L-024

# マルウェアの挙動情報と可視化を用いたマルウェア分類システム
# Malware Classification System Based on Behavior and Visualization of Malware

三島悠 † 田村尚規 † 甲斐博 † 森井昌克 ‡
Haruka Mishima Naoki Tamura Hroshi Kai Masakatsu Morii

## 1. はじめに

インターネットの急速な普及により多種多様なマルウェアが蔓延し，その被害は年々増加している. マルウェアとはウイルス，ワーム，トロイの木馬等の悪意を持って作成されたソフトウェアやコードの総称である. マルウェアの分類方法には，マルウェアを実際に動作させる動的解析やプログラムコードを分析する静的解析などによって得られた情報から分類を行う手法が多数提案されている. 例えば，正力らの提案手法 [1] は動的解析結果から API の呼び出し履歴や挙動情報を用いた分類システムを提案している.

本研究では，マルウェアの動的解析結果の情報をもとにマルウェアの可視化を行い，可視化画像からディープラーニングを用いた学習済みモデルにより既知のマルウェアとの亜種判定を行うことを目標とした新しい分類システムを提案する.

## 2. マルウェアの可視化手法

使用する動的解析ログとして，FFRI が提供している FFRI Dataset2015[3] を用いる. このデータセットには FFRI が収集したマルウェアの動的解析ログが JSON 形式で 3000 検体分用意されており，検体が実行時に生成したファイルや呼び出した API などのマルウェアが実験環境で行った振る舞いが詳細に記録されている.JSON 形式の動的解析ログの主な項目を表 1 に示す.

表 1: FFRI 動的解析ログの項目

| 項目 | 内容 |
|---|---|
| info | 解析の開始, 終了時刻等 |
| target | ハッシュ値等の検体のファイル情報 |
| virustotal | 様々なセキュリティソフトベンダーの検査結果等 |
| static | インポート API 等の検体のファイル情報 |
| dropped | 検体が実行時に作成したファイル |
| network | 検体が実行時に行った通信の情報 |
| behavior | 検体実行時の API ログや参照したファイル，レジストリ情報 |

項目の 1 つである behavior の中にある項目 calls にはマルウェアが呼び出した API が，呼び出した順番に記されている.calls に記録されている API の種別となるカテゴリ (項目 category) と API の名称 (項目 api) の情報を 3D モデルに対応させ，球体上に配置することでマルウェアの可視化を行う.

本論で実験に用いた 407 個の動的解析ログでは，process，system，registry など 12 種類の API のカテゴリが存在した. また，API の名前は,GetCursorPos，NtCreateSection，NtCreateFile など 57 種類が存在する.

可視化する際には，API のカテゴリをモデルの形状に，API の名前をモデルの色に対応させる. 例えば，カテゴリの「filesystem」を円柱型で表現し,「registry」は菱形で表現する. また，API の名前が「NtCreateFile」ならオレンジ色,「NtCreateKey」ならピンク色により円柱や菱形の 3D オブジェクトに色付けをする.

また，呼び出された API の順番にモデルを球状の物体に上部から下部にかけて螺旋状に配置し，API ログの時系列成分を可視化モデルに反映する.

実際に動的解析ログから可視化した例を図 1 に示す.

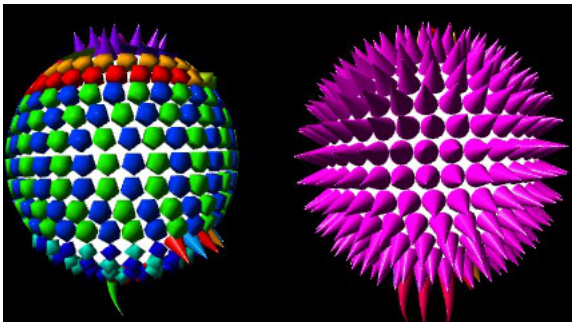

図 1: 可視化例

(左から Win32/Simda.B,Win32/Spy.Zbot.ACB)

## 3. 可視化による分類システムの提案

本研究では 2 節で示したマルウェアの可視化手法を利用して，マルウェアを亜種に分類するシステムの提案をする. 提案するシステムの流れを以下の手順に示す.

1. 動的解析ログからセキュリティソフトベンダーの分類結果のマルウェア名，API の名前とカテゴリとを抽出し，データベースに保存する.
2. 保存したデータから 2 節で示した可視化方法を用いて可視化した画像を保存する. 保存した画像と対応するマルウェア名を教師データとして Chainer[4] により学習を行う.
3. 学習済みモデルに対し，未知のマルウェアの動的解析ログから得た可視化画像を適用し，対象のマルウェアが既知のマルウェアのどの亜種であるかの分類を行う.

†愛媛大学，Ehime University
‡神戸大学，Kobe Unirversity

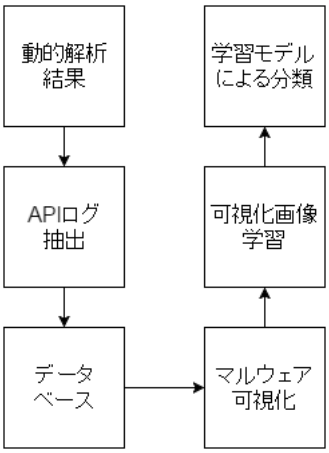


図 2: 提案システムの流れ

手順 1 で抽出するマルウェア名とは，表 1 で示した virustotal の項目に含まれる様々なセキュリティソフトベンダーが分類したマルウェアの名称である. 本研究では，この結果をディープラーニング学習時の正しい指標として用いる. 本実験では，ESET-NOD32 が分類した名前を用いて，3000 個の解析結果のファイルから，名前が同じもので数が多い上位 5 つの動的解析ログを可視化し学習を行った. 今回使用したマルウェアとそのファイル数を表 2 に示す.

表 2: 実験で用いた動的解析ログ

| マルウェア名 (ESET-NOD32) | ファイル総数 |
|---|---|
| Win32/Spy.Zbot.ACB | 114 |
| Win32/Spy.Zbot.AAQ | 90 |
| Win32/Simda.B | 73 |
| Win32/Neurevt.B | 70 |
| Win32/Agent.VPS | 60 |

表 2 の合計 407 個のファイルを Chainer[4] で提供される画像分類プログラム imagenet を用いて学習しテストを行った. 本実験では，307 個の可視化画像を訓練用データとし，残りの 100 個をテスト用データとして用いた. テストにおける分類結果の精度は 75%となった.

## 4. まとめ

本研究では，FFRI により提供されている動的解析ログを用いて，可視化モデルを用いたマルウェア分類システムの提案を行った. 但し，API のみによる可視化，及び分類を行ったが図 3 に示すようにセキュリティソフトベンダーの分類が同じものでも，可視化結果が大きく異なる場合が多数存在した.

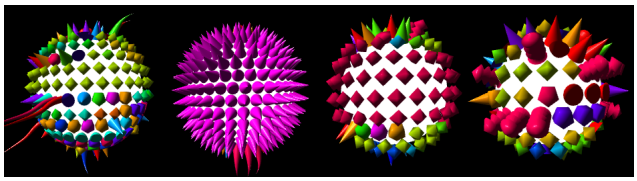

図 3: Win32/Spy.Zbot.ACB の解析結果の可視化結果

また，図 4 に示すように，違う分類結果でもよく似ている可視化結果になる場合も存在した.

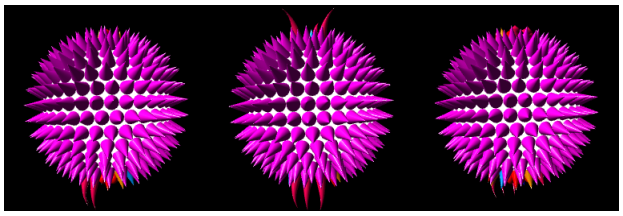

図 4: 異なる分類で類似した可視化結果

(左から Win32/Spy.Zbot.ACB,Win32/Agent.VPS, Win32/Spy.Zbot.AAQ)

このように，分類に使用する指標として API を用いるのは有効な場合もあるが，十分な特徴を表現できない場合もあると考えられる. 動的解析結果中の API 以外の，アクセスしたレジストリやファイル，生成したファイルなど他の解析情報を同様に可視化することにより類似性が確認できると考えられる.

また，他の提案されている分類システムとの比較を行い，本システムの評価を行うことが今後の課題である.

## 参考文献

[1] 正力達也，伊沢亮一，森井昌克，”マルウェアの挙動情報を用いたマルウェア分類システムの提案と実装”，SCIS2011,2011.

[2] 松重雄大，浦辻和也，甲斐博，森井昌克 ”Malware visualization based on the behavior and its classification”，第 13 回情報科学技術フォーラム (FIT2014),2014.

[3] マルウェア対策研究人材育成ワークショップ 2015(MWS2015), http://www.iwsec.org/mws/2015.

[4] Chainer, http://chainer.org/.