マルウェアによる感染活動の目的推定に向けた動的解析ログに基づく分析

森 優輝[1,a)]　金居 良治[2]　吉浦 裕[1]　市野 将嗣[1]

概要：近年，インターネットを介した攻撃が増加し，被害が深刻なものとなってきている．インターネットを介した攻撃は，一般的にマルウェアが用いられる．攻撃の数が増えると同時に，マルウェアの種類や数も日々増加して行く中，従来の様にマルウェア一つ一つを解析し，それぞれのマルウェア用に対策を講ずるのでは，対策が追いつかない．そこで攻撃の目的が事前にわかれば，その目的の部分を重点的に防御できるような対策を講ずることができると考えられる．本稿では，Soliton Dataset 2018 に含まれる動的解析ログを用いてマルウェアの挙動推定に関する検討を行うために，マルウェアの特徴的な挙動の分析を行ったのでその結果を報告する．

キーワード：マルウェア，目的分類，動的解析ログ

Analyze of Dynamic Analysis Log for Estimating Object of Malware Infection Activity

Yuuki Mori[1,a)]　Ryoji Kanai[2]　Hiroshi Yoshiura[1]　Masatsugu Ichino[1]

Abstract: In recent years, the number of attacks via the Internet has increased, and the damage has become serious. Attacks via the Internet generally use malware. As the number of attacks increases and at the same time the types and numbers of malware are increasing day by day, measures can not keep up with analyzing malware one by one as in the past and taking measures for each malware. Therefore, if the purpose of the attack is known in advance, it can be considered that measures can be taken that can intensively defend the objective part. In this study, we analyzed the commonality and differences between specimens having the same attack purpose using Soliton Dataset 2018 which is dynamic analysis log, and report the results.

Keywords: Malware, Object Classification, Dynamic Analysis Log

1. はじめに

マルウェアを用いた攻撃活動が増加し，被害が深刻なものとなっている [1]．現在の攻撃対策として，ウイルス対策ソフトやファイヤーウォール，IDS/IPS の導入などが挙げられる． 従来のシグネチャベースのマルウェア検知手法だけではなく，感染後の挙動からマルウェア感染を検知する手法も利用されている．また，近年では，複数のエンドポイント端末でのプロセス挙動を集約し，不審なプロセスの追跡などの対応が可能な EDR(Endpoint Detection and Response) 製品にも注目が集まっている [2]．標的型攻撃をはじめとするサイバー攻撃による被害事例が増加している．例えば，標的型攻撃が行われる際には，攻撃対象に合わせたマルウェアを作成するなど，攻撃者による入念な準備が行われていることが多い [3]．また，マルウェアに感染してから被害の発覚までにかかる時間が長期間に渡る場合があり，自組織で観測できる挙動情報から攻撃の推定を行う手法が必要であると考えられる [3]．標的型攻撃におけ

1 電気通信大学
The University of Electro-Communications
2 株式会社 FFRI
FFRI Inc.
a) y-mori@uec.ac.jp

© 2018 Information Processing Society of Japan

る，攻撃準備から目的実行までの一連の攻撃活動を階層化した Cyber Kill Chain 呼ばれるモデルが存在する [4]．具体的には偵察，武器化，配送，攻撃，インストール，コマンド＆コントロール，目的実行の 7 段階で構成されている．

従来はマルウェア検知が攻撃対策の中心であるが，攻撃を検知しただけではその時点での攻撃段階が把握できず，どこから対策を行えばいいのかすぐに判断できない．そこで，攻撃を検知した後にその攻撃目的を推定できれば，重点的に対策すべき箇所と対策方法を短時間で把握することができると考えられる．攻撃者にとってマルウェアは目的を遂行するためのツールの 1 つであり，攻撃段階に応じた挙動を行うことが知られている [5]．マルウェアの検知と同時にその挙動を推定することができれば，攻撃段階の推定や，一連の攻撃活動を見ていく中で攻撃目的の推定もできると考える．

そこで本研究では，動的解析のログデータの分析を行い，マルウェアの特徴的な挙動について調査した．また，機械学習を用いて挙動の推定を行った．

以下，2 章では関連研究を紹介する．3 章でログデータの分析について述べ，4 章では推測を行った結果を，5 章ではその考察を行う．最後にまとめと今後の課題について述べる．

2. 関連研究

本章では，攻撃の推測を行っている既存研究と，動的解析ログを用いたマルウェアの解析や分類を行っている既存研究について述べる．

2.1 攻撃活動の推測・分析

攻撃活動の検知や推定などの分析を行っている研究が行われている．西野らは，攻撃者による組織ネットワークへの攻撃活動に対して，攻撃者の通信挙動やプロセス挙動特異性に着目し一連のグラフを作成することで，大きな被害をもたらす攻撃を検知する手法を提案した [6]．川北らは，ソーシャルメディア上の情報を集めて分析することで，攻撃の兆しを早期に検知できることを示した [7]．芦野らは，自組織のネットワークで観測したノイズの挙動を分析することで，攻撃者の意図や攻撃システムを推測できることを示した [8]．

2.2 動的解析ログを用いたマルウェアの分析

動的解析ログを用いて分析を行っている研究として，藤野らは，引数も含めた API コールログに対してクラスタ分析を実施し，類似する挙動をするマルウェアの分類を行った [10]．ログデータから API 名とその引数を組み合わせて単語を抽出しトピック分析を行うことで，マルウェアの分類毎に特徴的な API コールを示している．青木らは，API コールから API 名を抽出し n-gram を作成することで，マルウェアに特徴的な API コールパターンの分析を行った [11]．川古谷らは，データの送受信に関わる API と，それによって呼び出される API との依存関係を用いて，送受信されたデータの内容の特定を行った [12]．

2.3 本研究の位置づけ

2.1 節で述べた攻撃活動の推測や分析を行う研究では，観測した攻撃情報から攻撃の意図や影響範囲を推定できることが示されている．また，2.2 節で述べた動的解析ログを用いたマルウェアの分析を行う研究では，ログデータの情報を適切に抽出することでマルウェアの検知やマルウェアファミリーの分類が出来ることが示されている．

しかし，文献 [6][7] および 2.2 節で紹介した研究はマルウェア検知の観点での分析が主であり，マルウェアの目的の観点での分析ではない．文献 [8] では攻撃者の意図やシステムの推定に通信挙動の観点のみで検討しており，ホスト内部の挙動情報は使用されていない．

本研究の目的はマルウェア検知やファミリー分類ではなく，攻撃目的の推定である．本稿では，攻撃目的の推定に向けて，マルウェアの特徴的な挙動についてマルウェアの目的という観点から分析を行い，動的解析ログと挙動の関係性を明らかにする．

3. データ分析

ここでは，マルウェアの感染活動の目的の推定を行うための検討として，ログデータの分析方法と検討結果について述べる．

3.1 分析概要

本研究では，マルウェアのログデータから抽出した特徴量を用いて，攻撃活動の目的の推定を行うことが目標である（**図 1**）．同じ目的のマルウェア間でどのような共通性があるか，あるいは異なる目的のマルウェア間でどこに差異があるかが分かれば，攻撃活動の目的推定が可能になると考える（**図 2**）．そのために，攻撃に使用されるマルウェアの役割の推定を行うことができれば，攻撃目的の推定に繋げられると考えられる．本稿では，Soliton Dataset 2018 に含まれる動的解析ログを用いて，攻撃活動におけるマルウェアの役割の推定に関する検討を行うために，マルウェアの特徴的な挙動の分析を行った．

3.2 使用データ

分析データとして，MWS Datasets 2018 の一部として提供されている Soliton Dataset 2018 を使用した [13]．Soliton Dataset とは，エンドポイントセキュリティ製品が導入された物理環境にて，2017 年 1 月から 2018 年 2 月までに取得したマルウェアを動作させて取得した動的解析ログである．収集時間は原則として 30 分である．一部の検体

© 2018 Information Processing Society of Japan

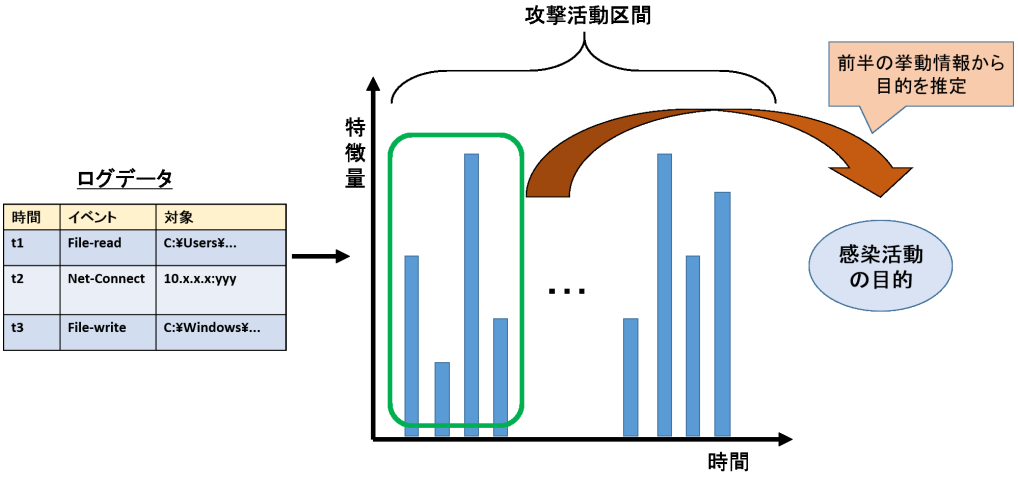

図 1　感染活動の目的の推定
Fig. 1　Estimating objects of infection activity

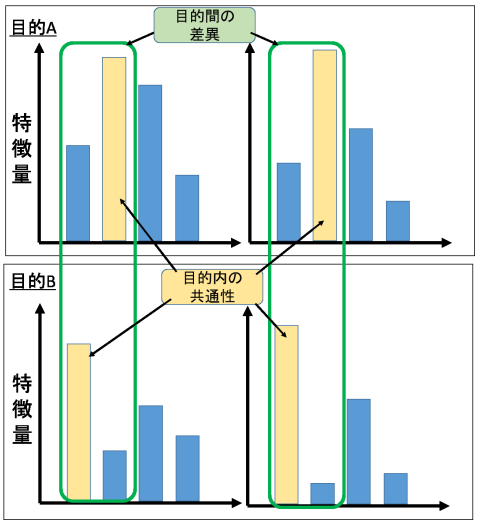

図 2　目的内および目的間の分析
Fig. 2　Analysis

では動作時間が 20 分のものや 2 時間のものが存在する．

本研究では，Soliton Dataset に含まれる 117 検体のうち，後述するラベル付けができた 81 検体分のログを分析に使用した．

3.3　分析方法

ログデータの分析を行うために，各検体の挙動に対してラベル付けを行い，決定木の作成と可視化を行った．これらの具体的な手順を以下に述べる．

3.3.1　挙動のラベル付け

マルウェアがどういう挙動をするものか紐付けるために，各検体に対してラベル付けを行った．ラベルとは，攻撃者による一連の攻撃の中でマルウェアが行う役割のことを指す．ラベル付けを行うにあたり，セキュリティベンダー各社が公開しているマルウェア分析レポートや文献 [14][15] を参考にした．ラベル付けを行った結果を**表 1** に示す．

表 1　ラベル付けの結果
Table 1　Result of behavior labeling

役割ラベル	検体数
データの暗号化・破壊	30
情報窃取・収集	29
ダウンローダ	15
拡散・感染活動	7

```
03/27/2018 19:38:45.110 +0900
evt=file subEvt=close
psGUID={7A166B96-2A44-4770-9D57-FF66A1750E32}
psPath="C:\Users\**\AppData\Local\Temp\fxvtd.exe"
path="C:\Users\**\AppData\Roaming\Microsoft\Windows\Cookies\index.dat"
read=32768 write=0 size=32768
```

図 3　動的解析ログの例
Fig. 3　Example of dynamic analysis log

3.3.2　単語の抽出

まずはログデータから一定の規則に基づいて文字列を抽出する．本稿ではこの文字列のことを単語と呼ぶ．

はじめにログデータに含まれるイベント情報を単語に変換する．ここで，ログデータの抜粋を**図 3** に示す．図中の evt, subEvt はそれぞれプロセスのイベント名，サブイベントである．psPath は操作を実行したプロセス名，path はイベントの対象となるファイル名である，1 つのイベントのログにはこれら以外にも様々な項目が存在する．これらのログから単語を作成した．単語は**表 2** のようにイベントによって作成方法を変えている．このとき，単語数の増加を抑えるために，ファイルパスやレジストリパス対してカテゴライズを，IP アドレスに対してマスク処理を実施している．具体的なカテゴリは**表 3** と**表 4** に示す．

3.3.3　特徴ベクトルの作成

各検体のログから単語を抽出したのち，n-gram を作成し出現頻度を単語ごとに算出した．検体毎に抽出できる単語の数は異なることから，全ての検体に出現する単語を網

© 2018 Information Processing Society of Japan

表 2 単語生成ルール
Table 2 Word creation rules

イベント名	サブイベント名	単語
ps	start	ps:start:<category> ps:start:<path>(Windows フォルダ以下かつデジタル署名されているファイルのみ)
	その他	ps:<subEvt>:<category>
file	close	file:read:<category>(read=!0 かつ write=0 の場合) file:write:<category>(read=0 かつ write!=0 の場合) file:close:<category>(read!=0 かつ write!=0 の場合)
	copy rename	file:copy:<src category>:<dst category> file:rename:<src category>:<dst category>
	その他	file:<subEvt>:<category>
reg	すべて	reg:<subEvt>:<category>
net	con, dcon	net:<subEvt>:<category>:<port num>
	その他	net:<subEvt>
その他	すべて	<evt>:<subEvt>

表 3 ファイルパスのカテゴリ
Table 3 Filepath category

カテゴリ名	対象パス
system	C:\Windows
pgfiles	C:\Program Files
userprofile	C:\Users\<username>
appdata	C:\Users\<username>\AppData\Roaming
temp	C:\Users\<username>\AppData\Local\Temp
public	C:\Users\Public
pgdata	C:\ProgramData
startup	C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Startup
startup-usr	C:\Users\<username>\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup
other	その他

表 4 レジストリのカテゴリ
Table 4 Registry category

カテゴリ名	対象パス
startup	HKLM\Software\Microsoft\Windows\CurrentVersion\Run など自動起動系 17 ヶ所
services	HKLM\SYSTEM\CurrentControlSet\Services\
inet_setting	HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings
other	その他

羅しているコードブックを作成し，各検体毎に全単語の出現頻度を算出した．本研究では，この単語数分の要素を持つベクトルを特徴ベクトルとした．

3.3.4 決定木の作成と木構造の可視化

各ラベル毎に特徴的な挙動を調べるために，決定木を作成し，その木構造の可視化を行った．決定木は，分類過程を木構造の形で可視化出来るという特徴がある．その木構造を分析することで，ラベル分類に使用される特徴的な挙動を調べた．

3.3.1 節にて作成したラベル情報を学習ラベルとし，ログデータから作成した特徴ベクトルを学習データとした．最後に，81 検体全てを使って決定木学習を行った．本稿では，作成した決定木を用いて学習データを識別させ，識別率が最も高くなった深さの木を分析に用いる．

表 5 抽出した単語数
Table 5 number of extracted words

	カテゴライズ前	カテゴライズ後
1-gram	318044	208
2-gram	365655	2575

表 6 特徴量の重要度（上位 5 件）
Table 6 Feature importance (Top 5)

特徴量名	重要度
file:create:other	0.207
reg:create:inet_setting	0.147
file:read:system	0.105
file:create:system	0.104
reg:create:other	0.0869

3.4 分析結果・考察

抽出した単語数や，可視化した決定木よりラベル内の共通性やラベル間の差異について分析を行った結果を述べる．

1-gram と 2-gram を用いて単語の抽出を行った．ファイルパスやレジストリパスのカテゴライズ前後で抽出した単語数を**表 5** に示す．

決定木は不純度指標にエントロピーを用いて深さ 9 のときに識別率 100% となった．このときの特徴量の重要度を**表 6** に示す．次に，1-gram の単語を用いて作成・可視化を行った決定木の一部を**図 4**，**図 5**，**図 6**，**図 7** に示す．

図中の “value” は，そのノードに含まれる各ラベルの検体数を表しており，左からダウンローダ (downloader)，暗号化・破壊 (encrypt)，拡大・感染活動 (spread)，情報窃取 (infotheft) ラベルを表している．各ノードから伸びる矢印は，各特徴量の閾値に対して，左下方向が True, 右下方向が False であることを示している．

図 4 は暗号化ラベルが付けられたノード #40 付近の決定木である．ノード #40 に含まれる検体はすべてランサ

© 2018 Information Processing Society of Japan

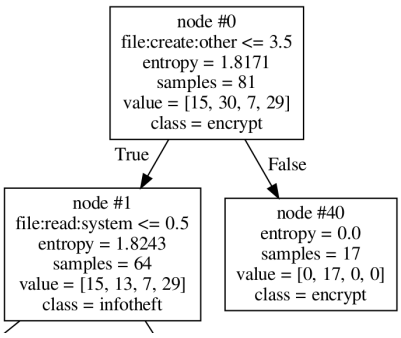

図 4 ノード #40 暗号化ラベル付近の決定木
Fig. 4 Decision tree near encrypt class

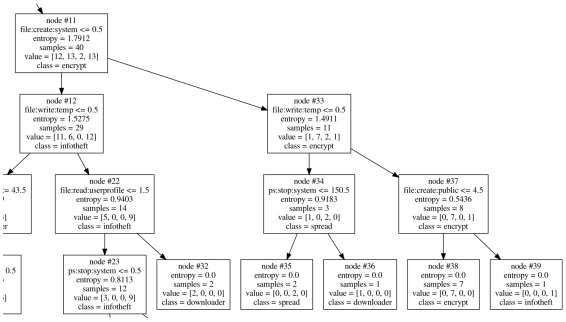

図 5 ノード #38 暗号化ラベル付近の決定木
Fig. 5 Decision tree near encrypt class

ムウェアであり，ファイル操作系の挙動が他のラベルと比較して高い傾向にある．ノード #40 の親ノードでは，‘file:create:other’ という単語が 3.5 回以上出現した場合に暗号化ラベルに分類されることがわかる．また，決定木のノード # 38 付近 (図 5) を見ると，“file:create:system”や“file:write:temp”が出現したときに暗号化ラベルのノードが作成されていることから，ファイル作成という挙動の多さが一つの特徴になっていることがわかる．

図 6 は情報窃取ラベルが付けられたノード #31 付近の決定木である．このノードに含まれる 7 検体の分類はトロイの木馬 2 検体，バックドア 2 検体，スパイウェア 2 検体，Generic 系が 1 検体であった．決定木の根ノードを辿ると，単語の出現頻度が 3 回以下の “file:create:other”，1 回以上の “file:read:system”，0 回の “file:create:system”，1 回以上の “file:write:temp”，1 回以下の “file:read:userprofile”，1 回以上の “ps:stop:system” と続いている．ノード #31 に分類された 7 検体のうち 5 検体で C:\Windows \System32 ディレクトリ内の DLL ファイルを読み込んでおり，“file:read:system” の出現頻度が大きくなっている傾向がある．例えば “`netsh firewall add allowedprogram <sample file>`” というコマンドが実行されたときに，そのプログラムを動作させるためのライブラリが多く読み込まれていた．読み込まれるシステムディレクトリ内の DLL ファイルを確認することで，その検体が行う挙動についての検討ができる可能性がある．また，Temp ディレクトリへの書き込みについて見ると，7 検体全てにおいて bat ファイルまたは exe ファイルのドロップと起動を行っている共通の挙動を確認できた．

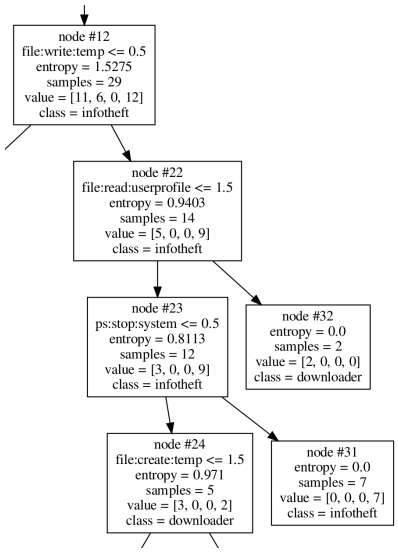

図 6 ノード #31 情報窃取ラベル付近の決定木
Fig. 6 Decision tree near encrypt class

図 7 はダウンローダラベルが付けられたノード #17 付近の決定木である．このノードに含まれる 5 検体の分類はトロイの木馬 3 検体，不正な HTML や FakeAlert 系が 2 検体であった．決定木の根ノードを辿ると，単語の出現頻度が 3 回以下の “file:create:other”，1 回以上の “file:read:system”，0 回の “file:create:system”，0 回の “file:write:temp”，43 回以下の “reg:create:inet_setting”，1 回以上の “file:read:other”，0 回の “file:read:appdata” と続いている．このノードでは，検体の実行はできたものの，通信に失敗するなどの何らかの理由で早期にプロセスが終了したと思われる検体が多く含まれていた．外部サーバへ通信を試行している検体は 10 分程度，それ以外の検体は数秒でプロセスが終了しており，全体的に単語の出現頻度が小さくなったために同じノードに分類されたと考えられる．他のノードでは，外部サーバと通信を行っている途中で収集時間の 30 分に達してしまい途中で終了したと思われる検体が集まっていた．そのノードでは “reg:create:inet_setting” が 200 回以上出現していた．この単語はインターネット通信を行う際に，Windows がプロキシ情報などのネットワーク設定を確認するために当該レジストリキーにアクセスしていることを表している．

拡散・感染活動ラベルでは，検体が 4 検体含まれているノードの根ノードを辿ると，単語の出現頻度が 3 回以下の “file:create:other”，0 回の “file:read:system”，202 回以下の “reg:create:inet_setting”，1 回以上の “file:chgattr:pgdata”（ファイルが隠しファイル属性に変更されたことを表す）と続いている．“file:chgattr:pgdata” は C:\Program Data ディレクトリ内のファイルが隠しファイル属性に変更さ

© 2018 Information Processing Society of Japan

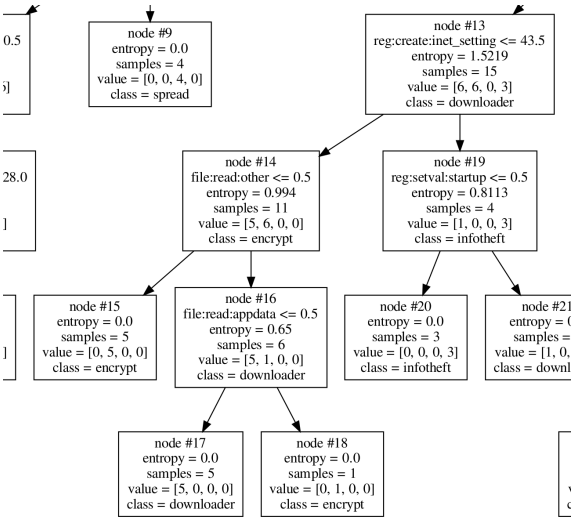

図 7　ノード #17 ダウンローダラベル付近の決定木
Fig. 7　Decision tree near encrypt class

れたことを表している．属性変更がなされたファイルは検体がドロップした exe ファイルであった．しかし，ダウンローダのときと同様に．4 検体とも数秒でプロセスが停止しており，全体的に単語の出現頻度が小さくなったために同じノードに分類されたと考えられる．また，外部サーバと通信を行っている途中で収集時間の 30 分に達してしまい途中で終了したと思われる検体が所属するノードも作られていた．

3.4.1　ヒストグラムの作成

ラベル毎の単語の出現頻度の差を確認するために，ヒストグラムを作成した．決定木を作成すると，同じラベルの検体が複数の葉ノードに分類されていることから，各葉ノードより 1 検体ずつピックアップしてラベル毎にまとめた．作成したヒストグラムを**図 8**，**図 9**，**図 10**，**図 11** に示す．ヒストグラムの横軸は各単語に割り振られた単語 ID を，縦軸は単語の出現頻度を表している．また，ヒストグラム中の線は “file” や “net” などのイベント名ごとに色を分けている．

また，ログデータを確認していると，検体の実行が途中で停止したと思われるものだけではなく，活動途中で収集の時間切れとなり途中で終了してしまったと思われるものがあった．これらの検体について，この後に行う推定実験で正しいラベルだと識別することができれば，感染活動中の挙動からマルウェアの目的が推定できると考えられる．

4.　挙動推定に関する実験

ログデータからマルウェアの挙動が推測できるか確認するために評価実験を行った．

4.1　実験方法

決定木および SVM を用いてマルウェア挙動の分類を

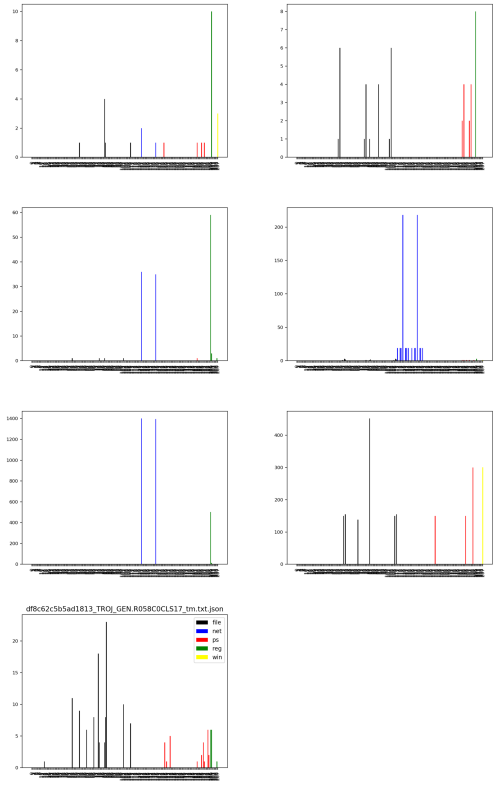

図 8　ダウンローダラベルのヒストグラム
Fig. 8　Histogram of downloader class

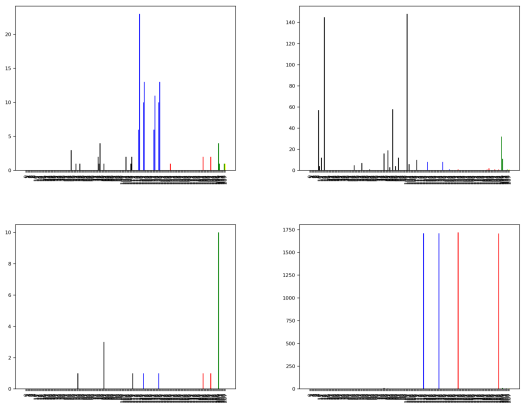

図 9　暗号化ラベルのヒストグラム
Fig. 9　Histogram of encryption class

行った．本研究では使用データ数として 81 検体を使用し，単語の抽出には 1-gram と 2-gram を使用した．評価の際には交差検定を実施した．ここでの交差検定は，各ラベル（クラス）から 1 検体ずつをテストデータとして選択し，それ以外の検体を学習データとした．すなわち，本実験では 4 クラス分類を行うため，4 検体をテストデータとして使用した．交差検定の組み合わせ数は表 1 に記載したラベルに含まれるそれぞれの検体数の積で表すことができ，$30 \times 29 \times 15 \times 7 = 91350$ 通りである．また，決定木および SVM の実装には，scikit-learn[16] 0.18.2 を使用した．

© 2018 Information Processing Society of Japan

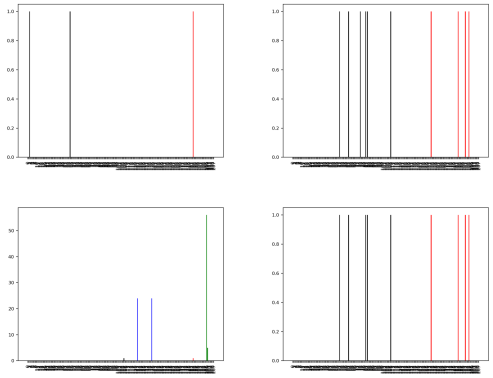

図 10　拡散・感染活動ラベルのヒストグラム
Fig. 10　Histogram of spread class

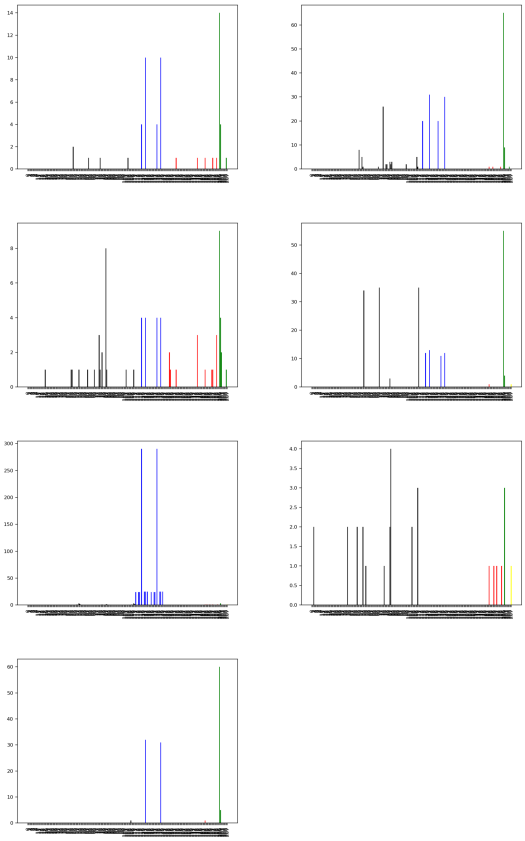

図 11　情報窃取ラベルのヒストグラム
Fig. 11　Histogram of information theft class

4.2　実験結果

決定木と SVM での識別率を**表 7** に示す．また，決定木での混同行列を**表 8**，SVM での混同行列を**表 9** に示す．表の数値はテストデータの数を表す．混同行列は，行方向が正解，列方向が予測結果である．例として表 8 の見方を説明する．1 行目では，ダウンローダのラベルを付けられた検体が一体どのラベルに識別されたのかを表しており，ダウンローダと識別されたのは 40883 回，暗号化と識別されたのは 12861 回，拡散と識別されたのは 5256 回，情報窃取と識別されたのは 32350 回であることを示している．

表 7　識別結果
Table 7　Classification rate

識別アルゴリズム	識別率 (1-gram)	識別率 (2-gram)
決定木	70.5%	53.2%
SVM	55.1%	56.6%

表 8　識別結果の混同行列 (決定木)
Table 8　Confusion matrix

		予測			
		ダウンローダ	暗号化	拡散	情報窃取
正解	ダウンローダ	40883	12861	5256	32350
	暗号化	7147	79600	1926	2677
	拡散	3434	2330	72435	13151
	情報窃取	15730	8318	2736	64566

表 9　識別結果の混同行列 (SVM)
Table 9　Confusion matrix

		予測			
		ダウンローダ	暗号化	拡散	情報窃取
正解	ダウンローダ	17415	20828	10830	42277
	暗号化	23911	47375	6351	13713
	拡散	0	0	78300	13050
	情報窃取	23031	9954	210	58155

5.　考察

3 章で行ったデータ分析の結果および 4 章で行った挙動推定実験の結果を基に考察を行う．

5.1　識別結果

本手法における決定木での識別率は 70.5% であったが，表 8 に示した決定木の混同行列を見ると，ラベル毎に分類できた割合が異なっている．最も識別率が高いのは暗号化ラベルの 87.1% であり，最も識別率が低いのはダウンローダラベルの 44.8% であった．また，情報窃取ラベルの識別率は 70.1%，拡散・感染活動ラベルの識別率は 79.3% であった．

しかし，図 4 は決定木の根ノードから 1 つ派生した部分に存在するため，ファイルアクセス系単語の出現頻度の大小だけで分類が行われてしまう．そのため，他の単語が分類に使用されず，他のラベルから暗号化ラベルへの誤識別が発生しやすいと考えられる．

ダウンローダラベルでは，図 8 より，主に通信系の単語の出現頻度が高いものと，ファイル系やプロセス系の出現頻度が高いものに分かれていることがわかる．ダウンローダ以外のラベルに含まれる検体で通信を行うものは多く，他ラベルと差別化できる特徴が現れにくかったと考えられる．

© 2018 Information Processing Society of Japan

表 10 ラベル付けの内訳
Table 10 breakdown of sample labeling

検体名／ラベル名	暗号化	情報窃取	ダウンローダ	拡散
Trojan	4	9	10	0
Worm	1	0	0	4
Ransom	24	0	0	0
Backdoor	0	9	0	0
Spyware	0	4	0	0
Other	1	7	5	3

拡散・感染活動ラベルでは，図 10 のヒストグラムを見ると他のラベルよりも単語の出現頻度が低く，それが特徴となってしまっている可能性がある．

誤識別が多かった検体の特徴として，感染初期のインストール活動後に動作が停止してしまったものがある．ファイルの配置と実行および通信試行の挙動は全てのラベルで共通して含まれており，感染初期の挙動だけではその後の挙動推定が困難であり，識別率低下の要因となったと考えられる．

単語の抽出に 2-gram を用いた場合，決定木では 1-gram と比較して識別率が 17.3% 低下した．単語の数は増加したが，その分出現頻度が 0 となる単語が多くなり，分類に使用できる特徴が現れにくくなったと考えられる．

5.2 各ラベル付けの共通性と差異

ベンダー各社は必ずしも挙動に基づいて検体を分類しているとは限らず，異なる分類間で類似した挙動を行う検体がある [17]．そのため本研究では，各検体に対して挙動という観点で，ベンダーのマルウェア解析レポートを参考にラベル付けを行った．作成ラベルとそれらに含まれる分類名の内訳を**表 10** に示す．

しかし，今回のラベル付けは一定のアルゴリズムに沿ったものではなく筆者らの主観に基づいて行われているため，実際の挙動と付与したラベルが異なっている可能性が考えられる．ラベル付け手法の検討や自動化については今後の課題とする．

6. まとめと今後の課題

本研究では，攻撃者の攻撃目的推定に向けたマルウェアの挙動推定を行うべく，動的解析のログデータからマルウェアの特徴的な挙動について分析を行い，推定実験を行った．その結果，マルウェアの役割毎に特徴的な挙動を明らかにし，70.5% の精度で挙動推定を行うことができた．

今後の課題として，検体数や 1 検体あたりのデータ量を増やして検討必要がある．また，単語の出現頻度を用いた検討だけではなく，時系列を考慮した挙動の推定も課題である．今回はログデータ全体を使用して挙動推定を行ったが，ログをある時間毎に区切った上で前半のログデータから後半の挙動を推定できれば，リアルタイムに推定ができると考えられる．

謝辞 本研究は JSPS 科研費 15H01684 の助成を受けたものです．

参考文献

[1] McAfee Labs: McAfee Labs 脅威レポート 2018 年 6 月 ，https://www.mcafee.com/enterprise/ja-jp/assets/reports/rp-quarterly-threats-jun-2018.pdf (2018/08/18 参照)
[2] トレンドマイクロ株式会社：EDR とは 標的型サイバー攻撃の次の一手，https://www.trendmicro.com/ja_jp/about/trendpark/apt-201607-01-01.html (2018/08/18 参照)
[3] トレンドマイクロ：標的型サイバー攻撃最新動向，https://www.trendmicro.com/ja_jp/about/trendpark/apt-201606-01-01.html (2018/08/19 参照)
[4] Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin: Intelligence-Driven Computer Network Defense Informed by Analysis of Adversary Campaigns and Intrusion Kill Chains, https://www.lockheedmartin.com/content/dam/lockheed-martin/rms/documents/cyber/LM-White-Paper-Intel-Driven-Defense.pdf (2018/08/18 参照)
[5] 船越 絢香，中村 祐，竹田 春樹：標的型攻撃で用いられたマルウエアの特徴と攻撃の影響範囲の関係に関する考察，コンピュータセキュリティシンポジウム (2015)
[6] 西野 琢也，菊地 亮太，丸橋 弘治，福田 大輔，齊藤 聡美，鳥居 悟，伊豆 哲也：テンソル分解に基づくグラフ分類による組織内ネットワーク攻撃活動検知，コンピュータセキュリティシンポジウム (2017)
[7] 川北 将，島 成佳：テクニカル分析によるサイバーインシデント予測手法の検討，電子情報通信学会総合大会 (2016)
[8] 芦野 佑樹，島 成佳：インターネットノイズに対する偽装応答機能の実装と観測に基づいた意図が不明なリクエストに関する考察，SCIS(2015)
[9] 渡部 正文，島 成佳，田辺 瑠偉，吉岡 克成：マルウェア活動背景にある攻撃者の意図に関する知見のナレッジ化方式，コンピュータセキュリティシンポジウム (2014)
[10] 藤野 朗稚，森 達哉：自動化されたマルウェア動的解析システムで収集した大量 API コールログの分析，コンピュータセキュリティシンポジウム (2013)
[11] 青木 一樹，後藤 滋樹：マルウェア検知のための API コールパターンの分析，電子情報通信学会総合大会 (2014)
[12] 川古谷 裕平，塩治 榮太朗，岩村 誠，針生 剛男：API コール間のデータ依存関係を利用したマルウェア通信内容の特定，コンピュータセキュリティシンポジウム (2013)
[13] 高田 雄太，寺田 真敏，松木 隆宏，笠間 貴弘，荒木 粧子，畑田 充弘：マルウェア対策のための研究用データセット ～MWS Datasets 2018～，情報処理学会，Vol.2018-CSEC-82, No.38, 2018 年 7 月.(2018)
[14] 鮫島 礼佳，金居 良治，吉浦 裕，市野 将嗣：マルウェアによる感染活動の目的推定のためのトラヒックデータと API ログに基づく分析，コンピュータセキュリティシンポジウム (2016)
[15] 佐々木 良一：現代電子情報通信選書「知識の森」 ネットワークセキュリティ，オーム社 (2014)
[16] scikit-learn：http://scikit-learn.org/
[17] 鮫島 礼佳，畑田 充弘，吉浦 裕，市野 将嗣：感染挙動の時系列情報のクラスタリングに基づくマルウェア検知手法，コンピュータセキュリティシンポジウム (2015)

© 2018 Information Processing Society of Japan