンプル数、 α は P と iP に関する重み係数とする場合、それぞれ以下の式で表される。 P の値が高いほど 1 つのクラスタに同じ検知名がまとまっており、 iP の値が高いほど同じ検知名の検体が 1 つのクラスタにまとまっていることを表す。 また F は P と iP の値がともに高いほど大きくなる。

$$P = \frac{1}{N} \sum_{i=1}^L \max(n_{i,j})$$

$$iP = \frac{1}{N} \sum_{i=1}^M \left(\frac{\sum_{j=1}^L n_{i,j} \max(n_{i,j})}{\sum_{i=1}^M n_{i,j}} \right)$$

$$F = \frac{1}{\alpha \times \frac{1}{P} + (1 - \alpha) \times \frac{1}{iP}} \quad (0 \leq \alpha \leq 1)$$

この時、 P と iP の重み α は均等に 0.5 とした。 クラスタに属するサンプルが 1 の場合も既知の通信挙動とみなして新種の未知検体の判定を行うため、 DBSCAN におけるパラメータ n は 1 とした。

$$\hat{\epsilon} = \arg \max_{\epsilon} F(\epsilon)$$

を満たす ϵ 、すなわち F を最大化する ϵ を求める。 ここで F は ϵ に依存することに注意されたい。

図 3 に、 ϵ を 0.2 から 10 まで 0.2 毎に変化させた際の、評価指標 P 、 iP 、 F の変化を示す。 図より、最大の $F=0.61$ をとる $\epsilon=4.6$ と導出することができた、ソフトウェア分類 [29] では $\max(F)=0.98$ 、テキスト分類 [30] では $\max(F)=0.79$ となっており、適用分野は異なるものの、マルウェアの通信のみで、亜種の識別子までも真の分類ラベルとした条件において、一定のクラスタリング精度を得ることができた。 この時、Max OS X (プロセッサ: 1.7 GHz Intel Core i7、メモリ: 8GB 1600MHz DDR3) の環境で、前処理を含めたクラスタリングの所要時間は 83.1 秒であった。 なお、 $\epsilon=4.6$ の時、439 クラスタを形成し、1 サンプルしか属さないクラスタも 310 クラスタ存在する。 その内 118 クラスタはアンチウイルスソフトによる検知ができなかった検体であり、全クラスタにおいて、検知できなかった検体については評価指標の算出から除外している。

3.2.2 可視化

図 4 に主要なクラスタと属するサンプルの特徴を可視化

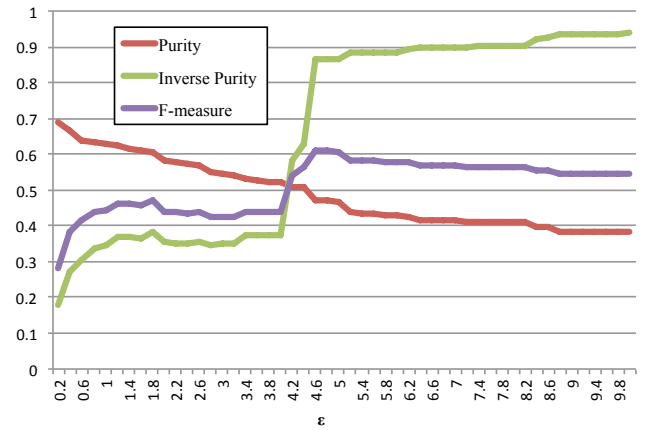
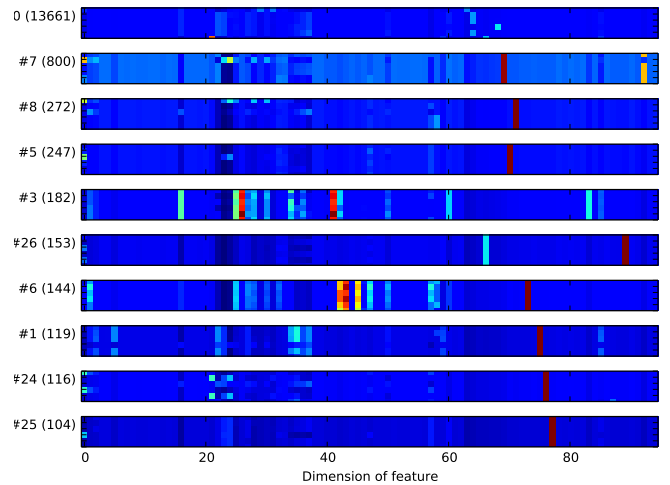
図 3 ϵ に対応する評価指標

図 4 サンプル数上位 10 のクラスタの可視化結果

した。可視化によりマルウェアの通信挙動を容易に把握することができる。 X 軸は 95 次元のマルウェアの通信モデルで定義した特徴量を表し、 Y 軸は”#クラスタ番号 (クラスタ内サンプル数)”毎に 5 個のサンプルを表示している。濃い青 (値が小さい) から濃い赤 (値が大きい) の色の範囲は各特徴量の大きさを表している。 形成されたクラスタの特徴的な通信挙動の具体例を以下に示す。

クラスタ番号 0: 最も多い 13,661 サンプルが属しており、308 種類の検知名の検体が含まれ、AdWare.Win32.MultiPlug.heur が約 43% を占めるクラスタである。 **インターネット接続確認とダウンロード** 挙動が見られ、HTTP GET リクエストが 4 回程度と POST リクエ

ストが2回程度あるが、特徴的な挙動が大きく見られない検体によりクラスタを形成していることがわかる。

クラスタ番号 7: AdWare.Win32.iBryte.jif として検知されるマルウェアが約 60%を占め、亜種を含めると約 87%となる。**通信開始時間**は 20~30 秒程度が多く、17 バイトと 2 バイトの**独自の User-Agent**で HTTP 通信を行っているが、1 件の HTTP レスポンスで 410 (Gone) を受け取っているものが多い。

クラスタ番号 3: 属する全てのサンプルは 182 個あり、全てが Trojan-Downloader.Win32.Adload.icjy として検知されるマルウェアである。**主要な Web サイトへのアクセスとダウンロード**挙動が見られ、約 4,000 もの SSL/TLS セッションがあるのが特徴的である。

3.3 分類

形成されたクラスタは複数種類の検知名のサンプルを含んでいる。これは、異なる検知名であったとしても、共通する通信挙動とみなせ、一方で同じ検知名であったとしても、異なる通信挙動であるといえることを示している。そのため分類結果の評価にあたっては、クラスタを代表する検知名を一意に特定するのではなく、複数の検知名を正しい分類結果とすることで柔軟性を持たせる。具体的には、サンプル数 K 以下のクラスタを除いたクラスタについて、クラスタ内の上位 $M\%$ の真の分類のラベルに対して、分類結果 (評価用データの真の分類のラベル) の出現率を、分類エラーとする閾値 $Th(3, 4.6, 7)$ 毎に評価した結果を表 4 に示す。表 4 から、分類エラーとする閾値 Th を 3 として、各クラスタの上位 50%を対象した場合、77.3%の出現率を得ることができた。分類エラーとする閾値 Th を大きくすると、誤ったクラスタに分類してしまうサンプルが多くなることがわかる。クラスタリングの所要時間を測定した同じ環境において、前処理も含めて最も距離の近いクラスタを算出するための所要時間は、1 サンプルあたり 0.29 秒であった。

一方、新種の未知検体の発見については、学習用データの全てのクラスタのサンプルから、閾値値 Th 以上距離が離れている評価用データ (エラー) を抽出することで、学習用データ (の真の分類のラベル) に存在していないものがどの程度発見できたかを評価した。表 3 で示した通り、学習用データの収集以降に評価用データを収集しているため、評価用データのみに出現する検知名のサンプルは新種とみなすことができる。前述の分類結果として最も出現率が高かったエラーとする閾値 Th が 3 の時の評価結果を表 5 に示す。この結果から、分類結果がエラーであり検知名のある 1,547 件 (120 種類) を対象とすると、そのうち 649 件 (90 種類) が新種の未知検体といえる。サンプル数では約 42%の確率だが、検知名の種類数では 75%の確率で、新種の未知検体を発見できた。また、分類ができた (分類 OK)

表 4 分類結果に対する真の分類のラベルの出現率
(a) $Th=3$

	$M=0.5$	$M=0.25$	$M=0.2$
$K=1$	0.773	0.769	0.559
$K=3$	0.773	0.769	0.559
$K=5$	0.773	0.769	0.559
$K=7$	0.773	0.769	0.559

(b) $Th=4.6$

	$M=0.5$	$M=0.25$	$M=0.2$
$K=1$	0.731	0.726	0.595
$K=3$	0.732	0.727	0.595
$K=5$	0.736	0.731	0.598
$K=7$	0.737	0.732	0.599

(c) $Th=7$

	$M=0.5$	$M=0.25$	$M=0.2$
$K=1$	0.579	0.568	0.505
$K=3$	0.592	0.581	0.516
$K=5$	0.595	0.584	0.519
$K=7$	0.595	0.584	0.519

表 5 新種の未知検体の発見結果

	サンプル数	検知名種類数
分類エラー	2,972	120
検知名あり	1,547	120
評価用データのみあり	649	90
学習用データにもあり	899	30
分類 OK	708	64
検知名あり	472	64
評価用データのみあり	106	44
学習用データにもあり	366	20

サンプルのうち、評価用データのみにあったサンプルは 106 件 (44 種類) ある。新種の未知検体の見逃しの可能性があるが、全てのサンプルが 3.2.2 で示したクラスタ番号 0 に分類されており、特徴的な通信挙動が大きく見られないマルウェアであった。

4. 関連研究

我々の研究は、多数の従来研究と解析経験に基づいて、多数の特徴量を基にしたマルウェアの通信モデルを定義し、クラスタリングの精度評価と、形成されたクラスタを利用して、既知のクラスタへの分類と同時に、新種の未知検体を発見する手法を提案している点で、以下に例として挙げる従来研究とは異なる。

BOTFINDER [7] は、通信フローのインターバル時間の平均時間やフロー毎の送受信バイト数等を用いて、6 種類のボットファミリーに対して教師なし学習を適用し検知する手法を提案している。また、文献 [5] では、マルウェアによる持続的な状態の変化に関する挙動を定義し、依存関係を表すグラフとして表現し、階層的クラスタリングに

よる有意なグループの抽出にとどまる。これらの先行研究は、教師なし学習を適用している点は本研究に類似するが、新種の未知検体の発見といった有用性の高い結果を得ていない。

HTTP や SMTP のようなプロトコルがわかっている通信と、未知のプロトコルの通信挙動を扱うモデルをもとに、20 ファミリーで 6,000 のマルウェアで複数の分類手法を評価している研究 [17] もある。文献 [6] では、トラフィックデータから抽出したフローの依存関係を挙動グラフとして表現し、その構造から 10 種類の特徴量を定義し、決定木を用いてマルウェアの分類を行う。アンチウイルスソフトで検知した 13 ファミリーの数千のマルウェアで評価を行っている。FIRMA [15] はマルウェアの分類とネットワーク・シグネチャの生成を提案している。送信元・宛先の IP アドレスとポート番号、トラフィックサイズ、HTTP や IRC 等のリクエストに含まれるホスト名といった特徴量を基に分類した結果で、類似するマルウェアのネットワーク・シグネチャを自動的に生成することを提案している。これらの先行研究は、多数のファミリー及びサンプルをもとに評価しているが、既知のマルウェアに関する分類にとどまる。

5. まとめ

マルウェアを実行するホストの外部で取得できる通信データのみを用いて、マルウェア特有の特徴量 25 種類と、基本的な通信の特性を表す特徴量 70 種類によって、マルウェアの通信モデルを提案した。このように多数の特徴量を利用することで、個々の特徴を隠蔽するような攻撃に対して耐性が高いと考えられる。また、21,717 個のマルウェアに対して、マルウェアの通信モデルを適用し、クラスタリングを行って、その精度を評価した。マルウェアの通信のみを用いて、亜種の識別子までも含むアンチウイルスソフトの検知名を判定条件とし、 $F=0.61$ を得られるクラスタリングのパラメータ $\epsilon=4.6$ を導出するとともに、クラスタリング結果の可視化と、クラスタが示す通信挙動の具体例を挙げた。クラスタリングの所要時間は 83.1 秒と軽量であることも示した。

形成されたクラスタを利用して、評価用データを 1 件あたり 0.29 秒で、最大 77.3% で適切に分類できることを示し、サンプル数では約 42%、検知名の種類数では 75% の確率で、分類エラーとなったサンプルに着目した新種の未知検体を発見できることを示した。アンチウイルスソフトで検知できないマルウェアも多く、妥当なラベル付けを定常的に更新していくこと自体が困難であるという現実的な課題に対して、本研究で得られた結果は非常に有益といえる。

今後の課題として、今回の実験では真の分類のラベルをアンチウイルスソフトの検知名としているため、検知できなかったサンプルについては分類の適切さや新種の未知検体の発見の評価ができていない。静的解析結果やトラ

フィックデータ以外の動的解析結果をもとにしたラベル付けによる評価をすることができる。また、分類結果の評価にあたっては、クラスタを代表する検知名を一意に特定するのではなく、複数の検知名を正しい分類結果とすることで柔軟性を持たせたが、より適切な評価方法を検討する必要がある。

謝辞 本研究の一部は JSPS 科研費 JP16H02832 の助成を受けたものです。

参考文献

- [1] “Malware Statistics & Trends Report — AVTest Institute.” <https://www.av-test.org/en/statistics/malware/>.
- [2] “Threat intelligence report - Unwanted software.” https://www.microsoft.com/security/portal/enterprise/threatreports_october_2015.aspx.
- [3] “Cuckoo sandbox.” <http://www.cuckoosandbox.org/>.
- [4] “Anubis.” <https://anubis.iseclab.org/>.
- [5] M. Bailey, J. Oberheide, J. Andersen, Z. M. Mao, F. Jahanian, and J. Nazario, “Automated classification and analysis of internet malware,” in *Proceedings of the 10th International Conference on Recent Advances in Intrusion Detection*, pp. 178–197, Sep. 2007.
- [6] S. Nari and A. A. Ghorbani, “Automated Malware Classification based on Network Behavior,” in *Proceedings of International Conference on Computing, Networking and Communications 2013*, pp. 642–647, Jan. 2013.
- [7] F. Tegeler, X. Fu, G. Vigna, and C. Kruegel, “Botfinder: Finding bots in network traffic without deep packet inspection,” in *Proceedings of the 8th International Conference on Emerging Networking Experiments and Technologies*, pp. 349–360, Dec. 2012.
- [8] Luca Invernizzi and Stanislav Miskovic and Ruben Torres and Sabyaschi Saha and Sung-Ju Lee and Christopher Kruegel and Giovanni Vigna, “Nazca: Detecting malware distribution in large-scale networks,” in *Proceedings of the ISOC Network and Distributed System Security Symposium (NDSS '14)*, pp. 1–16, Feb. 2014.
- [9] H. Zhang, D. D. Yao, and N. Ramakrishnan, “Detection of stealthy malware activities with traffic causality and scalable triggering relation discovery,” in *Proceedings of the 9th ACM Symposium on Information, Computer and Communications Security*, pp. 39–50, Jun. 2014.
- [10] H. Mekky, A. Mohaisen, and Z.-L. Zhang, “POSTER: Blind Separation of Benign and Malicious Events to Enable Accurate Malware Family Classification,” in *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, pp. 1478–1480, Nov. 2014.
- [11] J. M. Beaver, C. T. Symons, and R. E. Gillen, “A learning system for discriminating variants of malicious network traffic,” in *Proceedings of the Eighth Annual Cyber Security and Information Intelligence Research Workshop*, pp. 23:1–23:4, Jan. 2013.
- [12] V. Eliseev and Y. Shabalin, “Dynamic response recognition by neural network to detect network host anomaly activity,” in *Proceedings of the 8th International Conference on Security of Information and Networks*, pp. 246–249, Sep. 2015.
- [13] B. Cappers and J. van Wijk, “Snaps: Semantic network traffic analysis through projection and selection,” in *Pro-*

- ceedings of 2015 IEEE Symposium on Visualization for Cyber Security (VizSec), pp. 1–8, Oct. 2015.
- [14] R. Perdisci, W. Lee, and N. Feamster, “Behavioral clustering of http-based malware and signature generation using malicious network traces,” in *Proceedings of the 7th USENIX Conference on Networked Systems Design and Implementation*, pp. 1–14, Apr. 2010.
 - [15] M. Z. Rafique and J. Caballero, “FIRMA: Malware Clustering and Network Signature Generation with Mixed Network Behaviors,” in *Proceedings of the 16th International Symposium on Research in Attacks, Intrusions and Defenses*, pp. 144–163, Oct. 2013.
 - [16] “Tshark.” <https://www.wireshark.org/docs/man-pages/tshark.html>.
 - [17] M. Z. Rafique, P. Chen, C. Huygens, and W. Joosen, “Evolutionary algorithms for classification of malware families through different network behaviors,” in *Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation*, pp. 1167–1174, Jul. 2014.
 - [18] G. Gu, J. Zhang, and W. Lee, “Botsniffer: Detecting botnet command and control channels in network traffic,” pp. 1–18, Feb. 2008.
 - [19] G. Gu, R. Perdisci, J. Zhang, and W. Lee, “Botminer: Clustering analysis of network traffic for protocol- and structure-independent botnet detection,” in *Proceedings of the 17th Conference on Security Symposium*, pp. 139–154, Jul. 2008.
 - [20] S. Shin and G. Gu, “Conficker and beyond: A large-scale empirical study,” in *Proceedings of the 26th Annual Computer Security Applications Conference*, pp. 151–160, Dec. 2010.
 - [21] J. Morales, A. Al-Bataineh, S. Xu, and R. Sandhu, “Analyzing and exploiting network behaviors of malware,” in *Security and Privacy in Communication Networks*, vol. 50 of *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pp. 20–34, 2010.
 - [22] “Alexa Top Sites.” <http://www.alexa.com/topsites>.
 - [23] “WhatIsMyIP.com.” <https://www.whatismyip.com/>.
 - [24] “abuse.ch.” <https://www.abuse.ch/>.
 - [25] “Domain feed of known DGA domains.” <http://osint.bambenekconsulting.com/feeds/dga-feed.txt>.
 - [26] “Adblock plus.” <https://easylist-downloads.adblockplus.org/easylist.txt>.
 - [27] M. Ester, H. Peter Kriegel, J. S., and X. Xu, “A density-based algorithm for discovering clusters in large spatial databases with noise,” in *Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining*, pp. 226–231, Aug. 1996.
 - [28] “scikit-learn.” <http://scikit-learn.org/>.
 - [29] 岸本康成, 坂本啓, 小林透, “ソースコードと設計書を用いたソフトウェアの派生関係の抽出,” *情報処理学会論文誌*, vol. 53, pp. 1137–1149, Mar. 2012.
 - [30] E. Amigó, J. Gonzalo, J. Artiles, and F. Verdejo, “A comparison of extrinsic clustering evaluation metrics based on formal constraints,” *Information Retrieval*, vol. 12, no. 4, pp. 461–486, 2009.