リンク構造を用いた悪性 Web サイトの検知法

伊藤 大貴[†1] 永井 達也[†1] 高野 泰洋[†1] 神薗 雅紀[†2]
毛利 公美[†3] 白石 善明[†1] 星澤 裕二[†2] 森井 昌克[†1]

概要：Web サイトの閲覧によるマルウェア感染が多発しており，悪性 Web サイトの脅威が深刻化している．攻撃者は頻繁に Web サイトを更新し，未知の悪性 Web サイトを新たに生成しうる．従って，被害を未然に防ぐことは容易ではない．本研究では悪性 Web サイトのリンク構造には互いに類似性があると想定し，リンク構造を用いた悪性 Web サイト検知法について検討する．提案手法では，ニューラルネットワークを用いた教師付き学習によって悪性 Web サイトを検知する．最も単純な 3 層のニューラルネットワークで，Web クローラーを用いて収集した実際のリンク構造データを用いたときの悪性 Web サイトの正解率は 82%であった．

キーワード：Drive-by-Download 攻撃，リンクマイニング，機械学習，ニューラルネットワーク，ディープラーニング

A Malicious Web Site Detection Technique using Link Structure

Daiki Ito[†1] Tatsuya Nagai[†1] Yasuhiro Takano[†1] Masaki Kamizono[†2]
Masami Mohri[†3] Yoshiaki Shiraishi[†1] Yuji Hoshizawa[†2] Masakatu Morii[†1]

Abstract: Threat of malicious websites has become a serious security problem since browsing the malicious websites causes malware infection epidemically. Attackers frequently updates their website and can generate unknown malicious websites newly. It is, therefore, difficult to prevent from the infection. By assuming that there exists affinity between the link structures of malicious websites, this paper proposes a new technique to detect malicious websites using the link structure. The proposed method can detect unknown malicious websites by a supervised learning using the neural network. The experiment results show that the detection rate of malicious web site is 0.82 for real link structure data obtained in July 2016.

Keywords: Drive-by-download attack, Link mining, Machine learning, Neural network, Deep learning

1. はじめに

Web サイトの閲覧によるマルウェア感染の事例が多発しており，悪性 Web サイトの脅威がますます深刻化している．これらの被害を防ぐために，危険性が高いサイトの URL をブラックリストとして共有する取り組みが行われている．しかし，攻撃者は頻繁に悪性 Web サイトを変更するため，その URL は短期間で消滅・遷移するという特徴があり，悪性 Web サイトをブラックリストによって完全に検知することは困難である．また，正規の Web サイトが改ざんされることで悪性 Web サイトへ誘導し，マルウェアに感染する事例は後を絶たず，悪性 Web サイトによる被害を未然に防ぐことは容易ではない．

このような悪性 Web サイトの構築には Exploit Kit が用いられる場合も多い．Exploit Kit とは，様々なブラウザ及びプラグインの脆弱性に対応できるよう，複数の攻撃コードがパッケージ化されたツールである．専門的な知識や技術を必要としないため，この Exploit Kit を利用することで，攻撃基盤を容易に構築することが可能となる．一方で，Exploit Kit を利用して生成された悪性 Web サイト群の URL やコンテンツには類似性があることが確認されており，その類似性に基づいた悪性 Web サイト検出手法や解析手法が提案されている．芹田ら[1]は悪性 Web サイトの URL のパス部分の類似性に着目し，URL の構造に基づいたクラスタリングを行い，その結果から正規表現を自動で生成する手法を提案している．佐藤ら[2]は Exploit Kit を用いて作成された Web サイト群の URL の類似性に着目し，URL パス・クエリから作成した決定木を用いた悪性 URL 検出法を提案し，89.02%の精度で悪性 URL を悪性と判定可能であることを実際の通信データを用いた実験により示している．今野ら[3]は Exploit Kit を利用して作成された悪性 Web サイトのコンテンツがテンプレートを用いて作られていることに着目して分析し，悪性コンテンツ間に類似性があることを報告している．

しかし，Exploit Kit の中には新しい脆弱性を利用した攻撃コードを次々と追加し更新されるものも確認されている．そのため，Exploit Kit によって作成されたコンテンツ等の類似性に基づいた手法では，悪性 Web サイトを漏れなく検知・防御するために相応のコストがかかる．そこで，本稿では，悪性 Web サイトのリンク構造には互いに類似性があ

†1 神戸大学
Kobe University
†2 PwC サイバーサービス合同会社
PwC Cyber Services LLC
†3 岐阜大学
Gifu University

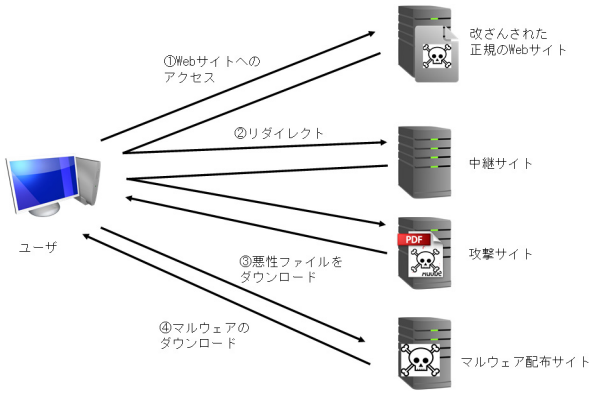

図 1：Drive-by-Download 攻撃の概要

ると想定し，リンク構造に基づく悪性 Web サイト識別手法について検討する．

望月ら[4]はリンク情報に着目した Web 改ざん検知支援システムを提案している．これは前回訪問時の構成情報と今回閲覧時の構成情報を比較することで改ざんを検知するものである．つまり，Web 閲覧者が Web ページを繰り返し訪問することを前提としており，未知の悪性 Web サイトを検知することは考えられていない．提案手法では Web サイトの良性・悪性の判定の際に，ニューラルネットワークを用いる．ニューラルネットワークは機械学習で扱われる計算アルゴリズムの 1 つであり，入力されたデータに対して自律的に学習を行う点が特徴である．提案手法では，良性及び悪性が既知である Web サイトのリンク構造データを学習用データとして，良性及び悪性の 2 つのパターンをニューラルネットワークに学習させる．学習完了したニューラルネットワークを用いて，学習用データに含まれていない Web サイトのリンク構造データに対して悪性判定を行う．つまり，悪性 Web サイトのリンク構造間である程度の類似性が存在すれば，提案手法によって未知の悪性 Web サイトを検出することが可能となる．

本論文は以下のように構成されている．まず第 2 章では，悪性 Web サイトの攻撃手法について説明する．第 3 章ではニューラルネットワークについて説明し，第 4 章で提案手法のアルゴリズム，第 5 章では提案手法の性能評価のために行った実験とその結果を報告する．第 6 章にまとめと今後の課題を述べる．

2. 悪性 Web サイトにおける攻撃

この章では，悪性 Web サイトにおける攻撃のうち，Web サイトにアクセスした不特定多数のユーザに対して攻撃を行う Drive-by-Download 攻撃(DbD 攻撃)とマルバタイズについて説明する．

2.1 Drive-by-Download 攻撃

まず，DbD 攻撃では図 1 に示す様な攻撃環境を構築する．この DbD 攻撃は入り口サイト，中継サイト，攻撃サイト，マルウェア配布サイトといった複数のサイトによって構成されている．多くの場合，入り口サイト以外は Exploit Kit によって構築される．入り口サイトはユーザのアクセスを中継サイトに不正リダイレクトする．この入り口サイトは，正規の Web サイトを改ざんするなどして構築される．中継サイトは攻撃サイトへアクセスをリダイレクトする．このとき，攻撃検出を回避または困難にするために，リダイレクトは複数用意されている場合が見られる．攻撃サイトでは，ユーザの OS や Web ブラウザ，Adobe Flash Player といったアプリケーションの脆弱性を突く攻撃コードをダウンロードさせる．そのコードがユーザをマルウェア配布サイトにアクセスさせ，自動的にマルウェアがダウンロード及び実行されることでユーザマシンが感染する．DbD 攻撃では，その振る舞いがリンク構造の特徴となる可能性が高いと考えられる．

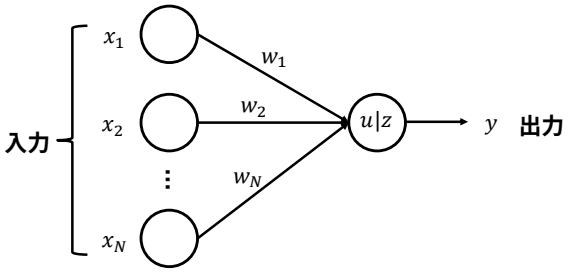

図 2：1 ユニットのパーセプトロン

2.2 マルバタイズ

マルバタイズは Web 広告に攻撃コードを埋め込み，ネットワーク広告を利用してマルウェアを配信する攻撃手法である．正規のサイトに不正広告が表示されることで，ユーザはマルウェア配布サイトにリダイレクトされ，脆弱性を攻撃されてマルウェアに感染する．マルバタイズにおいても，DbD 攻撃と同様の振る舞いをするため，リンク構造に特徴が表れると考えられる．DbD 攻撃では改ざんされた Web サイトにアクセスすると攻撃を受けるのに対し，マルバタイズではユーザが意図せず閲覧したネットワーク広告を介して攻撃を受ける．また，ネットワーク広告は広告配信会社が複数の広告掲載媒体に対して広告を配信するアドネットワークを利用して複数のサイトに配信される．このため，マルバタイズは DbD 攻撃より広範囲に影響及ぼす可能性がある．

3. ニューラルネットワーク

近年，画像認識や音声認識などの分野でニューラルネットワークを用いたディープラーニング[5]が注目されている．ディープラーニングとは，多層構造型ニューラルネットワークを用いた機械学習法の一つである．この章では，ディープラーニングの基礎となるパーセプトロン及びニューラルネットワークと同義である多層パーセプトロンについて説明する．

3.1 パーセプトロン

パーセプトロンは長さNの入力ベクトルに対して一つの値を出力する.図2に1ユニットのパーセプトロンを示す.入力ベクトルを$\mathbf{x} = [x_1, \cdots, x_N]^{\mathrm{T}}$，重みベクトルを$\mathbf{w} = [w_1, \cdots, w_N]^{\mathrm{T}}$とする．加重和ユニット出力$u$は，バイアス$b$を加味して

$$u = b + \sum_{n=1}^{N} w_n x_n = b + \mathbf{w}^{\mathrm{T}}\mathbf{x} \tag{1}$$

と表される．パーセプトロンの出力yは，活性化関数出力$z = \varphi(u)$と閾値tに対し

$$y = \begin{cases} 0 & z \leq t \\ 1 & z > t \end{cases} \tag{2}$$

と表される．活性化関数は，例えば，ロジスティック関数

$$\varphi(u) = 1/(1 + \exp(-u)) \tag{3}$$

や ReLU (Rectified Liner Unit)

$$\varphi(u) = \max(u, 0) \tag{4}$$

が用いられる.

3.2 多層パーセプトロン

3.2.1 構成

ニューラルネットワークとは多数のパーセプトロンを組み合わせたネットワークモデルである．図3にL層のニューラルネットワークを示す．ニューラルネットワークは入力層，隠れ層，出力層の3種類の層から構成される．図3において，第1層が入力層にあたり，第L層が出力層にあたる．第2から$L-1$層は隠れ層である．このような複数層からなるネットワークを多層パーセプトロン(multilayer perceptrons)またはMLPsと呼ぶ．第l層はN_lユニットのパーセプトロンで構成される．但し，l番目の隠れ層の入力は，第$l-1$層の活性化関数出力である．

3.2.2 ニューラルネットワークによる分類法

入力ベクトル$\mathbf{x}$に対するニューラルネットワークを用いた分類法を概説する．第l層のユニットへの入力と出力をそれぞれ

$$\mathbf{u}^{(l)} = \left[u_1^{(l)}, u_2^{(l)}, \ldots, u_{N_l}^{(l)}\right]^{\mathrm{T}}, \tag{5}$$

$$\mathbf{z}^{(l)} = \left[z_1^{(l)}, z_2^{(l)}, \ldots, z_{N_l}^{(l)}\right]^{\mathrm{T}} \tag{6}$$

と表す．但し，入力層($l = 1$)では，

$$\mathbf{u}^{(1)} = \mathbf{z}^{(1)} = \mathbf{x} \tag{7}$$

とする．また，第l層($l \geq 2$)での入力及び出力ベクトルはそれぞれ

$$\mathbf{u}^{(l)} = \mathbf{W}^{(l)}\mathbf{z}^{(l-1)} + \mathbf{b}^{(l)}, \tag{8}$$

$$\mathbf{z}^{(l)} = \varphi^{(l)}(\mathbf{u}^{(l)}) \tag{9}$$

と表される．ここで，第l層における活性化関数のベクトル出力は$\varphi^{(l)}(\mathbf{v}) = \left[\varphi^{(l)}(v_1), \cdots, \varphi^{(l)}\left(v_{N_l}\right)\right]^{\mathrm{T}}$であり,また，$\mathbf{b}^{(l)}$は第$l$層のバイアスベクトルを表す．更に，$\mathbf{w}_k^{(l)}$を第$l-1$層と

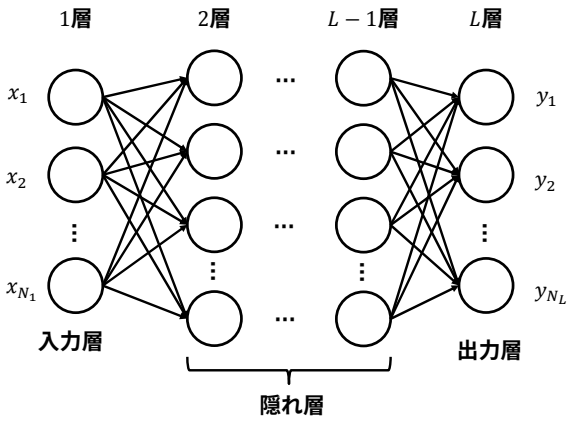

図3：多層パーセプトロンの例

第l層のk番目のノード間の重みベクトルとして，第l層の重み行列は$\mathbf{W}^{(l)} = \left[\mathbf{w}_1^{(l)}, \cdots, \mathbf{w}_{N_l}^{(l)}\right]^{\mathrm{T}} \in \mathbb{R}^{N_l \times N_{l-1}}$である．出力層$L$での$n$番目のユニット出力$y_n$は，閾値判定の他,ソフトマックス関数を用いて

$$y_n = \frac{\exp(u_n^{(L)})}{\sum_{j=1}^{N_L} \exp(u_j^{(L)})} \tag{10}$$

と定めてもよい．最後に，ベクトル$\mathbf{y} = [y_1, \cdots, y_{N_L}]$が入力ベクトル$\mathbf{x}$の分類結果となる．

4. 提案手法

この章では，ニューラルネットワークを用いた悪性Webサイト識別手法のアルゴリズムについて説明する．提案手法のアルゴリズムを図4に示す.

4.1 リンク構造解析

悪性 Web サイトにより構築された入り口サイトからマルウェア配布サイトまでの一連のリダイレクトを把握することは容易ではない．そこで，収集した通信データからアクセスした Web サイトの URL の接続関係を明らかにするためリンク構造解析を行う．リンク構造解析によって得られた結果は，ノードとエッジからなる有向グラフとなる．

4.2 入力データセットの作成

以下にニューラルネットワークに入力するデータセットの作成手順を示す.

(1) リンク構造解析によって得られたk番目の有向グラフを$M \times M$の隣接行列$\mathbf{X}_k$に変換する．
(2) 隣接行列$\mathbf{X}_k$をベクトル化し,長さM^2の入力ベクトル$\mathbf{x}_k$を得る．
(3) リンク情報が良性 Web サイトのものであれば 0，悪性 Web サイトのものであれば 1 と学習用データにラベル付けする．

4.3 良性/悪性 Web サイトのリンク構造の学習

4.2 の行程で作成された，良性・悪性が既知である学習

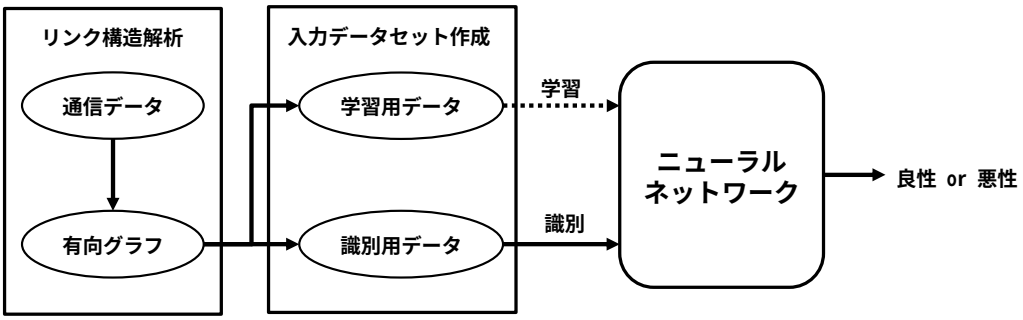

図 4：提案手法のアルゴリズム

表 1：リンク構造解析（Web クローラー）の環境

仮想マシン	VMWare Workstation 12 Player
OS	Windows7 Home Premium
Web ブラウザ	Internet Explorer 11
プラグイン	Java6 Update19
	Silverlight5.0.61118
	Adobe Flash Player15.0.0.108
収集期間	2016/07/27 ～ 2016/07/29 (悪性サイト) 2016/07/29 ～ 2016/07/30 (良性サイト)

用データセットをニューラルネットワークに入力し学習させる. 即ち, K個の学習データセットに対する交差エントロピーによる学習誤差

$$E(\Omega) = -\sum_{k=1}^{K}\sum_{n=1}^{N_L} d_{kn}\log y_n(\mathbf{x}_k;\Omega) \qquad (11)$$

が最小になるように重み行列の集合 $\Omega = \{\mathbf{W}^{(l)} \mid l = 2,\cdots,L\}$を決定する. ただし, k番目の学習データのラベルは$\mathbf{d}_k = [d_{k1},\cdots,d_{kN_L}]$とし, $y_n(\mathbf{x}_k;\Omega)$は入力ベクトル$\mathbf{x}_k$と重み集合Ωに対するn番目のユニット出力(式(10))を記す.

4.4 良性・悪性の識別

良性・悪性が未知であるリンク情報を 4.2 の(1)(2)の行程によって要素がM^2のベクトルに変換する. 入力データを学習済みの重み集合Ωで決定されたニューラルネットワークに入力することで, 識別結果が出力される.

5. 評価実験

提案手法であるニューラルネットワークを用いたリンク構造に基づく悪性 Web サイト識別手法の性能を評価する.

5.1 データセット

ブラックリスト Malware Domain List[6]に掲載されている悪性 Web サイトを実際に解析し, その結果から悪性 Web サイトのリンク構造の有効グラフを取得する. Malware Domain List に掲載されているサイトの内 Exploit Kit に関する URL に表 1 で示す解析環境でアクセスし, Wireshark[7]で通信データを収集した. 収集した通信データに対してリンク構造解析を行って得られた有向グラフ 138 個を悪性サイトのデータセットとする.

表 2：ニューラルネットワークの実装環境

OS	CentOS7
GPU	NVIDIA GeForce GTX 780 Ti
CPU	Core i7 4771 3.5GHz
Deep Learning Framework	Chainer ver. 1.12

表 3：ニューラルネットワークの構成

パラメータ	定義	値
L	層数	3
U_i	入力層のユニット数	78400
U_h	隠れ層のユニット数	1000
U_o	出力層のユニット数	2

良性 Web サイトについては Web サイト価値ランキング[8]に掲載されている企業サイトに対して同様の解析を行った. 得られた有向グラフのうち 97 個のデータを良性データセットとする.

5.2 ニューラルネットワークの構築

表 2 の実装環境に Deep Learning のライブラリである Chainer[9]を用いて評価実験用のニューラルネットワークを構築した. 提案アルゴリズム内で用いるニューラルネットワークの構成パラメータを表 3 に示す. 活性化関数は ReLU(式(4))を用い, 誤差関数はソフトマックス関数(式(10))の交差エントロピー関数(式(11))を用いた.

5.3 評価方法

5.1 で収集した実際の Web サイトのリンク構造のデータ$K = 235$個に対して提案手法を適用する. 収集したデータを有効活用するため, ある 1 つのデータX_{k_0}を試験データとし, 残りの$K-1$個のデータ$\{\mathrm{X}_k \mid 1 \le k \le K, k \ne k_0\}$を学習データとする. この学習データを 5.2 のニューラルネットワークに入力し, エポック数eで学習する. その後, 学習済みのニューラルネットワークに試験データX_{k_0}を入力し, 良性・悪性の判定を行う. 上記の実験を, 試験データを変更し, 合計でK回試行する. 正解率は,

(判定結果が正解であった回数) / K

表 4：判定結果の正解率($e = 10$)

	良性 Web サイト	悪性 Web サイト	全体
正解率	0.62	0.82	0.74

と定める．

5.4 結果

表 4 にエポック数$e = 10$に対する良性・悪性の判定結果の正解率を示す．表 4 より，良性 Web サイトにおける正解率より悪性 Web サイトにおける正解率の方が高いことが判る．これは，悪性 Web サイトのリンク構造は互いに類似性を有する可能性が高いという想定を裏付ける．一方，良性の Web サイトは互いに類似性が少ないため，判定成功率が低くなったと考えられる．

入力データに注目する．良性及び悪性サイトそれぞれのリンク構造の隣接行列を平均し，グレースケールで表現したものを図 5，図 6 に示す．図 5 の画像は全体的に灰がかっていることがわかる．このことから，良性サイトでは，サイト毎に異なるリンク構造を持つことを表し，リンク構造間に類似性が少ないことがわかる．一方で，図 6 の画像では，左上部が黒くなっており，その他の部分はほぼ白いことがわかる．このことから，悪性サイトのリンク構造は互いに類似していることがわかる．以上の良性・悪性サイトの入力データの差が学習・識別に影響し，今回の検証結果が得られたと考えられる．

6. まとめ

本稿では，良性及び悪性 Web サイトのリンク構造には互いに類似性が存在すると想定し，ニューラルネットワークを用いたリンク構造に基づく悪性 Web サイト検知手法について検討した．提案手法の性能の評価を行うため実際の Web サイトのリンク構造を用いた実験を行った．結果より，悪性 Web サイトのリンク構造には互いに類似性がある可能性があることを確認した．

Web ブラウザとプラグインの組み合わせが変われば，収集されるリンク構造データが異なり[10]，提案手法の判定正解率も変化すると考えられる．また，今回の実験では最も単純な 3 層のニューラルネットワークでのみで評価を行ったが，隠れ層の数やユニット数を変化させることでも判定正解率が向上することが考えられる．そこで今後の課題として，(a)様々なアクセス環境でリンク情報を収集し，アクセス環境ごとの性能の評価と(b)ニューラルネットワークの構成を変化させた場合の性能の評価を行う．

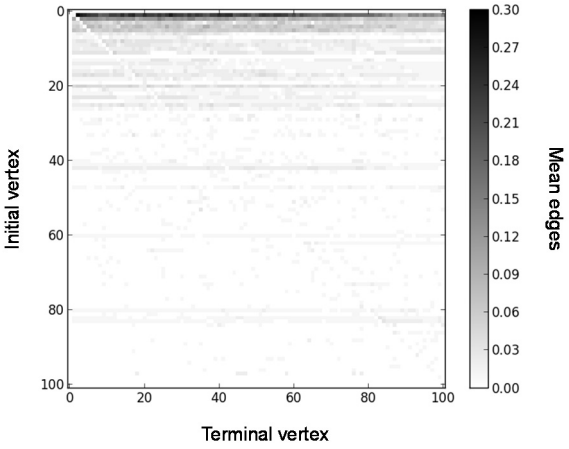

図 5：リンク構造の隣接行列の平均(良性サイト)

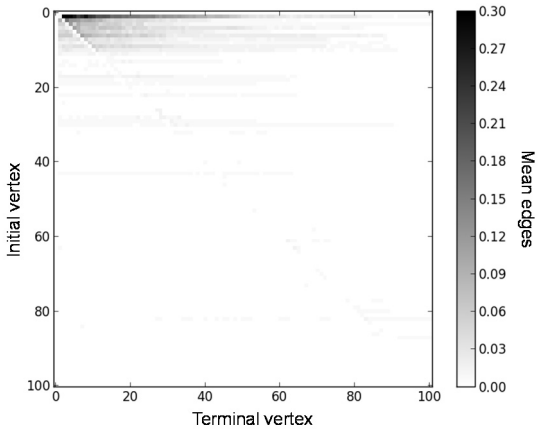

図 6：リンク構造の隣接行列の平均(悪性サイト)

参考文献

[1] 芹田進, 藤井康広, 角田朋, 吉竹利織, 大鳥朋哉, 木城武康, 寺田真敏, “URL 正規表現自動生成による悪性通信検知手法に関する一考察,” CSS, pp. 242-249, 2015.

[2] 佐藤祐磨, 中村嘉隆, 高橋修, “エクスプロイトキットで利用される文字列特徴を用いた悪性 URL 検出手法の提案,” IPSJ SIG Technical Report, pp. 1-6, 2016.

[3] 今野由也, 角田裕, “Exploit Kit で作成された悪性コンテンツの類似性調査,” CSS, pp. 1251-1257, 2015.

[4] 望月翔太, 高田哲司, “リンク情報の時間変化に着目した Web 改ざん検知支援システムの提案,” CSS, pp. 489-496, 2014.

[5] 岡谷貴之, 深層学習, 講談社サイエンティフィック.

[6] Malware Domain List, https://www.malwaredomainlist.com/mdl.php. (参照 2016.8.12)

[7] Riverbed Technology, Wireshark, https://www.wireshark.org/. (参照 2016.8.12)

[8] トライベック社, “Web サイト価値ランキング 2015 業種別ランキング,” http://japanbrand.jp/ranking/we-ranking/we2015-2.html. (参照 2016.8.12)

[9] Chainer, http://chainer.org/. (参照 2016.8.12)

[10] 永井達也, 神薗雅紀, 白石善明, 毛利公美, 星澤祐二, 森井昌克, “マルチ環境での Drive-by-Download 攻撃のリンク構造解析について,” ICSS2016-12, pp. 63 - 68, 2016.