悪性 Web サイトを分析するための マルチ環境解析における通信ログ解析の効率化

西尾 祐哉[†1]　廣友 雅徳[†1]　福田 洋治[†2]　毛利 公美[†3]　白石 善明[†4]

概要：Drive-by Download 攻撃に利用される悪性 Web サイトには，アクセスしてきた端末の環境によって挙動を変えるものがある．そのため，単一の解析環境では十分な情報を得ることができず，悪性な挙動を見逃してしまう可能性がある．また，悪性 Web サイトで利用される JavaScript は難読化が施されている場合が多く，それが解析を困難にしている要因にもなっている．著者らはマルチ環境解析の動的解析と難読化コードの解析を組み合せた方法を示している．一方，より多くの悪性サイトを分析するためには分析作業の効率化も必要となる．本稿では，マルチ環境解析の一部である通信ログ解析を自動化する方法を示す．

キーワード：Drive-by Download 攻撃，マルチ環境解析，Web サイト解析，通信ログ解析

Efficiency Improvement of Traffic Log Analysis of Multi-Environment Analysis for Malicious Websites

Yuya Nishio[†1]　Masanori Hirotomo[†1]　Youji Fukuta[†2]　Masami Mohri[†3]
Yoshiaki Shiraishi[†4]

Abstract: The malicious websites used by Drive-by Download Attacks change their behavior for web client environments. Therefore, single-environment analysis cannot obtain sufficient information and may miss a malicious behavior. In addition, JavaScript used in malicious websites is often obfuscated, and it becomes the factor making analysis difficult. In this paper, we propose the analysis method based on the multi-environment analysis of the traffic log analysis was partially automated in order to analyze efficiently the more malicious websites.

Keywords: Drive-by Download Attack, multi-environment analysis, web site analysis, traffic log analysis

1. まえがき

近年，マルウェアを利用したサイバー攻撃が世界中で発生しており，その中でも，悪性な Web サイトへのアクセスを契機として，強制的にマルウェア感染させる Drive-by Download（DBD）攻撃が猛威を振るっている．

現在の DBD 攻撃の対策として，ユーザの悪性 Web サイトへのアクセスを防ぐために，Microsoft や Google が提供している URL ブラックリストを用いたアクセスブロック機能が存在している[1][2]．しかし，攻撃者によって正規サイトが改ざんされ，マルウェアを配布する Web サイトにリダイレクトさせる悪性なコードが埋め込まれることがあることと，悪性 Web サイトの多くはその URL を短期間で遷移させていることから，ブラックリストによる防御には限界がある．また，悪性 Web サイトは，アクセスしてきた端末の環境を識別して挙動を変える特徴があるため，既存の Web サイト解析サービスだけでは十分な情報が得られるとは限られない．悪性 Web サイトの全容を把握するには複数環境による解析作業が必要となるため，文献[3]では，複数環境で実装された Web クライアント型ハニーポットを用いる解析技術が示されている．しかしながら，文献[3]では複数環境による解析手法を提案しているが，環境に導入しているプラグインの種類が少なく，十分な評価ができていない．筆者らは文献[4]では，マルチ環境解析を用いて悪性 Web サイトの挙動を詳細に把握することを目的に，マルチ環境解析による動的解析と難読化コードの解析を組み合わせた分析手法を示した．また，攻撃対象となるプラグインを十分に用意し，その分析手法を実装した環境を用いて分析した結果を報告した．

本稿では，マルチ環境解析の効率化を目的として，マルチ環境解析の一部である，通信ログ解析を自動化する方法を示す．これにより，通信ログ解析における悪性な通信を抽出する煩雑な作業の手間を削減し，解析作業の効率を上げられる．また，本システムを用いて悪性サイトの分析を行った事例を述べる．

†1 佐賀大学大学院工学系研究科
Graduate school of Science and Engineering, Saga University
†2 近畿大学理工学部情報学科
Faculty of Science and Engineering, Kindai University
†3 岐阜大学総合情報メディアセンター
Information and Multimedia Center, Gifu University
†4 神戸大学大学院工学研究科
Graduate School of Engineering, Kobe University

2. Drive-by Download 攻撃

2.1 Drive-by Download 攻撃の概要

Drive-by Download（DBD）攻撃とは，ユーザが悪性な Web サイトにアクセスすると，複数回のリダイレクトを経てマルウェア配布サイトへ誘導され，ブラウザやプラグインの脆弱性が悪用されて強制的にマルウェアがダウンロードされる攻撃である．従来のインターネットでの攻撃手法では，攻撃者が攻撃対象者に悪意のある情報を送る能動的攻撃であった．しかし，DBD 攻撃はユーザの Web サイトへのアクセスを攻撃の起点とし，攻撃者が攻撃対象者からの要求を受けて悪意のある情報を応答するため，受動的攻撃である．図 1 では DBD 攻撃の流れを示している．

攻撃に関与する Web サイトは，入口サイト，中継サイト，攻撃サイト，マルウェア配布サイトの 4 つからなり，それぞれの役割は次のようになる．

[入口サイト]

攻撃の起点となるサイトである．アクセスすると，中継サイトへリダイレクトさせる．攻撃者が作成したサイトとは限らず，正規サイトを改ざんし，入口サイトとすることが多い．

[中継サイト]

攻撃サイトへ中継するサイトである．リダイレクトによって複数の中継サイトを経由することが多い．

[攻撃サイト]

クライアント PC の OS や Web ブラウザ，Web ブラウザ上で動作するプラグインの脆弱性を悪用し，マルウェア配布サイトからマルウェアをダウンロードするスクリプトを実行させる．

[マルウェア配布サイト]

攻撃者によって乗っ取られた端末のリクエストに応えてマルウェアをダウンロードし，強制的に実行させることでマルウェアに感染させる．

DBD 攻撃は上記の複数の Web サイトが連動して行われるが，ダウンロード画面やインストール画面，リダイレクト時の画面変化等は表示されないため，ユーザは攻撃を受けたことに気付くことができない．入口サイトは，正規サイトを改ざんして外部の悪性サイトにアクセスさせていることが多い．この場合に HTML の iframe タグがよく使用される．iframe とは，1 つの Web ページ内に別の Web ページを埋め込み，一体的に表示することができるタグである．攻撃者はこの機能を悪用し，フレームの表示サイズを小さく設定することで挿入した悪性サイトを見えないようにしている．改ざんによって挿入されたコードには難読化が施されている場合があり，コードの目的を隠蔽している．

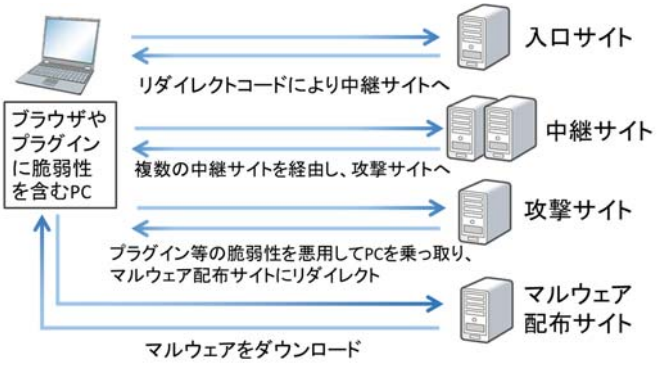

図 1　Drive-by Download 攻撃の流れ

2.2 悪性 Web サイトの挙動の特徴

DBD 攻撃で利用される悪性 Web サイトには，検知を逃れるために次のような特徴がある．

- **実行環境によって異なる挙動をする**

アクセスしてきた端末のブラウザやプラグインの種類，バージョンによって異なる挙動をする．図 2 は Exploit Kit を利用した悪性 Web サイトによる攻撃の一例を示している．この図を例に挙げると，攻撃サイトに到達した時点で攻撃対象端末のプラグインの種類とバージョンを識別している．もし端末に標的とするプラグインが存在していれば，そのバージョンに応じた脆弱性を悪用する攻撃を行う．しかし，端末に標的とするプラグインのバージョンが存在していなければ，攻撃を行わずに無害な Web サイトへリダイレクトさせるといった挙動を示す[5]．この特徴により，標的の環境以外でアクセスした場合，悪性な挙動を検知できなくなる．

- **初回アクセス時にのみ悪性な挙動をする**

アクセスしてきた端末の IP アドレスを記録し，初回アクセス時には悪性な挙動を示すが，2 回目以降のアクセスでは悪性な挙動を示さなくなる．また，Cookie を使用して初回アクセスか否かを判定するケースも存在する．

- **攻撃時に複数の脆弱性を悪用する**

マルウェアをダウンロードさせるために複数の脆弱性を悪用する．これは攻撃の成功率を高める意図と，どの脆弱性が悪用されたかを把握しにくくし，解析を困難にさせる意図がある．

悪性 Web サイトはこのような特徴を有しているため，解析が困難となる．特に，アクセスしてきた端末の環境によって挙動が変化すると，単一の解析環境では正確なデータを入手できない可能性があり，悪性 Web サイトの挙動の把握をすることも困難である．そこで，悪性 Web サイトの挙動を把握するには複数の環境で解析作業を行う必要がある．

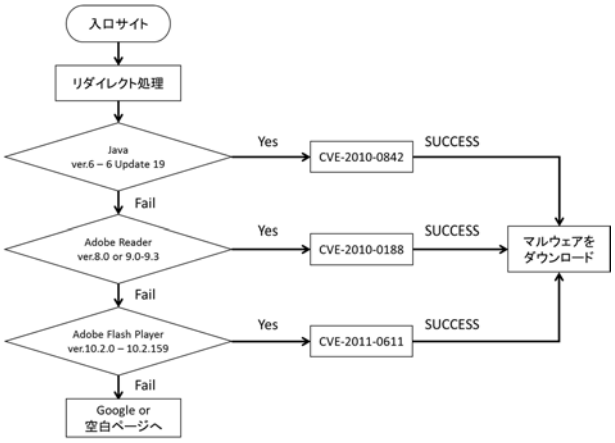

図 2　悪性 Web サイトによる攻撃の一例

3. マルチ環境解析の開発方針

3.1 マルチ環境解析の概要

マルチ環境解析の処理内容としては，高対話型の Web クライアント型ハニーポットである．一般的に Web クライアント型ハニーポットは，実際のアプリケーションを利用する高対話型と，アプリケーションをエミュレートしたものを利用する低対話型に分けられるが，本稿ではより詳細な情報を得ることを目的としているため，高対話型の Web クライアント型ハニーポットとしてマルチ環境解析を開発する．

マルチ環境解析の基本的な構成は，仮想端末上にそれぞれ種類やバージョンの異なるブラウザやプラグインを導入し，複数環境での解析作業を実現させたものである．様々な環境上で解析することにより，単一の環境では把握できなかった悪性 Web サイトの挙動を逃すことなく捕捉できるようになる．

3.2 分析手法

マルウェアの解析方法として，一般的に動的解析と静的解析の 2 つの手法がある．動的解析は実際にマルウェアを仮想環境などで実行させ，その動作を監視することによって解析を行う．それに対して静的解析では，リバースエンジニアリングなどの技術を用いてマルウェアのソースコードを解析していく．本稿ではマルチ環境解析を用いて主に動的解析を行い，さらに悪性 Web サイトのコードを静的解析していくことによって，より詳細な情報を取得していく．具体的には次のような分析手法を用いる．

[通信ログ解析]

マルチ環境解析で得られた通信データから，リダイレクト先 URL やリダイレクト元 URL を特定することができる．それらの URL を，ファイアウォールのログやアンチウイルスソフトのログと突合することで，脆弱性を悪用する攻撃サイトまでリダイレクトされているか，マルウェアのダウンロードまで至っているのかといった情報を得ることができる．また，各解析環境の通信データを比較することにより，悪性 Web サイトの環境による挙動の違いをある程度把握することができる．

[コンテンツ解析]

攻撃を受けた原因を明らかにするために，取得した通信データから，悪性 Web サイトの HTML ファイル，JavaScript ファイル，PDF などのコンテンツファイルを再構築し，これらのファイルに含まれる悪性なコードを解析することで，悪用される脆弱性などを把握する．また，悪性 Web サイトで利用されるコードは難読化が施されている場合が多いため，そのような難読化コードの解読を行うことで，通信ログ解析では把握しきれなかった悪性 Web サイトの挙動を捕捉する．疑わしいファイルに関しては，無料でファイルの分析を行ってくれる VirusTotal[6]などのサービスを活用し，ある程度悪性なのか判断する．

4. マルチ環境解析の実装

4.1 マルチ環境解析の構成

マルチ環境解析の構成を図 3 に示す．解析環境制御モジュールで仮想マシンの起動，解析対象 URL の送信を行う．Web ブラウザ制御モジュールでは，クローラによるクローリング処理を自動化させる．このクローラに Web アプリケーションのテストツールである Selenium Webdriver[7]を用いることで，Web ブラウザの操作を自動化させる．マルウェア解析モジュールでは，3.2 節で述べた分析手法を用いて解析を行う．

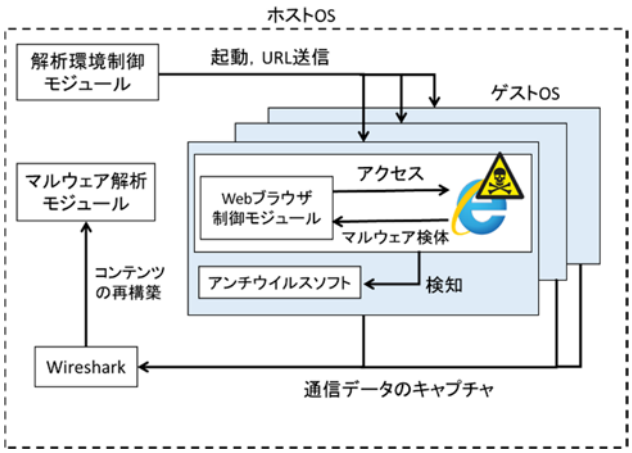

図 3　マルチ環境解析の構成

4.2 実装

マルチ環境解析の実装環境を表 1 と表 2 に示す．表 1 は各解析環境で使用しているソフトであり，Web サイトを解析するための複数の環境を VMware Player を利用して仮想的に構築した．また，表 2 で示している 2 種類のブラウザに対して，6 種類のプラグインを 1 つずつ導入して解析環境を構築した．具体的には，Internet Explorer 8 のブラウザに対して 6 種類のプラグインをそれぞれ 1 つずつ導入し

た解析環境を 6 個，同じように Internet Explorer 11 にも 6 種類のプラグインをそれぞれ 1 つずつ導入した解析環境を 6 個，つまり合計で 12 個の解析環境を構築した．使用したプラグインのバージョンは，2016 年 7 月 21 日時点で脆弱性の数が最も多いものと，2 番目に多いものを使用している．ただし Internet Explorer 11 では，JRE 1.6 update 24 以降のバージョンしか対応していないため，そのバージョンを使用している．

表 1　実装環境の共通使用ソフト

仮想マシン	VMware Player 12.1.0
ゲスト OS	Windows7 Professional
ホスト OS	Windows7 Professional
アンチウイルスソフト	Symantec Endpoint Protection

表 2　実装環境のブラウザとプラグイン

ブラウザ	Internet Explorer 8	Internet Explorer 11
導入したプラグイン	・Adobe Flash 10.0 ・Adobe Flash 19.0 ・Adobe Reader 10.1 ・Adobe Reader 9.1 ・JRE 1.7 ・JRE 1.6 update 22	・Adobe Flash 10.0 ・Adobe Flash 19.0 ・Adobe Reader10.1 ・Adobe Reader 9.1 ・JRE 1.7 ・JRE 1.6 update 24

4.3 通信ログ解析の自動化

解析作業を全て手作業で行う場合，多くの時間を要するため，通信ログ解析の一部を本システムで処理させることで，解析作業の効率化を図る．Wireshark でキャプチャした pcap 形式のパケットファイルを，tshark を用いて HTTP パケットのみ抽出し，さらにテキストファイル形式で出力する．本システムでは，このテキスト形式の HTTP パケットデータを基に解析を行う．また，本システムは Java で実装した．具体的なシステムの機能は次の 4 つである．

[リダイレクト情報の抽出]

HTTP ステータスコードがリダイレクションを表す 300 番台のレスポンスからリダイレクト情報を抽出する．該当する HTTP レスポンスから，フレーム番号，ステータスコード，リダイレクト元の IP アドレス，リダイレクト先 URL を取得し，テキストファイルに書き出す．しかし，リダイレクトは全て悪性のものとは限らず，正規の通信でもリダイレクトを発生させることはある．そこで，事前に用意したホスト名のホワイトリストを用いることで，不審なリダイレクトのみを抽出する．具体的には，リダイレクト先 URL のホスト名とホワイトリストに登録されているホスト名をマッチングさせ，もしマッチした場合は書き出しを行わない．

[HTTP リファラに着目したリダイレクト情報の抽出]

Drive-by Download 攻撃では，不正な iframe によって悪性 Web サイトへリダイレクトさせるケースが多い．しかし，HTML 上の iframe タグなどは難読化が施されていることが多いため，容易に iframe タグを見つけることができない．そこで，HTTP リファラに着目することで，iframe による不正リダイレクトの検出を試みる．HTTP リファラとは，あるページのリンクをクリックして移動した際などの，リンク元の URL のことである．

プログラムによる抽出方法の流れを図 4 に示す．具体的にはリファラヘッダが含まれる HTTP リクエストを見つけ，そのリファラのホスト名とホストヘッダに記述されているホスト名を比較する．その際にホスト名が異なる場合，外部のコンテンツにアクセスさせていることが分かる．しかし，外部の画像などを参照している場合も多いため，HTTP リクエストの GET メソッドの値を確認する．このとき，URL の末尾が.jpg や.png，.css などの場合は検出対象としない．また，事前に用意したホスト名のホワイトリストと，ホストヘッダの値が一致した場合も検出対象としない．一致しなかった場合は iframe による不正リダイレクトの可能性が高いと判断でき，その HTTP リクエストのフレーム番号，アクセス先 IP アドレスと URL，リファラの値をテキスト形式で出力する．

[アンチウイルスソフトのログとの突合]

アンチウイルスソフトが不正なリダイレクトや通信を検知した際に記録されるログデータを使用する．ログに記録されている URL や IP アドレスなどを抽出し，それに対応する HTTP パケットを書き出す．その情報によって，該当する Web サイトから何がダウンロードされたか，またリダイレクト情報などを知ることができる．

さらに，上記の HTTP リファラに着目したリダイレクト情報の抽出機能と組み合わせることで，検知 URL をリファラとしている iframe を利用したリダイレクトなどを検出できると考えている．

[ダウンロードコンテンツ情報の抽出]

パケットデータからダウンロードされたコンテンツに関する情報（ファイル名，ダウンロード元 URL など）を抽出し，csv 形式で一覧として出力する．実際にマルウェアに感染させる際には，実行ファイルや flash ファイルなどをダウンロードさせるため，そのような特定の種類のファイルのみ抽出するようにし，不要なファイルはできるだけ抽出しないようにする．具体的には HTTP レスポンスヘッダ内のコンテンツタイプの値でフィルタリングする．csv に出力する際には，HTTP レスポンスのフレーム番号，ダウンロード元のホスト名，コンテンツタイプ，ダウンロードファイル名を書き出す．

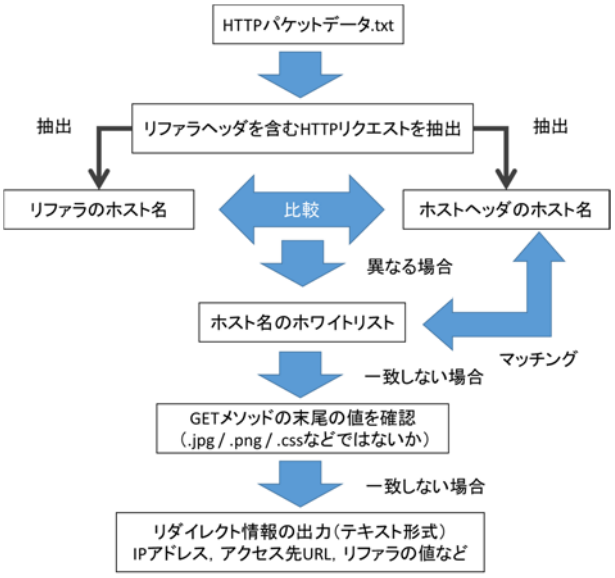

図 4　HTTP リファラに着目したリダイレクト情報の抽出

5.　分析事例

ブラックリスト Malware Domain List[8]に掲載されている悪性 Web サイトに実際にマルチ環境で解析し，その結果を分析する．今回の実験で利用した URL は，Malware Domain List に 2016 年 4 月 28 日から 7 月 21 日の期間に登録された，脆弱性を悪用する Web サイト 68 個である．表 1，表 2 で示した環境によって 2016 年 8 月 7 日に実験を行ったところ，HTTP レスポンスコードが 404 など，既に削除されているためにアクセスできなかった Web サイトは 68 個中 44 個であった．そのため，残りの 24 個の Web サイトについて解析を行った．

5.1　実験結果

24 個の Web サイトに 12 種類の解析環境でアクセスしたところ，Symantec Endpoint Protection（SEP）が検知した Web サイトは 24 個中 0 個であった．そのため，今回はアンチウイルスソフトのログと通信データの突合は行わずに，3.2 節で示した分析手法と，4.3 節で示した 解析プログラムを用いて解析を行う．

5.2　通信ログ解析

5.2.1 ダウンロードコンテンツ情報の分析

4.3 節のプログラムを用いてダウンロードコンテンツ情報の抽出を行ったところ，Adobe Flash Player を導入している解析環境から，合計 6 種類の swf ファイルがダウンロードされていることが確認できた．その 6 個のファイルを通信データから再構築し，VirusTotal で分析したところ，全てのファイルに対して検出率は 0 で，良性だと判定された．この 6 個の Flash ファイルを Flare[9]というツールでデコンパイルし，アクション部分を抽出して確認してみたところ，不審だと考えられるコードは特に記述されていなかった．また，この Flash ファイル以外には特に不審なコンテンツはダウンロードされていなかった．

5.2.2 リダイレクト情報の分析

4.3 節のプログラムを用いてリダイレクト情報の抽出を行い，リダイレクト先の URL について解析したところ，そのほとんどが 1×1px の GIF 画像であった．これはアクセス解析などで用いられる Web ビーコンだと考えられ，直接ユーザに危害を加えることはない．また，リダイレクト先が悪性 Web サイトだと考えられるものは確認できなかった．

5.2.3 HTTP リファラに着目した分析

4.3 節のプログラムを用いて HTTP リファラに着目したリダイレクト情報の抽出を行ったところ，17 個の Web サイトから同じ Web サイトにアクセスしていることが確認できた．これは全ての解析環境で確認することができた．また，抽出したリダイレクト情報を各解析環境同士で比較したところ，リダイレクト先 URL は短時間で変化していることが確認できた．実験は解析環境 1 台ずつ順番に行っており，実験時間と URL の変化のタイミングを考えると，リダイレクト先 URL のホスト以降のディレクトリ，ファイル名は約 1 時間毎に変わっており，ホスト名はさらに短い間隔で変わっていることが分かった．また，リダイレクト先 IP アドレスを見てみると，1～7 番目に実験した環境で同じで，8～12 番目に実験した環境で同じであった．このことから，リダイレクト先 IP アドレスは URL の変化よりもさらに長い間隔で変化していると考えられる．リダイレクト先 URL の遷移例を図 5 に示す．このリダイレクトを発生させている Web サイトについてコンテンツ解析を行う．

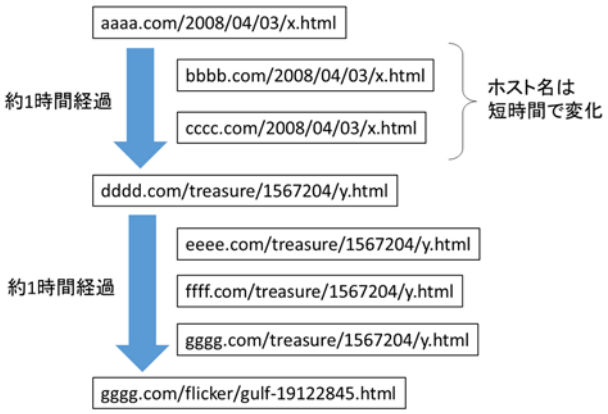

図 5　リダイレクト先 URL の遷移例

5.3　コンテンツ解析

先述のリダイレクトを発生させている 17 個の Web サイトの html ファイルを解析したところ，17 個全ての Web サイトに iframe タグが挿入されていた．次に，リダイレクト先 Web サイトを解析しようとしたが，全ての URL で応答が 404 Not Found となっており，これ以上コンテンツ解析を進めることができなかった．VirusTotal でリダイレクト先 URL を分析してみると，検出率は少なくとも 2/68 で，最

大で 6/68 であった．指定した URL を解析する urlquery.net[10]で，リダイレクトを発生させている Web サイトを解析させたところ，iframe によるリダイレクトを検出することができなかった．また，同様のサービスである aguse.jp[11]で同じ URL を解析させても，iframe タグを検出することはなかった．そこで試しに解析環境上の Firefox と Chrome のブラウザを用いて当該サイトにアクセスしてみたところ，図 6 のように iframe タグが消えていた．また，その直後に同じ Web サイトに Internet Explorer でアクセスしたところ，図 7 のように iframe タグが挿入されていることが分かる．このことから，Internet Explorer で解析対象の URL にアクセスした場合のみ，iframe タグが挿入され，外部のサイトへリダイレクトされていることが分かる．aguse.jp の仕様は把握できていないが，urlquery.net では Firefox のブラウザを用いて Web サイトの解析を行っていたため，iframe によるリダイレクトを検出できなかったと考えられる．

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN" "http://www.w3.org/TR/xhtml
1/DTD/xhtml1-transitional.dtd">↓
<html xmlns="http://www.w3.org/1999/xhtml" xm
l:lang="en-gb" lang="en-gb">↓
<head>↓
↓
```

図 6　Firefox や Chrome でアクセスした場合

```
<span style="position:absolute; top:-1068px;
width:312px; height:302px;">↓
<iframe src="http://[redacted]
co.uk/many/poet-nowhere-18907531" width="255"
 height="269"></iframe>↓
</span>↓
<noscript>↓
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01/
/EN" "http://www.w3.org/TR/html4/strict.dtd">↓
<html xmlns="http://www.w3.org/1999/xhtml" xm
l:lang="en-gb" lang="en-gb">↓
```

図 7　Internet Explorer でアクセスした場合

5.4 考察

5.2 節の通信ログ解析，5.3 節のコンテンツ解析の結果から，解析を行った 24 個の Web サイトの内，17 個の Web サイトが悪性 Web サイトへと誘導する入口サイトであると考えられる．iframe によって挿入される Web サイトの解析を行うことはできなかったが，VirusTotal での分析結果や URL の変化などを考慮すると，おそらく攻撃サイトなどの悪性 Web サイトが不正に挿入されていると考えられる．また，ブラウザが Internet Explorer の場合のみ iframe が挿入されていることから，おそらくその Web サイトを提供している Web サーバが疑似 Darkleech[12][13]に感染していると思われる．Darkleech[14]とは，2012 年頃から始まった攻撃で，Apache サーバにのみ感染し，攻撃者がリモートから悪質な Apache モジュールをアップロードできる状態にしてしまう．この悪質なモジュールによって感染先 Web サイトに不正な iframe を挿入し，悪性 Web サイトへと誘導させる．これに対して疑似 Darkleech とは，WordPress や Joomla!を使用している PHP ファイルを改ざんすることで，感染先 Web サイトに不正な iframe を挿入させる攻撃である．入口サイトとなっている 17 個の Web サイトについて調べると，全て PHP ファイルを使用しており，さらに Joomla!や WordPress，Drupal などのコンテンツマネジメントシステムを使用していることが分かったため，疑似 Darkleech に感染している可能性が高いと考えられる．改ざんされた PHP ファイルによって，アクセスしてきた端末のブラウザ情報を調べ，Internet Explorer の場合のみ，iframe を挿入した Web サイトを返答していると思われる．リダイレクト先 URL は Exploit Kit によって指定されていると考えられる．考察を交えた解析結果を図 8 に示す．

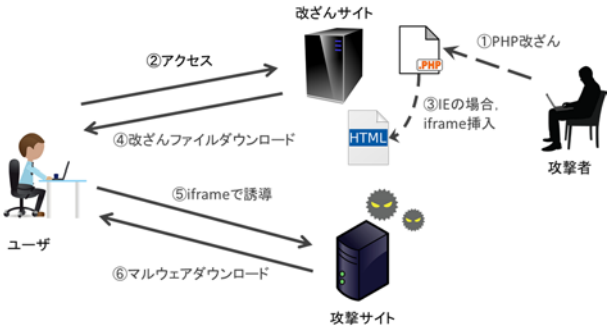

図 8　疑似 Darkleech による感染の流れ

6. まとめ

本稿では，環境によって挙動を変える悪性 Web サイトに対応するために，マルチ環境解析を用いて悪性 Web サイトの分析を行った．また，解析作業の効率化を図るために，通信ログ解析の一部を自動化させた．iframe タグが難読化によって挿入されている場合でも対応できるように，HTTP リファラとホスト名に着目した解析プログラムを構築した．Malware Domain List に記載の Web サイトに対してマルチ環境解析を行ったところ，Web サイトの多くが既に削除されていたこともあり，悪性 Web サイトの解析を行うまでに至らなかった．しかし，urlquery.net などの Web サイト解析サービスでは検知できなかった不正な iframe タグを検知することができた．マルチ環境解析の結果から，当該サイトの改ざんは Darkleech によって行われた可能性が高いと判断することができた．今後の課題として，マルチ環境解析の自動化をさらに進め，より多くの Web サイトを解析することなどが挙げられる．

参考文献

[1] SmartScreen フィルター – Microsoft Windows, http://windows.microsoft.com/ja-jp/internet-explorer/products/ie-9/features/smartscreen-filter, (参照 2016-08-12).

[2] Safe Browsing API，https://developers.google.com/safe-browsing/，(参照 2016-08-12).
[3] 義則，篠田，神薗，廣友，毛利，白石，岩田，“マルチ環境解析を利用した悪性 Web サイトアクセスが及ぼす影響の分析支援，”信学技報，ICSS2013-85，pp.161-166，Mar.2014.
[4] 西尾，廣友，福田，毛利，白石，“マルチ環境解析を用いた悪性 Web サイトの分析について，”信学技報，ICSS2016-11，pp.57-62，Jun.2016.
[5] Dell SonicWALL，Blackhole Exploit Kit: Rise & Evolution，http://software.sonicwall.com/gav/Blackhole%20Exploit%20Kit%20-%20Rise%20&%20Evolution.pdf，2011.
[6] VirusTotal，https://www.virustotal.com/ja/，(参照 2016-08-12).
[7] SeleniumHQ Browser Automation，http://www.seleniumhq.org/projects/webdriver/，(参照 2016-8-12).
[8] Malware Domain List，http://www.malwaredomainlist.com/，(参照 2016-08-12).
[9] no|wrap.de – Flare，http://www.nowrap.de/flare.html，(参照 2016-08-12).
[10] urlquery.net - Free URL scanner，http://urlquery.net/index.php，(参照 2016-08-12).
[11] aguse.jp: ウェブ調査，https://www.aguse.jp/，(参照 2016-08-12).
[12] 改ざんの標的となる CMS 内の PHP ファイル(2016-02-25) – JPCERT コーディネーションセンター，https://www.jpcert.or.jp/magazine/acreport-cms.html，(参照 2016-08-12).
[13] WordPress Malware Causes Psuedo-Darkleech Infection - Sucuri Inc，https://blog.sucuri.net/2015/03/pseudo-darkleech-server-root-infection.html，(参照 2016-08-12)
[14] こうして Web は改ざんされた (2):Gumblar から Darkleech Apache Module まで,巧妙化の足跡 - @IT，http://www.atmarkit.co.jp/ait/articles/1308/05/news002.html (参照 2016-08-12)