マルウェア通信検知手法における User-Agent の有効性の一考察

市田 達也†

†株式会社リクルートテクノロジーズ
100-6640 東京都千代田区丸の内 1 丁目 9-2 グラントウキョウサウスタワー
tatsuya_ichida@r.recruit.co.jp

あらまし サイバー攻撃に用いられるマルウェアは，バックドアやボットなど感染後に外部サーバーと通信を行うことが一般であり，特にHTTP等の一般的なプロトコルにて行う場合が多い．近年，HTTPのUser-Agentヘッダを用いてマルウェア通信を識別する(UA監査)機能が実装されているセキュリティ製品もあるが，User-Agentをブラウザに偽装することでその検知を逃れようとするマルウェアも確認されている．本考察では，UA監査機能を持つ製品の有用性向上のため，マルウェア通信で利用されたUser-Agentの正常との逸脱度を算出し，User-Agentの有効性の評価および今後に向けた考察を行う．評価にはFFRIデータセットおよび独自環境にて取得したマルウェアを用いた．

A study for effectiveness of User-Agent for malware communication traffic detection

Tatsuya Ichida†

†Recruit Technologies Co.,Ltd.
Grand Tokyo South Tower
1-9-2 Marunouchi, Chiyoda-ku, Tokyo 100-6640, JAPAN
tatsuya_ichida@r.recruit.co.jp

Abstract A lot of malware communicate with their C&C servers via HTTP protocol. So some security appliances implement User-Agent inspection. However some malware pretend to be the browser agent and therefore its effectiveness is not clear today. In this study, I focused on User-Agent feature and evaluate the effectiveness for malware communication traffic detection using FFRI Datasets and our original captured malware.

1 背景と目的

マルウェアを用いたサイバー攻撃による被害は年々増え続けている．IPA が発行している「2014 年度 情報セキュリティ事象被害状況調査」[1]によると，日本企業におけるマルウェア遭遇率も増加傾向にある．その侵入経路は Web サイト閲覧や電子メールがいまも多く，2014 年は特に標的型メール含め，電子メールでの遭遇が増加した傾向が見えた．

一方で企業におけるセキュリティ対策も進展している．約 9 割の企業にクライアント端

末へのアンチウィルスソフトが導入されており，次いでファイアウォール等のゲートウェイ製品が普及している [1].

しかしながら，昨今の攻撃者およびそのマルウェアの高度化により，アンチウィルスソフトでは侵入時に検知できないマルウェアが多く存在する．多くの企業はファイアウォールやウェブプロキシ，ウェブゲートウェイにマルウェア配信元サーバーや感染後の接続先サーバー（C&C サーバー）の IP アドレスや FQDN，URL を登録し，二次感染防止のために感染後の出口対策を施している．しかしこれらも，接続先サーバー情報というシグニチャに依存したものであり，未知の悪性サーバーに対してマルウェア感染および情報漏洩を防ぐことは難しい．セキュリティベンダーである FireEye 社および Mandiant 社の 2014 年度の年次レポート[2]によると，攻撃者による侵入を自社で検知できる企業の割合は 3 分の 1 にも満たないらしい.

そこで検知率を向上させるために，ウェブプロキシやウェブゲートウェイ，次世代ファイアウォールには HTTP リクエスト内の User-Agent ヘッダによりマルウェアの通信かユーザーの正常通信かを判断できる仕組み（以下 UA 監査機能）が実装されている（[3][4]など）．これは独自の User-Agent を用いて HTTP 通信を実施するマルウェアが確認されていること[5]や，パッチを適切に適用している環境や，通信をするソフトウェアの勝手なインストールを制限している環境では，不審な User-Agent を抽出することで，マルウェア検知できることを示唆している[6].

しかしながら，UA 監査機能を利用しマルウェア感染後のインシデントレスポンスオペレーションに繋げるには，最近のマルウェアにおけるその有効性を評価しつづけなければならず, そのために評価指標が必要と考える.

本考察では，User-Agent によるマルウェア通信検知手法が最近のマルウェアにも有効かを評価するために，MWS Datasets 2015[7]の FFRI Dataset および独自環境にて取得したメール添付マルウェアが利用する User-Agent の正常との逸脱度を経年変化と共に算出し、その有効性を評価した.

本稿は，2 章にて User-Agent について説明し，3 章にて関連研究の紹介を経て，4 章にて FFRI Dataset での評価方法の紹介と評価結果を考察し，5 章では独自取得マルウェアを用いた比較実験を行った．最後に 6 章にて総括し，本研究のおける課題と今度の活用方法を示す.

2 User-Agent について

ウェブサーバーにアクセスする際，一般にブラウザから User-Agent 文字列が HTTP ヘッダとして送信される．この文字列は，使用されているブラウザ，そのバージョン番号，およびシステムの詳細 (オペレーティングステムとそのバージョンなど) を示すために利用され, Web サーバーは User-Agent 情報に基づいて，そのブラウザ用に最適化されたコンテンツを提供できる.

例えば,「Mozilla/5．0 (compatible; MSIE 10．0; Windows NT 6．1; Trident/6．0)」はアプリケーション「Mozilla」，バージョン「5．0」，「compatible」，ブラウザ「MSIE 10．0」，プラットフォーム「Windows NT 6．1」の属性に分類でき,「Windows NT 6. 1」は Windows 7 のオペレーションシステムを意味し，「Trident/6．0」は HTML レンダリングエンジンとそのバージョンである.

一般にブラウザでのアクセスの場合は User-Agent は「Mozilla」から始まり，残りはブラウザ種別やそのバージョン，プラットフォーム環境によって決まる．ブラウザではないバックグラウンドプログラムの場合は「Microsoft BITS/7．5」のような User-Agent になるものも存在する.

3　User-Agent の関連研究

マルウェアの HTTP 通信の中にはかつてより User-Agent に独自の文字列を付与するものが少なからず確認されてきた. C&C サーバーにアクセスしてきたエージェントが意図したマルウェアであると攻撃者が判断するためである．同時に C&C サーバーへのセキュリティ調査機関によるクローラーや一般ユーザーからの偶発したアクセスを識別できるという利点もある．防御側としては，User-Agent にこのような特徴があるため，多くの研究者によってマルウェア通信検知に応用する研究がなされている. ここ 2〜3 年の関連研究を年代の若い順に紹介する.

Nizar Kheir の関連研究[8]では，User-Agent を各属性文字列に分割しクラスタリングを施した後，最長共通部分列（LCS）によるシグニチャ抽出を行っている．さらに User-Agent をプロセスごとにランダムに変更して通信するマルウェアについては，その動的な振る舞いを検査しシグニチャとしている.

また Van Bolhuis らの関連研究[9]でも User-Agent を各属性文字列に区切った後，ある一定時間の属性文字列の出現頻度をホストごとに算出することで，マルウェア感染ホストと非感染ホストの識別および異常検知に有効であると示唆している.

最後に Yang Zhang らの関連研究[10]ではサードパーティにより組み込みが入ったブラウザで生成されるネストされた User-Agent では正規表現による監査をすり抜けることを指摘し，独自の自由度が高い構文解析木を提案している．解析木ではブラウザの User-Agent を細かく属性分割し，漏れなく定義することで異常な HTTP 通信を検知している.

これらの研究実績もあり，セキュリティベンダー各社侵入検知装置や次世代ファイアウォールなどでマルウェアごとに User-Agent シグニチャが作成され，プロキシやウェブゲートウェイに任意でカスタムルールを正規表現にて登録できる仕様もある.

しかしながら，マルウェアは高度化し続けており，ブラウザに偽装することでその検知を逃れようとするマルウェアも確認されているため，既存の検知手法がいまも通用するとは限らない．よって本考察ではマルウェア通信データを用いてどれほどのマルウェアが独自 User-Agent を利用し，その文字列が正常からの逸脱度を持ち，特徴として有効であるかを調査した．その定量評価のために，後述の逸脱指標 D を定義した．マルウェア通信データには，マルウェアをインターネットに接続できる開環境サンドボックスで実行させた挙動データである FFRI Dataset を利用した.

4　FFRI Dataset での評価

本考察では，最近のマルウェア通信の経年変化も考慮し，FFRI Dataset 2013， 2014，2015 の 3 年分のデータから HTTP 通信を発生させたマルウェア検体を利用する．該当検体数を表 1 に，カスペルスキー社の検知名を用いた検体分類を表 2〜4 に示す. なお便宜上カスペルスキー社に検知されていなかった検体数は表には含まれておらず, 1%以下のものは表示対象外とし，検知名を「. (ドット)」で区切り先頭分類名を用いた.

また HTTP 通信において FFRI Dataset 固有通信および Windows 固有通信は評価対象から除外した．特に FFRI Dataset 2015 からは，実行環境が Windows 8.1（64 ビット）であるため Windows 8.1 標準インストールアプリの通信も評価対象から除外した.

表 1.　FFRIDataset 検体における該当検体数

データセット名称	検体数	HTTP通信を行う検体数
FFRI Dataset 2013	2,638	256
FFRI Dataset 2014	3,000	598
FFRI Dataset 2015	3,000	905

表 2.　FFRI Dataset 2013 の HTTP 通信を行う検体分類

検知名の先頭分類	割合（%）
HEUR:Trojan	19.14
not-a-virus:HEUR:AdWare	16.02
Trojan	14.06
Backdoor	13.28
not-a-virus:AdWare	3.91
Trojan-Downloader	2.34
Trojan-Ransom	1.56

表 3.　FFRI Dataset 2014 の HTTP 通信を行う検体分類

検知名の先頭分類	割合（%）
not-a-virus:AdWare	29.10
HEUR:Trojan	14.38
not-a-virus:Downloader	14.05
Trojan	12.71
not-a-virus:HEUR:Downloader	3.85
Trojan-Downloader	3.51
Hoax	2.68
Trojan-Dropper	2.34
Worm	2.34
Trojan-Spy	2.01
Trojan-Ransom	1.34

表 4.　FFRI Dataset 2015 の HTTP 通信を行う検体分類

検知名の先頭分類	割合（%）
Trojan	19.91
HEUR:Trojan	14.38
Backdoor	10.18
Trojan-Banker	8.85
UDS:DangerousObject	8.74
Trojan-Downloader	7.96
Trojan-Ransom	4.54
not-a-virus:AdWare	2.54
Trojan-Spy	2.54
Worm	2.21
Trojan-Dropper	1.88
HEUR:Trojan-Downloader	1.22
Trojan-PSW	1.11
Hoax	1.00

4.1　HTTP 通信の特徴

4.1.1　FFRI Dataset 2013

接続先 URL の特徴として，攻撃者の C&C サーバーに対して「.txt」「.png」へのアクセスや「.bin」「.exe」の追加マルウェアコンポーネントをダウンロードする通信が多く見られた．また HTTP を利用し，直 IP アドレスに対して宛先ポート TCP/80 番以外への通信も確認された．

4.1.2　FFRI Dataset 2014

接続先 URL の特徴として，攻撃者の C&C サーバーに対して「.php」「.gif」へのアクセスや「.exe」へ追加のマルウェアのコンポーネントをダウンロードする通信が多く見られた．他には「ads.yahoo.com」など広告サービスへのアドウェアの通信も多く確認された．

4.1.3　FFRI Dataset 2015

接続先 URL の特徴として，攻撃者の C&C サーバーに対して「.php」「.js」へのアクセスや感染端末のグローバル IP アドレスを確認できる Web サイトへの通信も多く見られた．他には「microsoft.com」や「msn.com」，「google.com」，「virtualbox.org」など正規ドメインへのインターネットへの接続確認のための通信が多く確認された．

4.2　評価方法

4．1 の結果より FFRI Dataset 2013～2015 の 3 年間において取得できているマルウェア通信の内容が変化していることが確認できた．本節では 4．1 にて取得できた User-Agent と非感染端末から一般に見られる HTTP 通信の User-Agent 文字列との類似度を Levenshtein 距離により算出し，マルウェアが利用する User-Agent の正常からの逸脱度の評価指標とする．補足として本実験では「User-Agent ヘッダなし」は「User-Agent:（空白）」同様に扱う．

4.2.1　基本アルゴリズム（Levenshtein 距離）

Levenshtein 距離は編集距離とも呼ばれ，文字列 A から文字列 B を作るために要素の文字を最小で何回「挿入」「削除」「置換」する必要があるかを示す数を距離とみなしている．例えば「test」と「street」の距離であれば，「s」の「挿入」，「r」の挿入，「s」→「e」の置換の 3 回編集なので距離は 3 となる．Levenshtein 距離では例えば「top」と「hop」の距離と「top」と「pop」の距離が同じになり，「hop」と「pop」の意味の違いなどは考慮できないが，本考察では文字列の逸脱度の指標として近年の関連研究[11]でも利用されている Levenshtein 距離を標準化した NLD（Normlised Levenshtein Distance）を利用する．その導出式を以下(1)に示す．NLD では，文字列 x と文字列 y が完全一致する場合は 0 となし，全く一致しない場合は 1 となる．

$$NLD(x,y) = \frac{Levenshtein\ Distance}{max(len(x), len(y))} (1)$$

4.2.2　NLD を用いた評価アルゴリズム

次節 4．3 の正常系 User-Agent 文字列 K 種類と 4．1 で抽出した各年の FFRI Dataset における HTTP 通信内の User-Agent 文字列 N 種類との距離を全て計算し，Dataset としての正常系との逸脱指標 D を(2)にて算出する．c は Dataset 内の各 User-Agent の出現回数である．D の導出式を以下(2)に示す．

$$D = \left\{ \frac{\sum_{k=1}^{K} \left(\sum_{n=1}^{N} NLD(x_{k,}y_n) c\,(y_n) \right)}{\sum_{n=1}^{N} c(y_n)} \right\} (2)$$

4.3　比較する正常系 User-Agent

本考察では，FFRI Dataset と比較するにあたり，Windows 7 非感染端末（32 ビット，64 ビット）における意図的に発生させたウェブブラウザ通信と OS バックグラウンド制御通信を数端末で長時間パケットキャプチャすることで取得した．ウェブブラウザは経年変化も考慮し，Internet Explorer 7 から 11 と FireFox および Google Chrome を用いた．この度利用した正常系 User-Agent のリストを表 5 に示す．

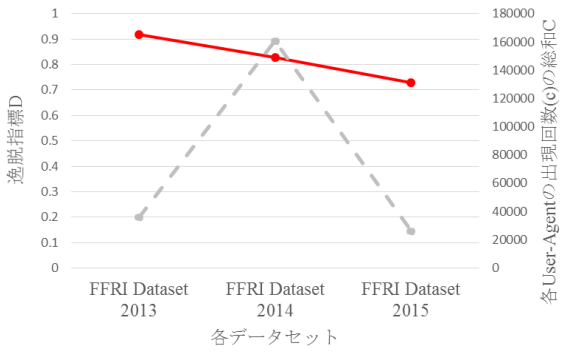
表 5．正常系 User-Agent 一覧

正常系User-Agent
Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322)
Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.1; Trident/4.0; .NET CLR 2.0.50727; .NET CLR 3.0.04506.30; .NET CLR 3.0.04506.648)
Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; WOW64; Trident/5.0)
Mozilla/5.0 (compatible; MSIE 10.0; Windows NT 6.1; Trident/6.0)
Mozilla/5.0 (Windows NT 6.1; WOW64; Trident/7.0; rv:11.0) like Gecko
Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.31 (KHTML. like Gecko) Chrome/26.0.1410.64 Safari/537.31
Mozilla/5.0 (Windows NT 6.1; WOW64; rv:30.0) Gecko/20100101 Firefox/30.0
Microsoft-CryptoAPI/6.1
Microsoft BITS/7.5

4.4　評価結果

FFRI Dataset における逸脱指標 D と各 User-Agent の出現回数(c)の総和 C の経年変化を図 1 に示す．

図 1．FFRI Dataset における逸脱指標 D の経年変化(実線：D<左軸>，破線：C<右軸>)



4.5　考察

図 1 より，逸脱指標 D（赤実線）は減少傾向であった．FFRI Dataset にて評価すると，年を重ねるほどマルウェアの User-Agent もブラウザもしくは OS バックグラウンド通信に似せてきていると考えられる．また User-Agent の出現回数の総和 C（破線）は FFRI Dataset 2014 のみ急激に増加しているが，これはアドウェアの広告通信が多量であったことに起因していた．

正常からの逸脱度が大きかった User-Agent は「User-Agent ヘッダなし」や「（空白）」，他に極端に短い文字列であった．これは Levenshtein 距離では削除距離も考慮されるためと考えられる．D の増加要因を調査したところ，これら文字列長が極端に短いか，または少数だがランダムに長い User-Agent であった．NLD が 1，もしくは 1 に限りなく近づいた 3 文字以下の User-Agent を持つ検体割合を図 2 に示す．

図 2.　User-Agent が 3 文字以下の検体割合（該当検体/HTTP 通信発生検体）

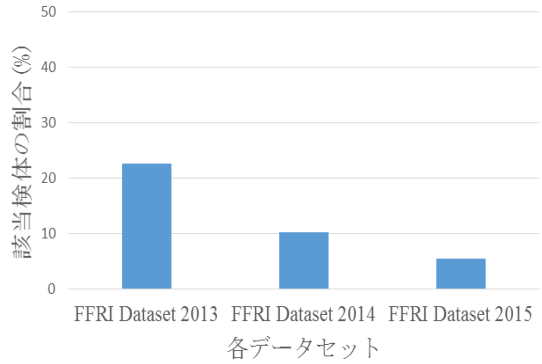


図 2 より HTTP 通信を発生させる検体中の User-Agent が 3 文字以下で通信する検体割合は経年で減少傾向であった．しかし，該当検体数は各年約 50 検体が確認された．つまり，FFRI Dataset 全体において 2%の検体が該当し，これらを User-Agent の長さに着目してマルウェア通信と判断できることが示唆できた．一方で割合が年々減少した理由は，ブラウザや Windows 固有通信の User-Agent を模倣して通信するマルウェアの増加が考えられる．また FFRI Dataset はウェブサイトのクローリング等によって主に取得できた検体が多いため，このような特徴が現れたとも考えられる．よって比較実験として，5 章でメールに添付されて侵入してくるマルウェアについても評価する．

5　独自取得マルウェアでの評価

本章ではメール添付マルウェアについて独自環境にて取得し，同様に逸脱指標 D を算出する．マルウェア取得期間は 2014 年 1 月～2015 年 6 月であり，Windows 7（32 ビット，64 ビット）のサンドボックス環境にて数分間動作させてマルウェアの HTTP 通信を取得した．4 章と同様に HTTP 通信が確認されたマルウェアのカスペルスキー社の検知名を用いた検体分類を表 6 に示す．

表 6. 独自取得マルウェアの HTTP 通信を行う検体分類

検知名の先頭分類	割合 (%)
Trojan-Downloader	38.89
Trojan	27.27
Trojan-Spy	13.47
Trojan-Dropper	4.38
Trojan-PSW	3.87
Trojan-Ransom	3.54
HEUR:Trojan	1.52
Backdoor	1.18

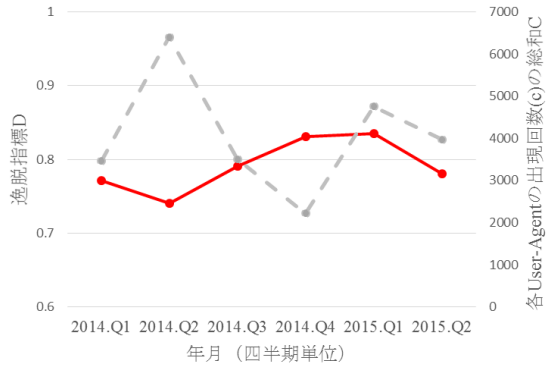
5.1　HTTP 通信の特徴

表 6 のカスペルスキーの検知名の先頭分類でも「Downloader」「Spy」が多く見られていた通り，「.exe」「.dat」「/update」への追加コンポーネントのダウンロード通信や Zbot 等のバンキングトロイが C&C サーバーの「.php」へアクセスする通信が多く確認された．

5.2　評価結果

独自取得マルウェアにおける逸脱指標Dと各User-Agentの出現回数（c）の総和Cの推移を2014年1月から四半期（Q）単位で図3に示す.

図 3. 独自取得マルウェアにおける逸脱指標Dの変化(実線：D<左軸>，破線：C<右軸>)



5.3　考察

図3より逸脱指標D（赤実線）は2014年にかけて増加傾向であったが，全体を通して四半期(Q)単位で顕著な増加傾向は見られなかった．この結果の要因は4．5節同様User-Agentが極端に短いか「User-Agentヘッダなし」の検体であった．図2同様にQごとの該当割合を図4に示す.

図 4. Qごとの User-Agent が 3 文字以下で通信を行う該当検体の割合(該当検体/HTTP 通信発生検体)

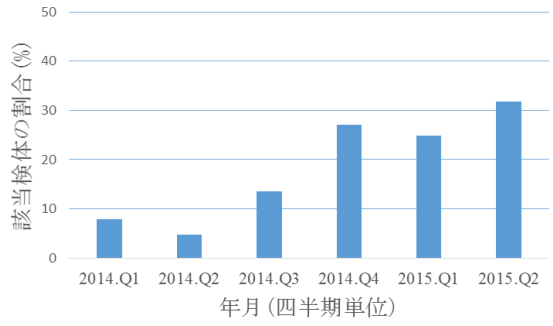


図4よりHTTP通信を発生させる検体群中のUser-Agentが3文字以下である検体の割合は，2014年からの四半期単位で増加傾向であり，今後も検知に有益な特徴である可能性がある．直近の2015年Q2（4月～6月）では，約30%のマルウェアHTTP通信をUser-Agentの長さのみで識別できる特徴が確認できた．また2015年Q2において逸脱指標Dを増加させる検体の割合が多いにもかかわらず，図3のように逸脱指標D（赤実線）が低下した理由は「Google Chrome」のUser-Agentを模倣して通信するマルウェアが多数含まれていたためである．つまり正常系User-Agentに模倣したデータによって逸脱指標Dは相殺されることがわかった．よって（2）式の逸脱指標Dのみでそのデータ空間のUser-Agentがマルウェア検知に有効であるかの評価は難しく，データ内部の要因分析結果とデータの偏りを反映できる算出式に最適化することが必要である.

6　本考察のまとめと今後の検討

本研究では，マルウェア感染後のHTTP通信を検知する手法におけるUser-Agentヘッダの有効性をFFRI Datasetおよび独自取得のメール添付マルウェアにて評価した．逸脱指標Dを用いた評価実験の結果，この度の逸脱指標Dのみを用いて総合評価することは難しいが，User-Agentが極端に短いHTTP通信を行う検体は最近も一定数存在し，特にメールに添付されるマルウェアでは顕著な増加傾向が見られた．本評価はTrue Positiveの観点のみかつ，1サンプルデータによる評価ではあるが，User-Agentヘッダの有無とUser-Agentの文字列長のみを利用しただけでも，HTTP通信を行うマルウェアのうち約20%～30%も検知できる可能性があることが確認できた．これより，User-Agentの特徴が今後も有効である可能性を確認した.

一般的な法人企業ではファイアウォールで業務に不必要な宛先ポート番号への通信はブ

ロックしていることが多く，業務でも利用頻度の高い HTTP 通信の監視に注力すべきと考える．当然ながら対外への通信に対して UA 監査機能の適用を考えた場合，False Positive による通信遮断を避ける必要がある．つまり独自 User-Agent を利用する巷のフリーツールや独自開発ツールの調査を十分に行った上で適用すべきである．その際，User-Agent の一覧として Web サイト[12]等も参考になりうる．

本考察の留意事項として加えて 2 点記載する．User-Agent が 3 文字以下に着目したが，この文字数は比較対象の正常と定義した User-Agent の長さに依存するものである．また「User-Agent ヘッダなし」の通信はセキュリティ製品の UA 監査機能では容易にフィルタできない可能性も高い．その場合でも少なくとも本特徴は，SIEM（Security Information and Event Management）等のログ相関分析・管理プラットフォームの一要素として利用できると考える．

一方でマルウェアにはブラウザをハイジャックするもの，MITB（Man In The Browser）攻撃や端末の User-Agent をマルウェアが調査し，通信に活用するものも多数存在し，また HTTPS 通信には対応していないため，あくまで User-Agent による識別は防御の 1 要素と考えている．

今後，定期的に User-Agent の有効性を容易に評価し続けるために，識別指標 D の磨き込みを検討する．既存研究でも利用されている属性ごとに区分した上で最長共通部分列（LCS）を用いることや，正常系文字列より一文字だけ変更されている類似度が高いものに対して，変更箇所に依らず逸脱度を判断できる指標を検討したい．

参考文献

[1] 「2014 年度 情報セキュリティ事象被害状況調査-報告書-」
http://www.ipa.go.jp/files/000043418.pdf
(参照 2015/08/15)

[2] 2015 年版 M-Trends：サイバー脅威最前線からの見解（M-Trends 2015: A View from the Front Lines）
https://www2.fireeye.com/WEB-2015RPTM-Trends.html (参照 2015/08/15)

[3] BlueCoat ウェブプロキシでの活用例
https://www.bluecoat.com/security-blog/2014-04-29/protecting-your-organization’s-web-browsing-new-internet-explorer (参照 2015/08/15)

[4] McAfee ウェブゲートウェイでの活用例
http://www.dit.co.jp/ditplus/focus/series_mcafee/vol1.html (参照 2015/08/15)

[5] Fortinet Security Blog
http://www.fortinet.co.jp/security_blog/131028-The-Stealthy-Downloader.html (参照 2015/08/15)

[6] 特定非営利活動法人 日本セキュリティ監査協会，“APT 対策入門: 新型サイバー攻撃の検知と対応，”APT による攻撃対策と情報セキュリティ監査研究会， Next Publishing 2012

[7] 神薗雅紀, 秋山満昭, 笠間貴弘, 村上純一， 畑田充弘, 寺田真敏 “マルウェア対策のための研究用データセット～MWS Datasets 2015～，” 情報処理学会 研究報告コンピュータセキュリティ（CSEC）Vol.2015-CSEC-70，No.6，pp.1-8，2015

[8] Nizar Kheir “Analyzing HTTP User Agent Anomalies for Malware Detection，” Data Privacy Management and Autonomous Spontaneous Security, 187-200，2013

[9] Van Bolhuis ら “Anomaly Detection on Internet Content Filter Data，” University of Amsterdam 2014

[10] Yang Zhang ら “Detecting Malicious Activities with User-Agent Based Profiles，” Minesota.Univ， Int. J. Network Mgmt 2015; 00:1-25

[11] 北條孝佳, 松浦幹太“文字列類似性を考慮した標的型攻撃のグループ化 手 法 ， ” MWS2014 (2014 年 10 月)

[12] List of User-Agents
http://www.user-agents.org(参照 2015/08/15)