通信先と端末内の挙動との依存関係に基づくマルウェアダウンロードサイト特定手法

幾世 知範† 青木一史† 八木 毅† 針生 剛男†

†NTT セキュアプラットフォーム研究所
180-8585 東京都武蔵野市緑町 3-9-11
{ikuse.tomonori, aoki.kazufumi, yagi.takeshi, hariu.takeo}@lab.ntt.co.jp

あらまし マルウェアはマルウェアダウンロードサイトからプログラムコードを随時取得し，機能の拡充を行う．これまで，機能拡充の阻止を目的としたマルウェアダウンロードサイトのリスト作成では，マルウェアを実行して通信先を分析する動的解析が行われてきた．しかし，マルウェアは悪性なプログラムコード以外にも正規ライブラリを取得する場合があるため，動的解析で得られるプログラムコード取得元の全てが悪性とは限らない．そこで，本稿では動的解析時の通信先とプログラムコードおよびファイルの依存関係において，既存の悪性情報との合致項目を起点とした通信先の悪性判定手法を提案する．MWS データセットと独自に収集した検体を用いた検証の結果，提案手法で発見可能なマルウェアダウンロードサイトの存在を確認した．

Malware Download Site Detection Based on Dependencies between Remote Servers and Malware Behavior

Tomonori Ikuse† Kazufumi Aoki† Takeshi Yagi† Takeo Hariu†

†NTT Secure Platform Laboratories
3-9-11, Midori-cho, Muashino-shi, Tokyo 180-8585, JAPAN
{ikuse.tomonori, aoki.kazufumi, yagi.takeshi, hariu.takeo}@lab.ntt.co.jp

Abstract Malware program downloads another program codes from malware download sites to update their functionalities. Generally, malware download sites have been identified using malware dynamic analysis. Because malware program downloads legitimate libraries from legitimate sites, not all remote servers which provide program codes to malware are malware download sites. In this paper, we propose a method to identify malware download sites based on known malicious information and runtime dependency between remote servers, programs and files. We evaluated the method with MWS datasets and our original datasets. Our evaluation showed that our method is able to find out a malware download site that was not known as the site.

1 はじめに

ボットやダウンローダをはじめとするマルウェアの多くは，悪性なプログラムが設置されたサイト（以降，マルウェアダウンロードサイトと呼ぶ）からプログラムコードを取得･実行し，機能拡充を行う．機能拡充では，外部サーバへの攻撃や情報搾取など更なる被害をもたらすため

の機能が追加される．このため，感染後の被害を最小限に抑えるには，マルウェアダウンロードサイトへの通信を妨害し，機能拡充を阻害しなければならない．

これまでは，マルウェアダウンロードサイトへの通信を妨害するために，マルウェアの動的解析により得られた通信先をブラックリスト化する対策が講じられてきた．しかしながら，マルウェアには正規サイトから正規ライブラリを取得し，実行するものが存在するため，動的解析で得られた通信先の全てをブラックリスト化すると，通信の誤遮断を引き起こす．

本稿では，テイント解析技術を用いて通信先とダウンロードデータの依存関係を抽出し，抽出された依存関係において既存の悪性情報との合致項目を起点とした分析によるマルウェアダウンロードサイト特定手法を提案する．本手法では，まず，マルウェアの動的解析中に観測された OS 上のオブジェクト（ファイルとメモリ上のプログラムコード）と通信先間の依存関係を抽出する．その後，公開ブラックリスト等により通信先やオブジェクトの悪性判定を実施し，悪性と判定された通信先やオブジェクトを起点に依存関係を遡ることでプログラムコードの取得元をマルウェアダウンロードサイトと判定する．これにより，動的解析で得られた通信先のうち，一部が悪性と判定できれば，関連する全てのサイトを悪性と判定できる．

提案手法を用いた実験では，MWS データセットと独自に収集した検体を動的解析し，依存関係の抽出を行った．さらに，通信先の悪性判定として公開ブラックリストを利用し，依存関係の遡りによるダウンロードサイト特定の有効性検証を行った．実験の結果，これまでマルウェアダウンロードサイトと特定できていなかったサイトを本手法で発見できることを確認した．

2　提案手法

2.1　概要

提案手法の概要を図 1 に示す．提案手法は依存関係グラフ構築処理と，悪性判定処理から成る．依存関係グラフ構築処理では, マルウェアを動的解析し，受信データに関わるデータの受け渡し関係を解析することで依存関係グラフを生成する．悪性判定処理では，依存関係グラフに対して既知の悪性情報をマッピングし, 悪性と判断されたノードから依存関係を遡り，悪性と判定されたノードに関わるプログラムコードを特定する．その後，悪性と判定されたプログラムコードの取得元をマルウェアダウンロードサイトと判定する．

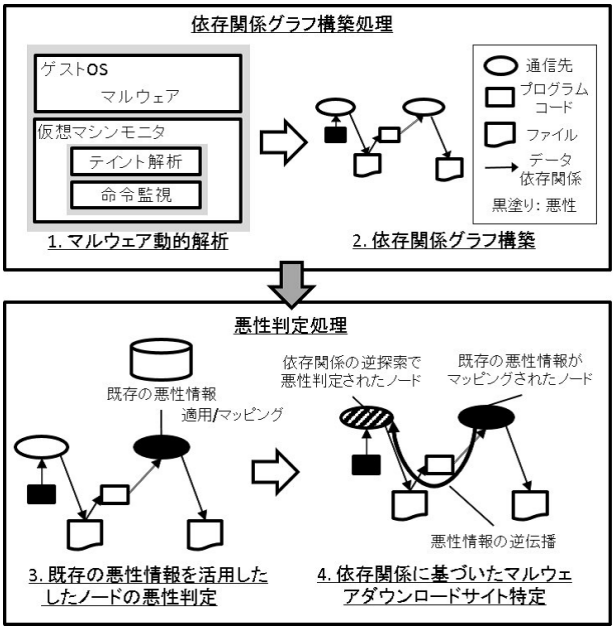

図 1: 提案手法概要

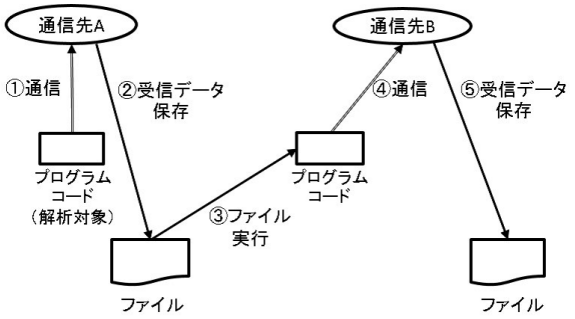

図 2: 新たなプログラムコード取得に関わる一連の動作

図 2 に，依存関係グラフの例を示す．この例は，通信先 A からダウンロードされたファイルが実行され，通信先 B から新たなファイルを取

得するまでの一連の依存関係を表している．通信先Bから取得したファイルが悪性と判定されるプログラムコードであれば，通信先Bをマルウェアダウンロードサイトと判断できる．もしくは，通信先Bが公開ブラックリストに掲載されている場合には，通信先Bを悪性な通信先と判断できる．提案手法では，通信先Bから取得されるファイルが悪性なプログラムコードである場合，または通信先Bが公開ブラックリストと合致した際に，一連の依存関係を遡り，通信先Aもマルウェアダウンロードサイトと判定する．

以下では依存関係グラフ構築処理と悪性判定処理の詳細を説明する．

2.2 依存関係グラフ構築処理

既知の悪性情報に基づいてマルウェアダウンロードサイトを特定するための依存関係グラフと構築手法について説明する．

2.2.1 依存関係グラフの構成

提案手法は，通信先とオブジェクトの関係を明らかにし，通信先またはオブジェクトが悪性と判断された際に，まだ悪性と判定されていない通信先ないしオブジェクトを悪性と判定する．本手法では依存関係グラフに対して既存の悪性情報をマッピングし，既存の悪性情報と合致した箇所を起点として関連するマルウェアダウンロードサイトを特定する．そのため，既存の悪性情報をマッピングできる粒度のノードを定義し，依存関係グラフを構成する．

また，提案手法では2者間のデータ依存関係をエッジにとる．エッジで表現される依存関係には，終点ノードの種別に応じて3つの種類が存在する（表1）．

- データ実行に関わる依存関係
  プログラムコードの取得元情報を保持する本依存関係は，データの実行有無を表す．この依存関係には，通信先からの受信データがメモリ上で直接実行される場合とファイルから読み込んだデータが実行される場合，別プログラムによってインジェクションされたデータが実行される場合が該当する．この依存関係は終点ノードにプログラムコードを持つエッジにより表現される．

表 1: 依存関係グラフ上に保持される依存関係

始点ノード	終点ノード	内容
通信先	プログラム	データ実行
ファイル	プログラム	データ実行
プログラム	プログラム	データ実行
通信先	ファイル	データ保存 (受信データ保存)
ファイル	ファイル	データ保存 (コピー)
プログラム	ファイル	データ保存 (自己切り出し)
通信先	通信先	通信先指定
ファイル	通信先	通信先指定
プログラム	通信先	通信先指定

- データ保存に関わる依存関係
  本依存関係はファイルデータの取得元情報を保持する．この依存関係を保持することで，ファイルの悪性判定結果に基づいてファイル内データの取得元を悪性判定することが可能となる．通信先から受信したデータがファイルに保存される場合，ファイルのコピーが行われる場合，プログラムコードが自分自身をファイルとして切り出す場合がこの依存関係に該当する．この依存関係は，終点ノードにファイルを持つエッジにより表現される．

- 通信先決定に関わる依存関係
  本依存関係は，通信先情報の出自を保持する．通信先の悪性判定結果に基づいて悪性を判断する場合，通信内容の出自ではなく，通信先を決定した通信先情報の出自が重要となる．本依存関係は，通信先の悪性情報に基づいて通信先情報の出自の悪性判定を可能とする．通信先やファイル，プログラムコードによる通信先の決定がこの依存関係に該当し，終点ノードに通信先を持つエッジが本依存関係を表現する．

以上の依存関係は，依存関係グラフ内に保持される．このような依存関係グラフにより，システム内の情報だけでなく，通信先に関する情

報も考慮した悪性判定が可能となる．

2.2.2 依存関係グラフの構築

依存関係グラフは，マルウェアの動的解析で得られる解析ログに基づいて構築する．動的解析では，マルウェア実行時の API 呼び出しの監視と，テイント解析技術を用いたデータ依存関係の解析を行う．テイント解析技術とはデータに対してタグと呼ばれる属性情報を付与し，解析システム内でのデータの伝播を追跡する解析技術である．テイント解析技術を利用する場合，タグを設定する場所であるテイントソース，データに付与されているタグを確認する場所であるテイントシンク，タグを伝播させる際の伝播ルールを目的に応じて設定する必要がある．以下では，表 1 に示した依存関係を特定するために，以下のようなテイントシンクとテイントソース，および伝播ルールを用いた．

テイントソース

テイントソースはデータ受信 API とファイル書き込み API の戻り値とする．データ受信 API の戻り値に対し，受信データの取得元を一意に特定可能なタグと監視対象であることを表すタグを設定することで，受信データの利用用途の特定を実現する．また，ファイル書き込み API の戻り値に対し，ファイルに書き込まれたデータであることを示す情報をタグに追加することでファイルから取得したデータであるか否かの区別を実現する．データがファイルに由来するものかどうかを識別することにより，ファイルに関する既存の悪性情報の適用対象であるか否かを判断する．

テイントシンク

テイントシンクは EIP レジスタの指すメモリ領域およびファイル書き込み API の引数，通信用 API の引数とする．EIP レジスタの指すメモリ領域に受信データを表すタグが付与されていることを確認することで，タグに紐付く通信先からの取得データが実行されたことを特定する．また，ファイル書き込み API および通信用 API の引数として利用されたデータにタグが付与されていることを確認することで，ファイル書き込みデータの出自および通信先情報の出自を特定する．

伝播ルール

一般的にテイント解析ではテイント伝搬が途切れる場合がある．伝播途切れが発生するとデータ依存関係の見逃しが発生するため，伝播途切れをできる限り回避しなければならない．提案手法では監視の継続性を確保するために，積極的なタグ伝播を行う．

具体的には，演算命令におけるタグ伝播とメモリ書き込み時のタグ伝播を適用する．演算時におけるタグ伝播では，データのコピーや演算時に，命令オペランドのいずれかに監視対象タグが設定されていた場合，書き込み先に監視対象タグを伝播させる．また，メモリ書き込み時のタグ伝播では，プログラムコードのアンパックや自己改変が行われることを想定し，監視対象タグの付与された命令がメモリ上にデータを書き込んだ際にも監視対象タグを伝播させる．

また，解析対象のプログラムコードを識別するため，動的解析時に最初に実行されるプログラムコードに対して，受信データと同様に監視対象タグを設定する．監視対象タグを付与されたデータが EIP レジスタに設定された場合に，当該 EIP レジスタの値が指し示す命令を監視対象として認識する．動的解析時には，この監視対象タグが付与された命令から呼び出された API を記録する．

監視対象タグを用いた解析を行う場合，監視対象タグのついたプログラムコードが監視対象ではないプログラムコードを呼び出す場合がある．この場合は，呼び出し先で実行された API 呼び出しは監視対象タグのついた呼び出し元プログラムコードの動作として記録する．これにより，監視対象のプログラムコードによって引き起こされた動作のみの記録する．

以上のように，テイント解析技術を用いて依存関係グラフ構築に必要な情報を動的解析中に収集し，依存関係グラフを構築する．

2.3 悪性判定処理

本処理では，依存関係グラフ構築処理で生成された依存関係グラフと，公開ブラックリストやアンチウイルスソフト等の既知の悪性情報を

用いて，マルウェアダウンロードサイトの特定を行う．

本処理では，まず，依存関係グラフの各ノードに対して既知の悪性情報をマッピングする．既知の悪性情報としては，公開ブラックリストに掲載されている通信先情報（IP アドレス，FQDN，URL) が挙げられる．さらに，ファイル自体の悪性判定を目的としたアンチウイルスソフトによる各ノードの悪性判定結果や，各種ヒューリスティックスを活用したプログラムコードの悪性判定結果が利用される．

次に，悪性と判定されたノードが出現する原因となったプログラムコードが悪性か否かを判定する．この判定処理では，依存関係グラフを遡り，悪性判定されたノードを生成する原因となったファイルを特定して悪性判定を行う．

最後に，悪性と判定されたファイルを取得した際の通信先をマルウェアダウンロードサイトと判定する．

3 実装

提案手法の実装には，オープンソースのテイント解析環境 TEMU[1] を利用した．また，API の呼び出し監視には API Chaser[2] と同様の仕組みを用いた．

なお，提案システムでは 2.2.2 節で述べた解析を行うため，主に表 2 に示した API を監視する．ここで，WSASend や WSARecv だけでなく，InternetOpenUrl 等を監視するのは，HTTP 通信に関して通信内容を分析することなく通信先の URL 情報を取得するためである．また，CreateProcess などのプロセス関連の API を監視するのは，監視対象でないプログラムコードの動作を呼び出し元の監視対象プログラムコードに紐付けるためである．

現段階では提案手法の有効性確認のために，提案手法として実装すべき機能のうち，テイント解析に基づいた依存関係解析機能と通信先に対する既存の悪性情報のマッピング機能，および公開ブラックリストと合致した通信先を起点とした依存関係の遡りによるダウンロードサイト特定機能を実装している．

表 2: 主要な監視対象 API

カテゴリ	API 名
ネットワーク	WSASend, WSASendTo
	WSARecv, WSARecvFrom
	gethostbyname
	getaddrinfo
	InternetOpenUrl
	InternetConnect
	InternetReadFile
	InternetWriteFile
ファイル	CreateFile
	WriteFile
プロセス	CreateProcess
	CreateRemoteThread
	ShellExecute

4 実験

4.1 実験概要

提案手法の有効性を検証するために，提案システムを評価した．本実験では，MWS2014 データセット D3M[3] で提供されている検体ハッシュ値と同じハッシュ値を持つ 31 検体（以降，D3M 検体と呼ぶ）と，2014 年 6 月 18 日から 2014 年 8 月 18 日までの期間にインターネット上の Web サイトを Web クライアント型ハニーポット Marionette[4] で巡回して採取した 327 検体を用いて，新たなマルウェアダウンロードサイトを発見できるか否かを実験した．なお，本実験では，解析中のマルウェアによるインターネットへの攻撃実施を回避するために，開環境型サンドボックス BotnetWatcher[5] の通信制御機能を利用して，解析環境からインターネット上の正規サイトへの通信に制限をかけた状態でマルウェアを動的解析した．また，本実験では動的解析の実施時間を 30 分間とした．

4.2 実験結果

4.2.1 D3M 検体の解析結果

D3M 検体 31 検体の内，本実験の解析環境下で外部サーバと解析を行った検体は 11 検体存在した．しかし，正常なレスポンスを得られてい

る検体は 6 検体のみであった．さらに，正常に応答が得られているのは，j.maxmind.com および api.wipmania.com と通信した場合に限られており，外部からのファイル取得等の動作は見られなかった．j.maxmind.com と通信を行った検体は IP アドレスを直接指定し，独自プロトコルで外部サーバとの通信を試みていたが，BotnetWatcher による通信制御により独自プロトコルの通信を遮断したため，結果的に外部サーバとの通信は行われなかった．

4.2.2 独自に収集した検体の解析結果

独自に収集した 327 検体の内，本実験の環境下で外部サーバとの通信を行った検体は 214 検体であった．その内，データをダウンロードし，ファイルに書き込んだ検体は 141 検体存在した．また，141 検体の内の 51 検体がダウンロードしたファイルの実行を行った．

次に，ダウンロードされたプログラムコードの通信先が悪性であった場合に取得元を悪性と判定する手法の評価結果を述べる．本実験では通信先の悪性判定は CleanMX[6] と hpHosts[7] に掲載されているホスト名の参照により実施した．その結果，51 検体のうちの 1 検体において，CleanMX および hpHosts に掲載されていないマルウェアダウンロードサイトを特定できたことを確認した．

提案手法でマルウェアダウンロードサイトを特定できたケースにおいて，悪性判定に関わるノードのみを記した依存関係グラフを図 3 に示す．この依存関係グラフでは，解析対象の検体が support.costmin.info からプログラムコードを取得して実行し，それによって新たに実行されたプログラムコードが datadownloadscan.info と通信したことを表している．実行されたプログラムコードが通信した datadownloadscan.info は hpHosts に掲載されていた．そのため，support.costmin.info から取得したプログラムコードを悪性と判定し，プログラムコードの取得元である support.costmin.info をマルウェアダウンロードサイトと判定した．Clean MX および hpHosts には当該ホストは掲載されていない．

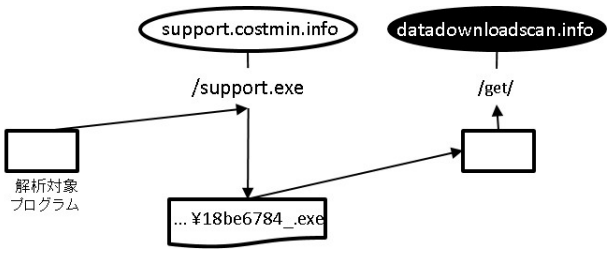

図 3: 通信先の新規悪性判定が実現された場合の依存関係グラフ

5 関連研究

テイント解析を応用してマルウェアの通信先の悪性度を判定する手法として JackStraws[8] が提案されている．JackStraws では，テイント解析を用いて通信に関連するシステムコール間のデータ依存関係を解析して依存関係グラフを構築することで，マルウェアの通信先とマルウェアの挙動の関連付けを実現している．本手法では，構築した依存関係グラフにおいて，既知の C&C サーバと通信する際に構築された依存関係グラフを事前に収集して教師データとすることで，C&C サーバとの通信に特徴的な依存関係グラフのテンプレートを生成し，マルウェアダウンロードサイトを含む C&C サーバの検知に利用している．JackStraws と提案手法は，教師データを用いた学習を必要とするか否かという点で異なる．JackStraws では，教師データを予め用意する必要があることに加え，教師データや学習時のパラメータに依存して見逃しが発生する可能性があることも示されている．感染端末の早期発見による被害の抑制のためには，様々なマルウェアに対処可能な教師データの収集や，マルウェアの挙動に応じた高速なパラメータ再設計が必要となるが，マルウェアの種類数が爆発的に増加している現状を考慮すると，実現は非常に困難である．

テイント解析を用いてデータ依存関係を解析するシステムとして Panorama[9] が提案されている．Panorama は，プログラムコードの動作を解析し，情報を漏えいするプログラムコードを悪性と判定するシステムである．Panorama は，提案手法と同様に，依存関係グラフの構築

と，既存の悪性情報を用いた悪性判定を行う仕組みを採用している．しかし，プログラムコードの悪性判定を目的としており，通信先の悪性判定を実現できる機能は具備していない．具体的には，受信データがファイルに格納されるまでの一連の流れは追跡できるが，受信したデータの実行に関わる依存関係の分析を行う機能は具備していない．このため，ダウンロードしたデータに由来する通信先間の依存関係を解析することができず，通信先の悪性判定を実施できない．

6 考察

- **提案手法の有効性**

  本実験を通して，提案手法によって発見可能なマルウェアダウンロードサイトの存在を確認できたが，確認できたのは1検体についてのみであった．この原因の1つとして，検体の鮮度の問題が考えられる．マルウェアダウンロードサイトが発見され，通信遮断等の対策が行われることに対し，攻撃者もマルウェアダウンロードサイトのホスト名を変更するなどの対策をとっている．したがって，マルウェア取得から期間が経過した場合，通信先への接続性が失われる可能性が高い．そのため，可能な限り検体取得から解析までの時間差を無くした上での評価を今後実施する予定である．

- **テイント解析に起因する制限事項**

  圧縮や暗号化･復号処理においてテイントタグの伝播途切れが発生することが知られている．本研究では，テイントタグを積極的に伝播させる手法を実装しているが，テイントタグの伝播途切れを完璧に解決することはできない．そのため，本実験において，伝播途切れが少なからず発生している可能性がある．この伝播途切れに対処することで，見逃していた依存関係が明らかとなり，新たなマルウェアダウンロードサイトを発見できる可能性がある．今後は，特定のAPI呼び出しにおける入出力の依存関係を強制的に構築する [10] 手法やデータ依存関係を精緻に解析して伝播ルールを変更 [11] する等の手法の利用を検討する．

- **提案手法と既存の悪性情報の関係性**

  提案手法は，既存の悪性情報を利用し，プログラムコードの悪性判定とそれに基づいたマルウェアダウンロードサイト特定を行う手法である．そのため，提案手法は既存の悪性情報を拡張する手法だと言える．しかしながら，提案手法は既存の悪性情報を信頼できる情報として扱うため，提案手法の出力結果は既存の悪性情報の信頼性の影響を直接受けてしまう．そのため，本手法によりマルウェアダウンロードサイトを特定する際には，使用する悪性情報の選定を適切に行う必要がある．

- **既存手法との連携**

  5節では，JackStraws と Panorama に関して提案手法との違いを説明した．しかし，いずれの手法に関しても，提案手法での活用，もしくは連携が可能である．まず，JackStraws に関しては，提案手法を用いて特定されたマルウェアダウンロードサイトの情報とその際に記録されたAPI間のデータ受け渡し関係が，JackStraws の教師データとして活用できる．また，JackStraws を使って特定された通信先の悪性情報は，提案手法における依存関係グラフへの悪性情報マッピング時に活用できる．

  一方，Panorama に関しても，提案手法での活用が可能であると考えられる．Panorama はプログラムコードの悪性判定を行う仕組みである．そのため，提案手法における依存関係グラフへの悪性情報マッピング時に悪性判定手法として活用できる．

7 まとめ

本稿では，通信先やダウンロードデータの依存関係を既存の悪性情報と合致した箇所を起点に遡り，マルウェアダウンロードサイトを特定する手法を提案した．MWS データセットと独自に収集した検体を用いて提案手法の有効性の検

証を行い，提案手法で発見可能なマルウェアダウンロードサイトの存在を確認した．今後は，テイントタグの伝播途切れに対する対策の検討およびアンチウイルスソフトによるファイル悪性判定結果を起点とした分析の有効性評価を行う．

参考文献

[1] D. Song, D. Brumley, H. Yin, J. Caballero, I. Jager, M.G. Kang, Z. Liang, J. Newsome, P. Poosankam, and P. Saxena, “BitBlaze: A New Approach to Computer Security via Binary Analysis,” Proc. ICISS, pp.1–25, dec 2008.

[2] Y. Kawakoya, M. Iwamura, E. Shioji, and T. Hariu, “API Chaser: Anti-analysis Resistant Malware Analyzer,” Proc. RAID’13, pp.123–143, 2013.

[3] 秋山満昭，神薗雅紀，松木隆宏，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 2014〜,” 情報処理学会 研究報告コンピュータセキュリティ (CSEC)，pp.1–7，jun 2014.

[4] M. Akiyama, Y. Kawakoya, and T. Hariu, “Scalable and Performance-Efficient Client Honeypot on High Interaction System,” Proc. SAINT’12, pp.40–50, jul. 2012.

[5] 青木一史，川古谷裕平，岩村誠，伊藤光恭，“半透性仮想インターネットによるマルウェアの動的解析,” コンピュータセキュリティシンポジウム 2009 論文集，pp.1–6，oct 2009.

[6] “Clean mx”. http://support.clean-mx.de/clean-mx/viruses.php

[7] “hphosts”. http://www.hosts-file.net

[8] G. Jacob, R. Hund, C. Kruegel, and T. Holz, “JACKSTRAWS: Picking Command and Control Connections from Bot Traffic,” Proc. USENIX Conference on Security, pp.29–29, aug 2011.

[9] H. Yin, D. Song, M. Egele, C. Kruegel, and E. Kirda, “Panorama: Capturing System-wide Information Flow for Malware Detection and Analysis,” Proc. CCS’07, pp.116–127, oct 2007.

[10] 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 論文集，pp.745–752，oct 2013.

[11] M.G. Kang, S. McCamant, P. Poosankam, and D. Song, “DTA++: Dynamic Taint Analysis with Targeted Control-Flow Propagation,” Proc. NDSS’11, pp.0–0, feb 2011.