情報処理学会論文誌 Vol.62 No.5 1341–1349 (May 2021)

DGAマルウェアにより自動生成された悪性ドメインの判別

佐藤 彰洋[1,a)] 林 豊洋[1,b)] 和田 数字郎[1,c)] 福田 豊[1,d)]

受付日 2020年1月27日, 採録日 2021年2月2日

概要：マルウェアはインターネットにおける重大な脅威の1つである．多くのマルウェアには，検出を回避するための機能としてDGA（Domain Generation Algorithm）が実装されている．DGAとは，C&C（Command-and-Control Server）のドメインを頻繁に変更することで，マルウェアからC&Cへ向けた通信であるコールバックを隠蔽するための仕組みである．本稿では，表層的なドメイン文字列の解析により，DGAマルウェアのコールバックのために自動生成した悪性ドメインの判別を試みる．本手法の独自性は，DGAに関する事前情報をまったく必要とせず，ドメイン文字列の意味の有無からドメインの良性と悪性を推定する点にある．また実験を通じて，提案手法が0.9960の再現率と0.9029の適合率で悪性ドメインを判別可能であることを確認した．この結果から，ネットワークに内在するマルウェアへの迅速な対処が可能となるため，ネットワークにおける安全性の向上が期待できる．

キーワード：マルウェア，C&C，ドメイン生成アルゴリズム，ドメイン名，ネットワークセキュリティ

An Approach for Identifying Malicious Domain Names Automatically Generated by DGA Malware

Akihiro Satoh[1,a)] Toyohiro Hayashi[1,b)] Sujiro Wada[1,c)] Yutaka Fukuda[1,d)]

Received: January 27, 2020, Accepted: February 2, 2021

Abstract: Some of the most serious security threats facing computer networks involve malware. Many types of malware have DGAs (Domain Generation Algorithms) to avoid detection. A DGA is a mechanism for hiding the callback communications of malware by frequently changing the domain name of a C&C (Command-and-Control Server). In this paper, we attempt to detect the callback communications of DGA malware by superficially analyzing domain names. Our approach distinguishes between benign and malicious domains based on the meaning of their character string and does not require any prior knowledge about the DGAs. Our evaluation indicates high performance, with a recall of 0.9960 and a precision of 0.9029. By enabling one to swiftly address various malware, our approach contributes to dramatically improving network security.

Keywords: malware, C&C, domain generation algorithm, domain name, network security

1. はじめに

マルウェアはインターネットにおける重大な脅威の1つである．サイバー犯罪者は，C&C（Command-and-Control Server）を介してマルウェアに感染した端末を操作することで，機密情報窃取，フィッシング詐欺，標的型攻撃などの悪意ある活動を試みる．米McAfee社の報告によると，約30万のマルウェアが日々誕生しており，それによる世界の総損失額は年間6,000億ドルを超える[1]．そのため，マルウェアに対抗するための技術の確立が急務である．

マルウェアによる被害の抑止のため，管理者は自身のネットワークに内在する感染端末を迅速に排除することが

[1] 九州工業大学
Kyushu Institute of Technology, Kitakyushu, Fukuoka 804–8550, Japan
a) satoh@isc.kyutech.ac.jp
b) toyohiro@isc.kyutech.ac.jp
c) swada@isc.kyutech.ac.jp
d) fukuda@isc.kyutech.ac.jp

求められる．一方，多くのマルウェアには，検出を回避するための機能としてDGA（Domain Generation Algorithm）が実装されている [2]．DGAとは，C&Cのドメインを頻繁に変更することで，マルウェアからC&Cへ向けた通信であるコールバックを隠蔽するための仕組みである．具体的には，マルウェアはDGAに基づいて多数のドメインを自動生成した後，それらドメインの名前解決を試みる．その名前解決の結果，正しい応答を返したドメインをC&Cのものと見なし，そのドメインとの間で通信を確立する．ここで留意すべきは，コールバックのために生成されるドメインの生存時間が極端に短い点である．ゆえに，従来のブラックリストを用いた通信の監視では，DGAマルウェアのコールバックを検出することが困難となる [3]．

マルウェアとC&Cとの通信を補足するために，パケットのペイロードを参照するDPI（Deep Packet Inspection）が用いられてきた [4], [5]．一方，2017年の時点でインターネットにおける暗号化通信の割合は50%を超えること，それとあわせて約70%のマルウェアが通信を暗号化することが確認されている [6]．このように暗号化は情報保護の一般的な手段であり，それが占める割合に反比例してDPIを適用可能な通信はごくわずかなもののみとなっている．

本稿では，DNS（Domain Name System）に対する膨大な数の名前解決から，DGAマルウェアのコールバックのために自動生成したドメインの判別を試みる．DNSに着目した理由は，マルウェアによる通信に先んじて必ず名前解決が生じること，その名前解決は暗号化による通信内容の隠蔽が困難であることに起因する．DGAマルウェアにより生成された悪性ドメインは生存時間が極端に短いため，その判別に利用可能な特徴が限定的である．その問題をふまえ，我々は表層的な文字列解析に基づく悪性ドメイン判別手法を提案する．本手法の独自性は，DGAに関する事前情報をまったく必要とせず，ドメイン文字列の意味の有無からドメインの良性と悪性を推定する点にある．これは，人為的に生成された良性ドメインが，組織や商品の名称など，その意図を反映した文字列をなすこと，自動的に生成された悪性ドメインが，登録済みのドメインとの衝突を避けるため無意味な文字列をなすことから，良性ドメインと悪性ドメインの文字列において明確な差異が現れるがゆえである．また実験を通じて，提案手法が0.9960の再現率と0.9029の適合率で悪性ドメインを判別可能であることを確認した．すなわち，悪性ドメインの名前解決を予兆としてDGAマルウェアを高精度で検出できることを示した．この結果から，ネットワークに内在するマルウェアへの迅速な対処が可能となるため，ネットワークにおける安全性の向上が期待できる．

本稿の構成は次のとおりである．まず，2章で既存研究とその問題点を整理する．3章で文字列解析の基づく悪性ドメイン判別手法を提案した後，4章で提案手法の有効性を議論する．最後に5章で本研究の貢献と課題をまとめる．

2. 関連技術

本章では，DGAマルウェアを中心とした関連技術について述べる．2.1節でDGAマルウェアの詳細について説明した後，2.2節で既存研究とその問題点を整理する．

2.1 DGAマルウェア

ConfickerやKrakenなど，世界で深刻な被害をもたらしたマルウェアには，その機能の一部としてDGAが実装されている．また，それらのマルウェアを広範囲に拡散するため，ウェブページやウェブ広告への不正コードの埋め込みが観測されている [7], [8]．

図1にDGAマルウェアによるコールバックの概要を示す．ここで，図中のQで示す通信はマルウェアからRDNS（Recursive DNS Server）に対する名前解決を，Rはその応答を意味する．また，C&Cのものとして，DGAにより生成されたドメインが事前に登録されているものとする．まず，マルウェアはDGAに基づいて複数のドメインを自動生成し，それらドメインを自身の属するネットワーク内のRDNSに問い合わせる．RDNSは，ドメインが登録済みであった場合，そのドメインに対応付けられたアドレスを，ドメインが未登録であった場合，エラーメッセージとしてNXDOMAIN（Non-Existing Domain）を応答する．最終的に，マルウェアは正しい応答があったドメインをC&Cのものと見なし，そのドメインに対してコールバックを試みる．

DGAの目的は，マルウェアとC&Cの間に可用性の高い通信経路を確立することにある．具体的には，C&Cのドメインを変更することで，ブラックリストに基づく通信の遮断を容易に回避することが可能となる．加えて，ネッ

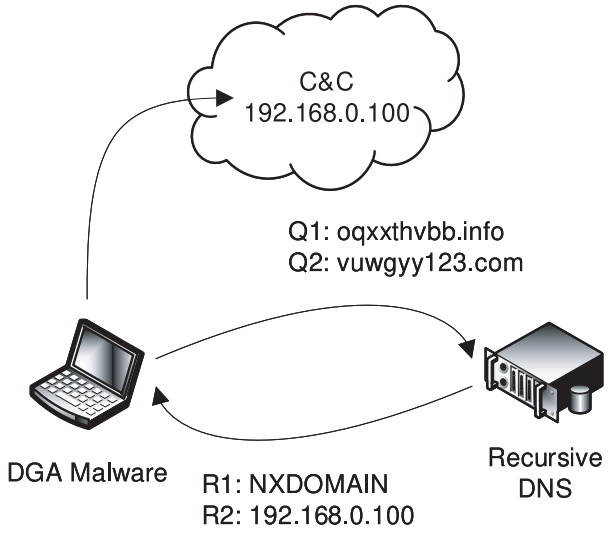


図1 DGAマルウェアとC&Cとのコールバック通信
Fig. 1 Callback communication between DGA malware and C&C.

トワーク内から外へ向けた通信は宛先が多岐にわたるためコールバックの発見が困難となること，アドレス変換やファイアウォールにより通信を制限されないことがあげられる．ここで留意すべきは，マルウェアと C&C とで同一 DGA を用いることで，ドメインの変更にいっさいの情報交換を必要としない点である．

2.2 既存研究と問題点

ブラックリストの高度化については頻繁な研究が行われており，現在もネットワークにおける脅威防御戦略の中核をなしている．Soldo らは，複数の参加者から提供される過去の攻撃ログに基づいて，新たにブラックリストを生成する方法を提案している [9]．また，Freudiger らは，P2P の技術を応用することで，機密性を担保した攻撃ログの共有を実現している [10]．それに対して，DGA マルウェアは，C&C のドメインを頻繁に変更することにより，ブラックリストに基づく通信の遮断を回避する機能を有している．

Gu らは，DPI に基づく受動的なネットワーク監視システムとして BotHunter を実装した [11]．BotHunter は，マルウェアの一般的な挙動をモデル化して，それと関連の強い通信を感染の根拠とする．また，DPI の性能改善に向けた取り組みなどが報告されている [12], [13]．しかしながら，ネットワークにおける暗号化の担う役割に反比例して，DPI を適用可能な通信はごくわずかなもののみとなっている．ゆえに，マルウェアの検出のための情報源として，暗号化の影響を受けない DNS の名前解決が注目されている．

Rahbarinia らは，DNS の名前解決において既知の悪性ドメインと高確率で共起するドメインから未知の悪性ドメインを発見する Segugio を開発した [14]．Segugio は次の直感的知見，(1) 同一マルウェアファミリに感染した端末は，同一悪性ドメイン群と通信する傾向にあること，(2) 未感染の端末は，悪性ドメインと通信することがないことに基づいている．一方，DGA マルウェアにおいては，コールバック通信に生存時間が極端に短い一時的な悪性ドメインを用いるため，その一時的な悪性ドメインと共起するドメインは存在しえない．ゆえに，このシステムは DGA マルウェアの通信に対して効果をなしえない．

Bilge らは，DNS の名前解決とその応答から計測可能な特徴量と機械学習を用いてドメインを評価する Exposure を開発した [15]．特徴量の例は，ドメインの生存期間，ドメインに割り当てられたアドレスの数，ドメインの文字長などである．このシステムが採用する教師あり機械学習において，その精度は一般的に学習用データセットの数と質に依存する．しかしながら，DGA マルウェアにおける多くの悪性ドメインは NXDOMAIN を応答するため特徴量が得られないこと，C&C に対応付けられた悪性ドメインは生存時間が極端に短いことから，十分な量の学習用データセットを確保することが困難となる．

Berger らは，名前解決におけるアドレスとドメインの関係の変化を継続的に学習する DNSMap を構築した [16]．DNSMap は，C&C のアドレスが複数のドメインに，そのドメインが複数のアドレスに対応付けられること，それらの対応関係が時間経過にともない急速な変化を示すことに着眼している．一方，Wang らは，名前解決の挙動と分布特性に基づいて DGA マルウェアの検出する DBod を実装した [17]．その検出は，同一 DGA により生成された候補ドメイン群に対して接続を試みるため，同一マルウェアファミリに感染した端末による名前解決が特定の期間中に高い類似性を示すことに基づいている．これらのシステムは広範囲にわたる DNS トラフィックの観測を必要とするため，その適用は ISP などの大規模なネットワークに限定される．

これまでに，マルウェアが生成するドメインの解析結果や [18]，その解析を自動化する仕組みなどが報告されている [19]．その結果をふまえ，いくつかの研究では文字列の特徴のみを用いたドメインの判別が試みられている．Truong らは，ドメイン文字列のみから良性と悪性を判別する手法を提案した [20]．この手法は，教師あり機械学習とバイグラムモデリングによりドメインにおける頻出文字パターンを学習する．Anderson らは，深層学習を用いた文字レベルのモデリングにより，その手法を拡張した [21]．また，Vinayakumar らにより，多様な機械学習と深層学習を用いた判別精度の比較結果が示されている [22]．これらの手法は，悪性ドメインを生成するためのルールに識別可能な偏りが存在することに基づいている．しかしながら，その識別には事前の学習を必要とするため，未知の DGA マルウェアに対する性能の低下が懸念される．

3. 提案

本稿では，DNS に対する膨大な数の名前解決から，DGA マルウェアにより自動生成されたドメインの判別を試みる．**表 1** に，DGA マルウェアにより生成されたドメインの例を示す．2.2 節で述べたとおり，同一マルウェアファ

表 1 DGA マルウェアにより生成されたドメインの例
Table 1 Examples of domain names generated by DGA malware.

ChinAd	xe0d7fazyrvvwl9f.ru
	7qvdqaw561dtasyi.com
Conficker	xjjjvqpoh.com.ai
	pfnnwjoeuee.biz
Locky	btlwubflhfllshn.info
	jlbroeji.biz
NewGOZ	1ygx14u1vnf8hb1twhv8619h8ygr.net
	cipu0wdgsnq9u8st8m1lym0hq.com
Nymaim	embonxn.info
	ghhimbgbrpx.biz

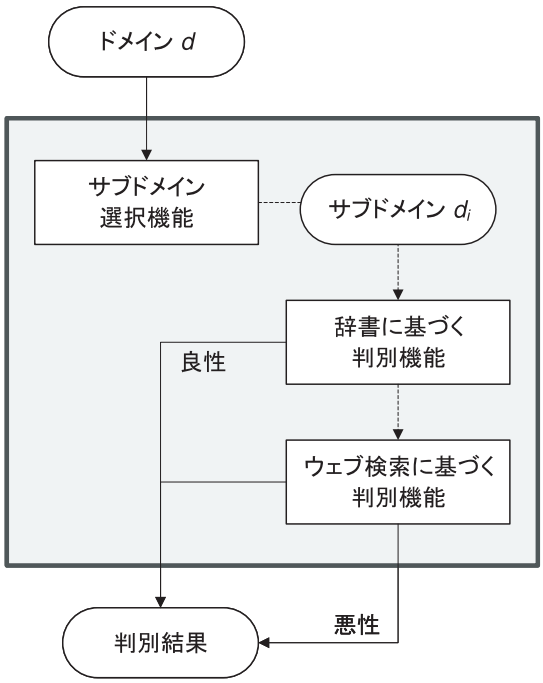


図 2 表層的な文字列解析に基づく悪性ドメイン判別手法の概要
Fig. 2 Overview of the proposed approach for identifying malicious domain names based on superficial analysis of their character strings.

ミリにおける名前解決の類似性を利用するためには，広範囲にわたる DNS トラフィックの観測が必要となる．それに加え，DGA マルウェアにおける悪性ドメインは生存時間が極端に短いため，その判別に利用可能な特徴が限定的である．その問題をふまえ，我々は表層的な文字列解析に基づく悪性ドメイン判別手法を提案する．その理由は，(1) 人為的に生成された良性ドメインが，組織や商品の名称など，その意図を反映した文字列をなすこと，(2) 自動的に生成された悪性ドメインが，登録済みのドメインとの衝突を避けるため意味のないランダムな文字列をなすこと，(3) それゆえに，良性ドメインと悪性ドメインの文字列において明確な差異が現れることに起因する．マルウェアの検出を容易にするためのランダム化の概念は新しいものではなく，この概念に基づくいくつかの研究が発表されている [20], [21], [22], [23], [24]．たとえば，Lin らはサイバー犯罪者へのトレースバックのためにマルウェアの通信を解読する手法を [23]，Wahab らはマルウェアに侵害された仮想マシンを検出するための手法を提案している [24]．また，ドメイン文字列の特徴のみを用いた良性と悪性の判別が試みられている [20], [21]．なお，文献 [20] と [21] の両手法については，提案手法との類似性から 4 章で定量的な判別性能の比較を実施した．ここで特筆すべきは，本手法は DGA に関する事前情報をまったく必要とせず，ドメイン文字列の意味の有無からドメインの良性と悪性を推定する点にある．

図 2 に提案手法の概要を示す．本手法は 3 つの機能，(1) サブドメイン選択機能，(2) 辞書に基づく判別機能，(3) ウェブ検索に基づく判別機能により構成される．以降の節で，各機能の詳細について述べる．

3.1 サブドメイン選択機能

本機能は，効率的な判別のため，ドメイン d から文字長が最大のサブドメイン d_i を選択する．ここで，d_i はドメイン d を構成する i レベルのサブドメインを意味する．たとえば，ドメイン d が `xjjjvqpoh.com.ai` である場合，文字長が最大のサブドメインは d_3 の `xjjjvqpoh` となる．これは，短い文字列のドメインは正当な組織に占有されている可能性が高いこと，登録済みのドメインとの衝突を避ける必要があることから，DGA はドメインとして比較的長い文字列を生成することに起因する．

3.2 辞書に基づく判別機能

本機能は，まず辞書 $\mathbb{D}$ を参照することにより，サブドメイン d_i の文字列を単語群 w に分割する．次いで，その単語群 w の特徴からサブドメイン d_i のランダム性を推定する．その結果，サブドメイン d_i が意味のある文字列であると判断された場合，それを含むドメイン d を人為的に生成された良性ドメインと見なす．

ドメイン文字列の単語分割において参照するため，6 種類の辞書を準備した．その 1 つは，ウェブクローリングで作成したコーパスと英語辞書からなる．ほかの 5 つは，上述の辞書にフランス語，ドイツ語，スペイン語，ロシア語，および日本語の辞書を追加したものである．それら各言語の辞書は，ドメインの表記を考慮して非アルファベットをアルファベットで置換している．その置換では，発音記号を取り除き，たとえば á を a で，î を i で，ü を u で，ç を c で表記した．また，ドイツ語の ß を ss で，スペイン語の ñ を n と nn で，日本語はヘボン式ローマ字で，ロシア語のキリル文字は文献 [25] の規則に基づいて置き換えた．なお，スペイン語の ñ のように，一般的に複数の代用表記が用いられる場合は，それら規則を 1 単語に対して適用している．具体的には，スペイン語 año の場合，ano と anno の 2 単語が辞書に登録されることになる．

式 (1) により，(1) 各単語の文字長が最大，かつ単語数が最小になること，(2) 極端な選択率の差により辞書に含まれる単語を優先することの，2 つの観点に基づいてサブドメイン d_i の文字列を単語群 w に分割する．

$$\mathcal{F}(d_i) = \arg\max_{w \in \mathbb{W}(d_i)} \frac{1}{n} \prod_{j=1}^{n} \mathcal{P}(w_j)$$
$$\mathcal{P}(w_j) = \begin{cases} 1 & (w_j \in \mathbb{D}) \\ 1/|\mathbb{D}|^{|w_j|} & (w_j \notin \mathbb{D}) \end{cases} \quad (1)$$

ここで，$\mathbb{W}(d_i)$ はサブドメイン d_i の文字列における全分割候補の集合，w は単語 $w_1, \cdots, w_j, \cdots, w_n$ からなる候補の

単語群，$|w_j|$ は単語 w_j の文字長，$|\mathbb{D}|$ は辞書 $\mathbb{D}$ の総単語数をそれぞれ意味する．また，$\mathcal{P}(w_j)$ は，単語 w_j が辞書 $\mathbb{D}$ に含まれるか否かに基づいて，単語 w_j の選択率を導出する関数である．この処理において，6 種類の辞書を参照した単語分割から最良の結果を選択する．その結果，$n = 1$，すなわちサブドメイン d_i 自体が辞書 $\mathbb{D}$ に含まれる場合，ドメイン d を良性ドメインと判断する．

サブドメイン d_i が人為的に生成された文字列である場合，単語群 w において各単語の文字長が長いこと，かつ単語数が少ないことが特徴としてあげられる．その特徴をふまえ，式 (2) によりサブドメイン d_i のランダム性を推定する．

$$y_\alpha = \sum_{k=1}^{n} u_k \mathcal{L}_k(w) \tag{2}$$

ここで u_k は，$\mathcal{L}_k(w)$ に重みを付与する係数を，また $\mathcal{L}_k(w)$ は，単語群 w を文字長に基づいて降順に並べ替え，その k 番目の値を導出する関数を意味する．たとえば，単語群 w が `kyutech`，`local`，`domain`，`name` からなる場合，$\mathcal{L}_2(w)$ は `domain` の文字長である 6 を，$\mathcal{L}_4(w)$ は `name` の文字長である 4 を出力する．特筆すべきは，辞書に含まれない単語は，その文字長を 0 と見なす点である．この結果，$y_\alpha > th_\alpha$ を満たす場合，ドメイン d を良性ドメインと判断する．

3.3 ウェブ検索に基づく判別機能

本機能は，辞書の不足を補うため，ウェブ検索の結果を参考にサブドメイン d_i のランダム性を推定する．たとえば，辞書を持たない言語のドメイン，固有名詞を用いたドメイン，頭字語を用いたドメイン，国際化ドメインなどは辞書に基づく文字列解析で良性と悪性を判別できない．

まず，(1) サブドメイン d_i の文字列との完全一致を検索すること，(2) th_β 番目以降の候補を表示することの，2 つの条件を満たすクエリをウェブ検索エンジンに対して発行する．これは，文字列における意味の有無が，それが関連付けられたウェブページの数から間接的に推定可能であることに着眼している．式 (3) にウェブ検索エンジンに対するクエリの例を示す．

$$str ='' d_i''\&num = th_\beta \tag{3}$$

ここで，str は検索対象の文字列を指定する変数であり，その二重引用符により検索結果を文字列 d_i と完全一致するウェブページのみに限定している．また，$num = th_\beta$ は，検索結果の表示を th_β 番目以降のウェブページとすることを意味する．留意すべきは，文字列 d_i が `xn---` で始まる国際化ドメインである場合，それを事前に元の文字列に復号する点である．その結果，$y_\beta > th_\beta$，すなわち文字列 d_i が関連付けられたウェブページの数 y_β が閾値 th_β を超える場合，ドメイン d を人為的に生成された良性ドメインと判断する．それ以外を DGA により自動的に生成された悪性ドメインと見なす．

4. 評価

本章では，実験を通じた提案手法の評価により，DNS に対する膨大な数の名前解決から悪性ドメインを判別できること，その結果に基づくことで DGA マルウェアのコールバックを高精度で検出できることを示す．4.1 節で実験の諸元について述べた後，4.2 節と 4.3 節で結果について議論する．

4.1 諸元

提案手法との比較のため，文献 [20] と [21] を参考にドメインの文字列のみから悪性ドメインを判別する 2 種類の方法を実装した．それらの実装に用いた機器の構成を**表 2** に示す．第 1 の実装は，ドメイン文字列に対するバイグラムモデリングと教師あり機械学習により判別する手法であり，第 2 の実装は，深層学習の 1 つである LSTM（Long Short Term Memory）ネットワークを用いた文字レベルのモデリングにより判別する手法である．ここで留意すべきは，提案手法とは異なり，これらの実装は判別のために良性と悪性ドメインのデータセットによる学習が必須な点である．

表 3 に実験に用いたデータセットを示す．悪性ドメインは，逆行解析を通じて明らかになったマルウェアの DGA により生成したものである [26]．実験に用いたマルウェアは，ChinAd，Conficker，Locky，NewGOZ，Nymaim などの計 19 種である．また，良性ドメインは Alexa [27] が公開するアクセス数の上位 1,000,000 である．2 種類の実装のために，良性ドメインから 5%，全悪性ドメインから 15%を無作為に抽出したものを学習用データセットとした．残りが検証用データセットである．ネットワークにおいて実測されたデータに代わり，これらデータセットを採用した理由は，良性と悪性が正確に付与されたドメインでないと厳密な精度の比較が困難なことに起因する．

提案手法における辞書 $\mathbb{D}$ として，ウェブクローリングで

表 2 実装に用いた機器の構成
Table 2 Specifications of the machine used for implementation.

CPU	Xeon Silver 4110 (8core 2.10 GHz)
GPU	NVIDIA GeForce RTX 2080 Ti (11 GB GDDR6)
RAM	96 GB DDR4-2666
SSD	Seq. Read and Write up to 560 MB/sec and 530 MB/sec
Kernel	Linux 3.10.0-957.1.3.el7.x86_64
Software	TensorFlow 2.0.0, CUDA Toolkit 10.0, cuDNN 7.6.5

表 3 各データセットにおけるドメインの数
Table 3 Numbers of benign and malicious domains in the datasets.

$D0$	$D1$	$D2$	$D3$	$D4$	$D5$	$D6$	$D7$	$D8$	$D9$
Alexa	ChinAd	Conficker	CoreBot	Fobber	Kraken	Locky	NewGOZ	Nymaim	PadCrypt
1,000,000	3,000	3,000	3,000	3,000	3,000	3,000	3,000	3,000	3,000
$D10$	$D11$	$D12$	$D13$	$D14$	$D15$	$D16$	$D17$	$D18$	$D19$
Proslikefan	Pykspa	Qadars	Ramnit	Shiotob	Simda	Symmi	Tempedreve	Tinba	Vawtrak
3,000	3,000	3,000	3,000	3,000	3,000	3,000	3,000	3,000	3,000

表 4 実験結果
Table 4 Experimental results.

	Recall	Precision
Truong et al. [20]	0.7915	0.5366
Anderson et al. [21]	0.8447	0.6462
Our work	0.9960	0.9029

作成したコーパス [28], [29]，および Aspell [30] に登録されている単語を使用した．その総単語数は約 8,000,000 語である．また，各種パラメータを経験的に $u_1 = 2$，$u_2 = 1$，$u_3 = 0.25$，$th_\alpha = 15$，$th_\beta = 50$ に設定した．その y_β は，`bing.com` と `yahoo.com` の 2 つのウェブ検索エンジンによる結果のうち，大きい方の値とした．これらの最適化は今後の課題とする．

4.2 定量的評価

各手法における悪意ドメインの判別性能を定量的に評価するために，一般的な 2 つの指標を用いた．再現率は，悪性ドメインの総数に対する悪性と判別されたドメインの数の比率であり，適合率は，悪性と判別されたドメインの総数に対する真に悪性であるドメインの数の比率である．

実験結果を**表 4** に示す．この結果から，提案手法は 0.9960 の再現率と 0.9029 の適合率を達成しており，2 つの実装よりも高い精度を示すことが見て取れる．2 つの実装の精度が低下した理由は次のとおりである．まず，学習用データの量が不十分であったこと，学習用データに偏りがあったことがあげられる．次いで，単純な文字の並びのみからドメインの良性と悪性を判別することの限界である．提案手法は，データセットを用いた事前の学習が不要であること，ドメインの文字列解析により良性と悪性を判別することから，精度低下の要因を除外できたと考えられる．

各データセットにおいて誤判したドメインの数を**表 5** に示す．提案手法は，14 種類のマルウェアにおける誤判は 5 ドメイン未満であり，非常に高い精度を達成している．しかしながら，Nymaim，Proslikefan，Pykspa，および Vawtrak は誤判は 20 ドメイン程度，Alexa と Conficker に至っては誤数が 5,189 と 102 ドメインとなり，他と比較して精度の低下が目につく結果となった．Conficker が生成するドメインを調査したところ，長さが 4 から 12 までのランダムな文字列であった．これらの中で誤判されたのは，5 文字以下の文字列からなるドメインのみである．すなわち，ドメインの文字長が短い場合，頭字語を用いた良性ドメインと機械的に生成された悪性ドメインを区別できないことが誤判の原因である．その他のマルウェアにおけるドメインの誤判は，長さ 6 前後の文字列が偶然にも意味のある単語をなしたことに起因している．その具体例は，`docket.com`，`wouled.biz`，`olleman.com` である．一方，Alexa における誤判は次の 4 種，(1) 頭字語を用いたドメイン，(2) 非アルファベットをアルファベット表記したドメイン，(3) 数字を多く含むドメイン，(4) ランダムな文字列からなるドメインであった．ここで，(1) のドメインは Conficker と同様の原因，(2) と (3) のドメインは，その辞書を持たないことが原因である．(4) のドメインは，良性と悪性の差が文字列に現れないため，提案手法による判別が仕組み上困難である．その具体例は，`3lqjnuhra3xf585jgt` `khk71exuhu6yrkna.com` や `ctxxgxdnhctxxgxdnh.xyz` である．

提案手法における辞書に基づく判別機能のパラメータ u_k と th_α は，ドメインを良性と判別する基準となる，文字列 d_i に占める単語の割合を決めるものである．本実験を通じて，この機能により Alexa の 541,029 ドメインを正確に良性と判別できたこと，それに対して悪性を良性と誤判したのが Nymaim の 11 ドメインと Pykspa の 7 ドメインのみであったことから，それらパラメータはおおむね適当な値であったといえる．一方，ウェブ検索に基づく判別機能のパラメータ th_β は，ウェブ検索結果に強く依存した値となっている．そのため，判別の精度に強く影響を及ぼすのは，パラメータの微細な設定よりもウェブ検索エンジンの選択であると考えられる．具体的には，Alexa に含まれる `fatosdesconhecidos.com.br` は，`yahoo.com` による検索では th_β の値を下回り悪性と誤判されるが，`yahoo.br` による検索では正確に良性と判別されることを確認している．このことから，特に非アルファベットをアルファベット表記したドメインの場合，その ccTLD（Country Code Top Level Domain）に応じたウェブ検索エンジンの結果を参照することでさらなる精度の向上が期待できる．

以上の議論より，いくつかの課題があるにしても提案手法が 0.9960 の再現率と 0.9029 の適合率で悪性ドメインを

表 5 各データセットにおけるドメインの誤判数
Table 5 Numbers of misidentified domains in the datasets.

	$D0$	$D1$	$D2$	$D3$	$D4$	$D5$	$D6$	$D7$	$D8$	$D9$
Truong et al. [20]	33,108	110	660	98	340	432	509	18	652	282
Anderson et al. [21]	22,407	22	522	34	146	202	274	2	514	222
Our work	5,189	0	102	0	0	2	1	0	25	0
	$D10$	$D11$	$D12$	$D13$	$D14$	$D15$	$D16$	$D17$	$D18$	$D19$
Truong et al. [20]	674	702	278	448	246	638	1,586	632	314	1,480
Anderson et al. [21]	576	580	286	218	150	626	1,272	396	152	1,326
Our work	19	21	0	0	0	3	1	0	0	17

表 6 提案手法と既存手法との定性的な比較
Table 6 Qualitative comparison of our work with other well-known detection methods.

	(1) DGA malware detection	(2) On-line detection	(3) Robust to encryption	(4) Network scale independent	(5) No need for a priori dataset
Soldo et al. [9]	Poor	Good	Good	Good	Poor
Gu et al. [11]	Poor	Good	Poor	Good	Good
Rahbarinia et al. [14]	Poor	Good	Good	Fair	Poor
Bilge et al. [15]	Poor	Fair	Good	Good	Poor
Berger et al. [16]	Good	Good	Good	Poor	Fair
Wang et al. [17]	Good	Good	Good	Poor	Good
Truong et al. [20]	Fair	Good	Good	Good	Poor
Anderson et al. [21]	Fair	Good	Good	Good	Poor
Our work	Good	Poor	Good	Good	Good

判別できることを確認した．この結果は，これら悪性ドメインの名前解決を予兆として，ネットワークに内在する DGA マルウェアを高精度で検出できることを示唆している．

4.3 定性的評価

表 6 に提案手法と既存手法との定性的な比較を示す．その比較の観点は，(1) DGA マルウェアの検出性能，(2) 検出の実時間性，(3) 暗号化に対する頑健性，(4) ネットワークの規模に対する依存の有無，(5) データセットを用いた学習など事前知識の有無である．まず前節で述べたように，事前の学習を必要とすることなく，提案手法は DGA マルウェアの高精度な検出を実現している．機械学習や深層学習に基づく手法 [20], [21] は，判別精度の維持に潤沢な学習用データセットの準備が必須となるが，そのデータセットに対して良性と悪性のラベルを付与する作業は非常に煩雑である [31]．加えて，Sivaguru らは，最新の研究成果の比較を通じて，ホワイトリストやブラックリストを学習用データセットとして使用することに実用性の観点から疑問を呈している [32]．これは，ホワイトリストやブラックリストが，ネットワークで実測される良性ドメインと悪性ドメインの特徴を十分に反映するものでないことに起因する．それに対して，本手法が頼るのは DGA マルウェアとは直接関係しない辞書とウェブ検索のみである．これらは一般に公開されているものであり，その利用が容易である点を留意されたい．

提案手法による検出は，Gu らの BotHunter [11] とは異なり，マルウェアによる通信の暗号化の影響を受けないこと，Berger らの DNSMap [16] や Wang らの DBod [17] とは異なり，大規模なネットワークの観測を必要としないことがあげられる．特に，ネットワークにおける暗号化通信の重要性をふまえ，これまでに DoT（DNS over TLS）の標準化が推進されている．このプロトコルは，DNS の名前解決を TLS でカプセル化することにより，盗聴や改竄などを防ぐことを目的としている．一方，提案手法が良性と悪性を判別するために必要とするのは，ドメインの文字列のみである．したがって，ネットワーク内の RDNS が記録するシステムログの情報のみで動作可能であり，DoT による暗号化の影響を受けることはない．ただし，RDNS に対する名前解決が生じない DoH（DNS over HTTPS）については，本手法を適用することはできない．

提案手法の欠点は，ウェブ検索の結果に頼るため，他と比較して良性と悪性の判別に時間を要する点にある．各機能における 1 秒あたりに処理可能なドメイン数を算出すると，サブドメイン選択機能と辞書に基づく判別機能はそれぞれ 2968.51 と 1341.39 であるのに対して，ウェブ検索に基づく判別機能は 1.35 と非常に低速であるがゆえに，ネットワーク内で生じるすべての名前解決に適用することは困

難である．加えて，判別するドメイン数の増加にともない，ウェブ検索エンジンに対して多大な負荷をかけることとなる．一方，文献 [33] では，DGA によるコールバック先の変更にともない NXDOMAIN が発生することに着目している．この知見をふまえ，NXDOMAIN の頻度から感染の疑われる端末を絞り込むこと，そのいくつかの名前解決に対して本手法を適用することで，それら問題の大幅な緩和が期待できる．

Sood らは，自身が有する辞書の単語を連結することで悪性ドメインを生成する DGA マルウェアの代表例として，Rovnix について言及している [34]．そのような悪性ドメインの文字列は良性ドメインのものと類似した特徴を有するため，提案手法による判別は困難である．具体的には，`accelerateaccountant.in.net` や `accelerate actress.in.net` など，Rovnix が生成したドメインは，提案手法における辞書に基づく判別機能により良性と見なされる．この問題の解決には，Pereira らの取り組みのように，ドメイン文字列における単語間の関係性を考慮するなどの対策が必要となる [35]．

特筆すべきは，非常に限定された情報のみに依存するため，提案手法は高い汎用性を有する点である．具体的には，学習用データセットなどの事前知識を必要とせず，その判別はドメイン文字列のみに基づいている．ゆえに，他手法の機能の一部として組み込むことが容易であり，その判別性能の改善に大きく寄与すると考えられる．

5. おわりに

本稿では，DGA マルウェアの検出のため，表層的な文字列解析に基づく悪性ドメイン判別手法を提案した．その独自性は，DGA に関する事前情報をまったく必要とせず，ドメイン文字列の意味の有無からドメインの良性と悪性を推定する点にある．また実験を通じて，提案手法が 0.9960 の再現率と 0.9029 の適合率で悪性ドメインを判別可能であること，すなわち悪性ドメインの名前解決を予兆として DGA マルウェアを高精度で検出できることを確認した．この結果から，ネットワークに内在するマルウェアへの迅速な対処が可能となるため，ネットワークにおける安全性の向上が期待できる．

特筆すべき特徴は，本手法が有する高い汎用性である．これは，高精度な判別を実現することに加え，その判別はドメイン文字列のみに基づくこと，学習用データセットなどの事前知識を必要としないことに起因する．ゆえに，他手法の機能の一部として組み込むことが容易であり，その判別性能の改善に大きく寄与すると考えられる．

今後は，大規模なネットワークで観測した通信を対象に，本手法の有効性を評価する予定である．また，他手法との組合せの検討と，それが判別性能に与える効果を明らかにする．

謝辞 本研究は JSPS 科研費 JP18K11296 の助成を受けたものである．また，本研究の一部は電気通信普及財団の助成による成果である．ここに深く謝意を示す．

参考文献

[1] Lewis, J.A.: Economic Impact of Cybercrime – No Slowing Down (2018), available from ⟨https://www.csis.org/analysis/economic-impact-cybercrime⟩.
[2] Fu, Y. et al.: Stealthy Domain Generation Algorithms, *IEEE Trans. Information Forensics and Security*, Vol.12, No.6, pp.1430–1443 (2017).
[3] Kührer, M. et al.: Paint It Black: Evaluating the Effectiveness of Malware Blacklists, *Proc. International Symposium on Research in Attacks, Intrusions and Defenses*, pp.1–21 (2014).
[4] Chen, Z. et al.: Malware Characteristics and Threats on the Internet Ecosystem, *Journal of Systems and Software*, Vol.85, No.7, pp.1650–1672 (2012).
[5] Nelms, T. et al.: ExecScent: Mining for New C&C Domains in Live Networks with Adaptive Control Protocol Templates, *Proc. USENIX Conference on Security Symposium*, pp.589–604 (2013).
[6] Cisco Systems Inc.: Cisco Annual Cybersecurity Report (2018), available from ⟨https://www.cisco.com/c/dam/m/hu_hu/campaigns/security-hub/pdf/acr-2018.pdf⟩.
[7] Kim, D.: Potential Risk Analysis Method for Malware Distribution Networks, *IEEE Access*, Vol.7, pp.185157–185167 (2019).
[8] Dwyer, C. et al.: Malvertising – A Rising Threat to the Online Ecosystem, *Journal of Information Systems Applied Research*, Vol.10, No.3, pp.29–37 (2017).
[9] Soldo, F. et al.: Blacklisting Recommendation System: Using Spatio-Temporal Patterns to Predict Future Attacks, *IEEE Journal on Selected Areas in Communications*, Vol.29, No.7, pp.1423–1437 (2011).
[10] Freudiger, J. et al.: Controlled Data Sharing for Collaborative Predictive Blacklisting, *Proc. International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment*, pp.327–349 (2015).
[11] Gu, G. et al.: BotHunter: Detecting Malware Infection Through IDS-Driven Dialog Correlation, *Proc. USENIX Conference on Security Symposium*, pp.167–182 (2007).
[12] Cascarano, N. et al.: Optimizing Deep Packet Inspection for High-Speed Traffic Analysis, *Journal of Network and Systems Management*, Vol.19, No.1, pp.7–31 (2011).
[13] Parvat, T.J. et al.: Performance Improvement of Deep Packet Inspection for Intrusion Detection, *Proc. IEEE Global Conference on Wireless Computing & Networking*, pp.224–228 (2014).
[14] Rahbarinia, B. et al.: Efficient and Accurate Behavior-Based Tracking of Malware-Control Domains in Large ISP Networks, *ACM Trans. Privacy and Security*, Vol.19, No.2, pp.4:1–4:31 (2016).
[15] Bilge, L. et al.: Exposure: A Passive DNS Analysis Service to Detect and Report Malicious Domains, *ACM Trans. Information and System Security*, Vol.16, No.4, pp.14:1–14:28 (2014).
[16] Berger, A. et al.: Mining Agile DNS Traffic Using Graph Analysis for Cybercrime Detection, *Computer Networks*, Vol.100, pp.28–44 (2016).
[17] Wang, T.S. et al.: DBod: Clustering and Detecting DGA-based Botnets using DNS Traffic Analysis, *Computers & Security*, Vol.64, pp.1–15 (2017).

[18] Plohmann, D. et al.: A Comprehensive Measurement Study of Domain Generating Malware, *Proc. USENIX Conference on Security Symposium*, pp.263–278 (2016).
[19] Viglianisi, G. et al.: SysTaint: Assisting Reversing of Malicious Network Communications, *Proc. Software Security, Protection, and Reverse Engineering Workshop*, pp.1–12 (2018).
[20] Truong, D. et al.: Detecting Domain-Flux Botnet based on DNS Traffic Features in Managed Network, *Security and Communication Networks*, Vol.9, No.14, pp.2338–2347 (2016).
[21] Anderson, H.S. et al.: DeepDGA: Adversarially-Tuned Domain Generation and Detection, *Proc. ACM Workshop on Artificial Intelligence and Security*, pp.13–21 (2016).
[22] Vinayakumar, R. et al.: Evaluating Deep Learning Approaches to Characterize and Classify the DGAs at Scale, *Journal of Intelligent and Fuzzy Systems*, Vol.34, No.3, pp.1265–1276 (2018).
[23] Lin, W. et al.: Traceback Attacks in Cloud – Pebbletrace Botnet, *Proc. International Conference on Distributed Computing Systems Workshops*, pp.417–426 (2012).
[24] Wahab, O.A. et al.: Optimal Load Distribution for the Detection of VM-based DDoS Attacks in the Cloud, *IEEE Trans. Services Computing*, Vol.13, No.1, pp.114–129 (2020).
[25] LinguaJunkie.com: Russian Alphabet Guide, available from ⟨https://www.linguajunkie.com/wp-content/uploads/RussianAlphabetGuide_1_12-1.pdf⟩.
[26] Bader, J.: Some Results of My DGA Reversing Efforts, available from ⟨https://github.com/baderj/domain_generation_algorithms⟩.
[27] Hacker Target Pty. Ltd.: Download Top 1 Million Sites, available from ⟨https://hackertarget.com/top-million-site-list-download/⟩.
[28] Norvig, P.: Natural Language Corpus Data: Beautiful Data, available from ⟨http://norvig.com/ngrams/⟩.
[29] Linguatools: Wikipedia Monolingual Corpora, available from ⟨https://linguatools.org/tools/corpora/wikipedia-monolingual-corpora/⟩.
[30] Atkinson, K.: GNU Aspell, available from ⟨http://aspell.net⟩.
[31] Roh, Y. et al.: A Survey on Data Collection for Machine Learning: A Big Data – AI Integration Perspective, *IEEE Trans. Knowledge and Data Engineering* (2019).
[32] Sivaguru, R. et al.: An Evaluation of DGA Classifiers, *Proc. IEEE International Conference on Big Data*, pp.5058–5067 (2018).
[33] Antonakakis, M. et al.: From Throw-Away Traffic to Bots: Detecting the Rise of DGA-Based Malware, *Proc. USENIX Conference on Security Symposium*, pp.491–506 (2012).
[34] Sood, A.K. et al.: A Taxonomy of Domain-Generation Algorithms, *IEEE Security & Privacy*, Vol.14, No.4, pp.46–53 (2016).
[35] Pereira, M. et al.: Dictionary Extraction and Detection of Algorithmically Generated Domain Names in Passive DNS Traffic, *Proc. International Symposium on Research in Attacks, Intrusions, and Defenses*, pp.295–314 (2018).

佐藤 彰洋 （正会員）

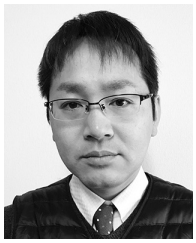
九州工業大学情報基盤センター助教．2011 年東北大学大学院情報科学研究科博士後期課程修了．博士（情報科学）．ネットワーク運用技術，ネットワークセキュリティに関する研究に従事．電子情報通信学会会員．

林 豊洋 （正会員）

九州工業大学情報基盤センター助教．2006 年九州工業大学大学院情報工学研究科博士後期課程修了．博士（情報工学）．情報システム，ロボットビジョン，パターン認識に関する研究に従事．電子情報通信学会会員．

和田 数字郎

九州工業大学飯塚キャンパス技術部技術専門職員．2003 年九州芸術工科大学大学院芸術工学研究科博士前期課程修了．修士（芸術工学）．ネットワークの運用に関する業務に従事．



福田 豊 （正会員）

九州工業大学情報基盤センター准教授．2005 年九州工業大学大学院情報工学研究科博士後期課程修了．博士（情報工学）．情報ネットワーク，無線 LAN に関する研究に従事．IEEE，電子情報通信学会会員．