Vol.2014-DPS-158 No.41
Vol.2014-CSEC-64 No.41
2014/3/7

情報視覚化によるDrive-by Download攻撃対策の一検討

尼子 雄大[1,a)]　高田 哲司[1,b)]

概要：Web ブラウザやプラグインの脆弱性を利用しマルウェアを感染させる Drive-by Download(DbD) 攻撃による被害が深刻化している．DbD 攻撃の特徴として，Web ページの訪問者にマルウェアを感染させることを目的としている為に，ユーザや管理者が感染に気づきにくくなっている．DbD 攻撃を認知するには，これまではログの監視や HTTP のヘッダ，リダイレクト情報に着目することが主であった．DbD 攻撃への対策として，Web トラフィック着目した情報視覚化によるマルウェア感染認知支援システムを提案する．DbD 攻撃に特徴的なトラフィックの視覚化によって，ユーザまたは管理者にマルウェア感染を認知させ，対処を促すことを目的とする．

キーワード：情報視覚化, Drive-by Download 攻撃

The Network Visualization Tool for detecting the Drive-by Download attacks.

Amako Katsuhiro[1,a)]　Takada Tetsuji[1,b)]

Abstract: The Drive-by Download(DbD) attack, which is one of the intrusion method of malware, is a now major threat to the Internet. Detecting the DbD attack is difficult for administrators since there are no changes on the screen. A conventional approach for these attacks mainly utilizes the logs based on character information or focus attention on the http header or redirection. We propose the network visualization tool for detecting the DbD attacks. The proposed network visualization tool makes users and administrators to take action for malware.

Keywords: Visualization, Network Security, Drive-by Download

1. はじめに

Drive-by Download(DbD) と呼ばれる攻撃が猛威を振るっている．DbD 攻撃とは，マルウェアを感染させる攻撃手法の一つであり，IBM SOC レポート [1] によれば，このDbD 攻撃の検知件数は，2012 年下半期の 956 件に比べて2013 年上半期は 3,972 件と，約 4.2 倍へと急増している．またこれら DbD 攻撃に対し，Web サイト管理者のみならず，システム管理者やネットワーク管理者にも DbD 攻撃によるマルウェア感染の対処が必要とされる．一方で次のような理由から，システム管理者やネットワーク管理者の対処が限定的なものとなっている．

- 正規の Web サイトが改ざんされ，DbD 攻撃に用いられており，従来の危険な・不確かな Web サイトにアクセスしないという対策では通用しない
- Web サイト閲覧時にユーザが視覚的に認知することなくマルウェアがインストールされる為に，感染に気づけない
- 最新のソフトウェアを利用していても，ゼロデイ攻撃を用いた DbD 攻撃を防ぐことは出来ない

これら DbD 攻撃は，個人ユーザだけではなく，特定の組織や団体をターゲットととした標的型攻撃にも使用されており，攻撃への対策が望まれている．

DbD 攻撃には，次の 3 つの役割を持つサイトから構成

[1] 電気通信大学
Uniersity of Electro-Communications, Tokyo, Japan
[a)] amako.k@uec.ac.jp
[b)] zetaka@computer.org

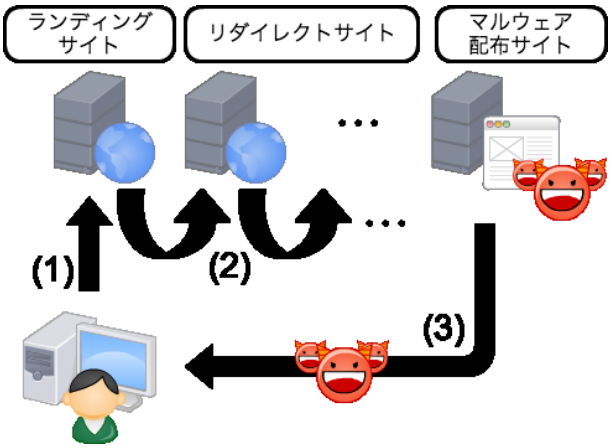

図 1 DbD 攻撃の概要

されていると言われている [4].

- **ランディングサイト**
  マルウェア配布サイトへと転送される，攻撃者によって改ざんされた正規の Web サイト．信頼されたページビュー (PV) の多い Web サイトを攻撃者が狙う傾向にある．
- **リダイレクトサイト**
  ランディングサイトとマルウェア配布サイトをリダイレクトさせて結ぶ役割を持った Web サイト群．
- **マルウェア配布サイト**
  マルウェア本体を設置している Web サイト．

これら 3 つのサイトを用いて攻撃者は，次のようにしてマルウェアをユーザに感染させる (図 1).

(1) ユーザが，改ざんされた正規 Web サイトを閲覧する．
(2) Web サイトに埋め込まれた悪意のある Web サイトへのリダイレクトサイト群へ転送される．
(3) ユーザの計算機環境の脆弱性を利用し，悪意のある Web サイトからマルウェアがダウンロードされ，インストールおよび実行される．

以上の動作がすべてバックグラウンドにおいて行われ，セキュリティ知識の乏しいシステム管理者やネットワーク管理者には，DbD 攻撃の認知は困難である．

本研究では，DbD 攻撃の一連の通信に見られるリダイレクト (転送処理) とマルウェア感染に特に用いられているファイルのダウンロードに着目した．DbD 攻撃では，マルウェアのダウンロードに至る過程において，複数のリダイレクトが発生しており，正規のサイト閲覧と異なる通信挙動を示す．さらに，これら通信挙動とマルウェアに感染に特に用いられているファイルのダウンロードを，ユーザに視覚化して提示する．DbD 攻撃の一連の流れを視覚化情報としてユーザに提示し，DbD 攻撃によるマルウェア感染の認知を支援する．

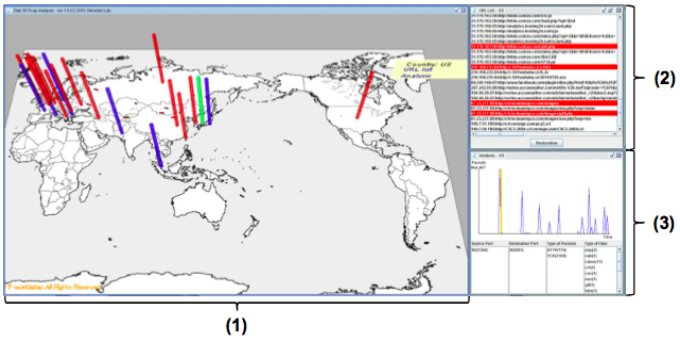

図 2 義則ら [2] の提案した “Flow Visualizer”

2. DbD 攻撃の対策手法と既存研究

2.1 ブラックリストによる DbD 攻撃対策手法

既存の DbD 攻撃対策手法として，URL ブラックリスト方式が挙げられる．現在複数の URL ブラックリストが運用されており，代表的なものとして，Google Safe Browsing[6] や Mcfee SiteAdvisor[7] が挙げられる．これら URL ブラックリスト方式の採用により，ユーザは DbD 攻撃を受ける悪意のある Web サイトへのアクセス時に警告画面を表示することで，DbD 攻撃を回避可能となっている．しかし，これら URL ブラックリスト方式は，Google 社や Mcfee 社がクローリングを行い危険性を判断し，ブラックリストに登録されることになる．したがってクローリングによって，ブラックリストに登録が行われるまで，警告が発せられず対処できない問題点がある．また DbD 攻撃は，特定の IP アドレスを攻撃の対象とした標的型攻撃にも利用されており [8]，DbD 攻撃サイトがブラックリストに登録されない可能性が存在する．

2.2 既存研究

2.2.1 DbD 攻撃通信の可視化システム Flow Vizulizer

義則ら [2] の提案した可視化システムでは，通信の可視化に “Flow Visualizer” を提案している (図 2).

図 2 の (1) では，ハニーポット中で収集した DbD 攻撃サイトの通信データを国別に分け，世界地図画面にプロットし，さらに通信頻度で色分けを行い，表示している．(2) では，ハニーポットがクローリングした一連の URL を示している．(3) において，ハニーポットがクローリング対象の URL にアクセスした場合に発生するトラフィックの通信量と時間変化を表している．本可視化システムの対象者であるセキュリティ技術者は，これら (1),(2),(3) の視覚的に提示された情報を用いて，ハニーポットで収集した通信データに含まれている DbD 攻撃に使用されたマルウェアの動的解析まで行うことができる．

2.2.2 Gumbler に感染した PC の可視化

金子ら [3] の提案した可視化システムでは，Gumbler に

感染した PC の挙動を観察するために可視化システムを提案している．仮想環境上の PC に Gumbler に感染させ，その後用意したハニーポットである FTP サーバにアクセスすることによって，攻撃者をハニーポットへ攻撃するよう誘導する．攻撃者によるハニーポット上の FTP サーバーへの改ざん行為を監視し，(1) 発生した通信トラフィックと (2) 攻撃者の FTP サーバー上での (攻撃者によるファイルの読み込みと書き込み等の) 挙動の分析を行う．これら分析結果を，世界地図上にマッピングして可視化を行っている．

2.2.3 検知を目指した不正なリダイレクトの分析

寺田ら [5] は，DbD 攻撃のリダイレクトに着目し，アクセス履歴の特徴に明らかにした．アクティブ型ハニーポットによって収集した DbD 攻撃サイトの巡回データとマルウェアが含まれているデータセットを用いている．DbD 攻撃のリダイレクト動作によるアクセス遷移を抽出し，アクセスが生じる理由となったリクエストを親リクエストとして，アクセスの相互関係を分析している．その結果，(1)HTTP リダイレクトが全体として少なく，(2) 親リクエストによる応答に含まれない URL へのアクセスがマルウェアである可能性があること，(3)PDF ファイルは，短いアクセス遷移であるのに対して，SWF ファイルやバイナリファイルは長いアクセス遷移である傾向を示した．また，危険なファイルのダウンロードの URL を機械学習で判定するロジックを機械学習を用いて抽出している．

3. 研究目的

1 章で述べたように，DbD 攻撃の脅威はますます高まっている一方で，セキュリティ技術者ではないシステム管理者やネットワーク管理者が DbD 攻撃を認知するには，主に URL ブラックリスト方式による防御策に限られていた．しかし，DbD 攻撃にみられるマルウェアの侵入や感染し活動を行う過程において，マルウェアの存在や感染をユーザに認知しにくい手法が主流となっている現在において，これら既存対策手法では不十分である．本研究の目的は，セキュリティに詳しくないが基本的なシステムの管理を行えるシステムまたはネットワーク管理者に対して，DbD 攻撃に特徴的なネットワークトラフィックを視覚化することによって，DbD 攻撃の発生や DbD 攻撃によるマルウェア感染の認知を支援することである．

既存の視覚化システムでは，義則ら [2] の実装した “Flow Vizulizer”や金子ら [3] の提案した可視化システムを挙げた．しかしこれらシステムは，前者はセキュリティ技術者がハニーポット等を用いて収集した DbD 攻撃のトラフィックデータを解析することに主眼が置かれたシステムである．さらに後者も，セキュリティ技術者がハニーポットを用いて Gumbler による攻撃を収集したトラフィックデータに対する可視化システムである．両者ともに，DbD 攻撃のユーザやシステム管理者による認知を目的とする本研究に目指す点が異なっている．

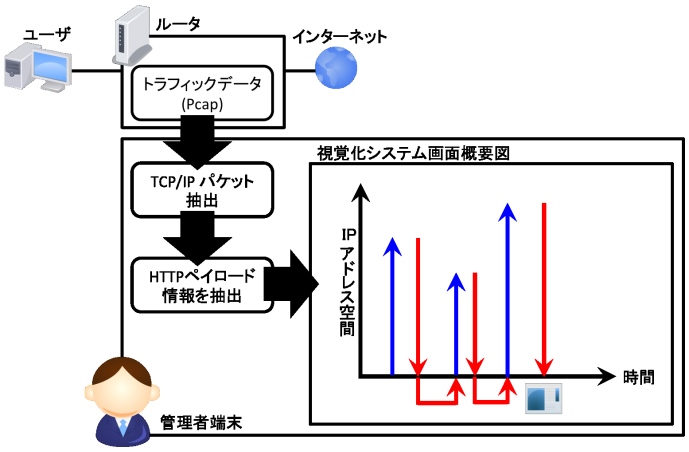

図 3 提案視覚化システムの概要と構成

また寺田ら [5] の DbD 攻撃の分析で，DbD 攻撃の特徴を分析し機械学習による判定機能を抽出しているが，常に誤検知という問題がつきまとう．本研究では，人による思考判断を支援し，攻撃を認知することが主眼であり，機械学習による判定を補完すると考えられる．

4. 提案視覚化システム

本章では，提案視覚化システムについて述べる．3 節の目的である視覚化システムには，DbD 攻撃に特化した視覚化システムを実装するにあたって，次のように DbD 攻撃の特徴をとらえた視覚化が必要であると考えた．

- マルウェアの可能性の低いファイルのダウンロード通信の非表示
- Drive-by Download 攻撃に特徴的なリダイレクトを表示
- HTTP 通信における IP アドレスの偏在性の提示

ネットワークトラフィックをすべて視覚化してしまうと，視覚化画面を多数のトラフィックによって重なりあい，必要な情報が見えなくなってしまう「隠れの問題」が発生する．したがって，DbD 攻撃でマルウェアの可能性の高いファイルのダウンロードを識別し，それのみをユーザに提示する．詳細については，後述する．また，DbD 攻撃には，マルウェアの配布サイトへと誘導される過程にリダイレクトが発生する特徴 [4] を利用し，リダイレクトの流れを視覚化しユーザに提示する．さらに，リダイレクトサイトやマルウェア配布サイトは，国を跨いで分散されたサーバー上にホストされる場合が多く，IP アドレスが広く分布する．リダイレクトの提示と併用することによって，DbD 攻撃に特徴的なリダイレクト動作を提示する．

4.1 視覚化システムの概要と構成

提案する視覚化システムの構成図を，図 3 に示す．

システムは，入力にネットワークトラフィックのキャプチャーファイル (pcap) を用いている．これは，図 3 のように，ユーザの端末とインターネットの中間にあるパケット転送装置上で収集するシステム管理者を想定した．入力の Pcap は，TCP/IP 通信のみを抽出 (フィルタリング) を行い，TCP パケットの再構築処理を行った後，HTTP トラフィックのみを抽出する．HTTP トラフィックには，HTML や動画像等のリソースを要求する，HTTP 要求と，要求に答え各リソースを返す HTTP 応答がある．HTTP 要求と HTTP 応答が判別出来る状態になるまで，TCP パケットを再構築する下処理を行う．

パケットが再構築され HTTP トラフィックが解釈できる下処理をした後，各 HTTP 要求と HTTP 応答を関連付け，HTTP 応答に着目して視覚化を行う．キャプチャデータの読み込みから，HTTP トラフィックの再構築までの下処理部を，Python3.3 と pcap ライブラリである pypacker を用いて実装した．視覚化部では，Java を用いて下処理したデータを用いて表示を行う．

4.2 視覚化方法

実際の視覚化画面を図 4 に示す．視覚化画面では，縦軸に **IP アドレス空間**を，横軸に**時間軸**をとっている．IP アドレス空間は，縦軸最上部が IP アドレス “0.0.0.0” を表し，最下部が “255.255.255.255” を表し，線形に割り当てている．横軸は，pcap ファイルの起点時刻を基準に，30 秒間の通信を表示し，30 秒以降のデータは，スクロールバーでスクロールして全通信内容を視覚化して見ることが出来る．

HTTP 要求を送信する計算機を最下部の時間軸を起点に，HTTP 要求の宛先 IP アドレスをマッピングして，**青い矢印**で表す．HTTP 応答は，HTTP 応答の送信元 IP アドレスにマッピングし，**赤い矢印**で表す．HTTP 要求と HTTP 応答は，基本的に一対一対応となるが，サーバーが存在しない場合や応答しない場合には，HTTP 要求の矢印の上部にバツの印が表示され，HTTP 応答が無かったことを表す．HTTP 応答で受信したデータを，ヘッダ部の Content-Type を用いて，特に DbD 攻撃で多用される，以下の要注意であるファイルのアイコンを表示している．

- Adobe Flash Player の脆弱性を突く “Small Web Format 形式” のファイル
- Adobe Reader の脆弱性を突く “Portable Document Format 形式” のファイル
- Oracle Java の脆弱性を突く “Java Archive 形式” のファイル
- Windows 環境の実行ファイルフォーマットのファイル

また感嘆符記号のアイコンは，HTTP 通信においてリダイレクトの発生を表している．

4.3 隠れの問題と JavaScript 難読化への対処

すべての HTTP 要求と HTTP 応答を表示すると，図 5 のように表示され，必要な情報をユーザが得ることができなくなる問題が発生する．これを隠れの問題と呼ぶことにする．しかしすべての通信を表示するのではなく，DbD 攻撃と疑わしいと思われる通信をユーザに提示することで，隠れの問題を解消しようと試みた．

そこで，次のように表示する HTTP 要求と HTTP 応答の組を選別した．

(1) HTTP 応答コードが 300 番台のリダイレクトである HTTP 応答とその HTTP 要求の組の表示
(2) 前述した要注意であるファイルのダウンロードに際し，そのファイル名が過去 1 分間に通信した内容に含まれていない HTTP 要求と HTTP 応答の組の表示

(1) のリダイレクトのみを用いた表示では，リダイレクトサイトからマルウェア配布サイトへのリダイレクト動作が視覚化できない．これはリダイレクトサイトからマルウェア配布サイトに誘導する過程またはマルウェア配布サイトに誘導された後において，JavaScript による動的な攻撃コードの生成やバックグラウンド通信機能を用いてマルウェアをダウンロードさせている為に，HTTP の 300 番台におけるリダイレクトが発生しない．

そこで，JavaScript による動的な攻撃コードそのもの自体が難読化されていることに着目した．難読化とは，プログラムのソースコードを人間に理解しにくい形へと変換されたソースコードであり，主な目的がソースコードの解析を妨害することや DbD 攻撃の検知を逃れる為である．本提案視覚化システムでは，JavaScript の難読化を解析したり解くというアプローチではなく，JavaScript による攻撃コードは難読化されていることを利用した．動的にマルウェアをダウンロードさせる JavaScript の攻撃コードによってダウンロードされるならば，マルウェアと疑わしいファイルのダウンロードが発生した際に得られる URL 情報からファイル名を抽出し，そのファイル名が過去 1 分間に受信した HTTP 応答のどの部分にも含まれない場合は，マルウェアのダウンロードではないかと仮定し表示することとした．

5. 視覚化事例

5.1 正規利用者による Web アクセスの視覚化事例

正常なネットワークトラフィックの視覚化事例として，CiNii[9] へアクセスし，学術論文 PDF ファイルをダウンロードするまでの一連の流れのトラフィックデータを収集した．本提案視覚化システムで視覚化を行うと，4.3 節で述べた視覚化の条件に当てはまらないため，本来は視覚化画面に表示されないが，本節では異常な視覚化の例と対比するために，すべての通信を表示するように設定している．視覚化を行った画面を，図 6 に示す．

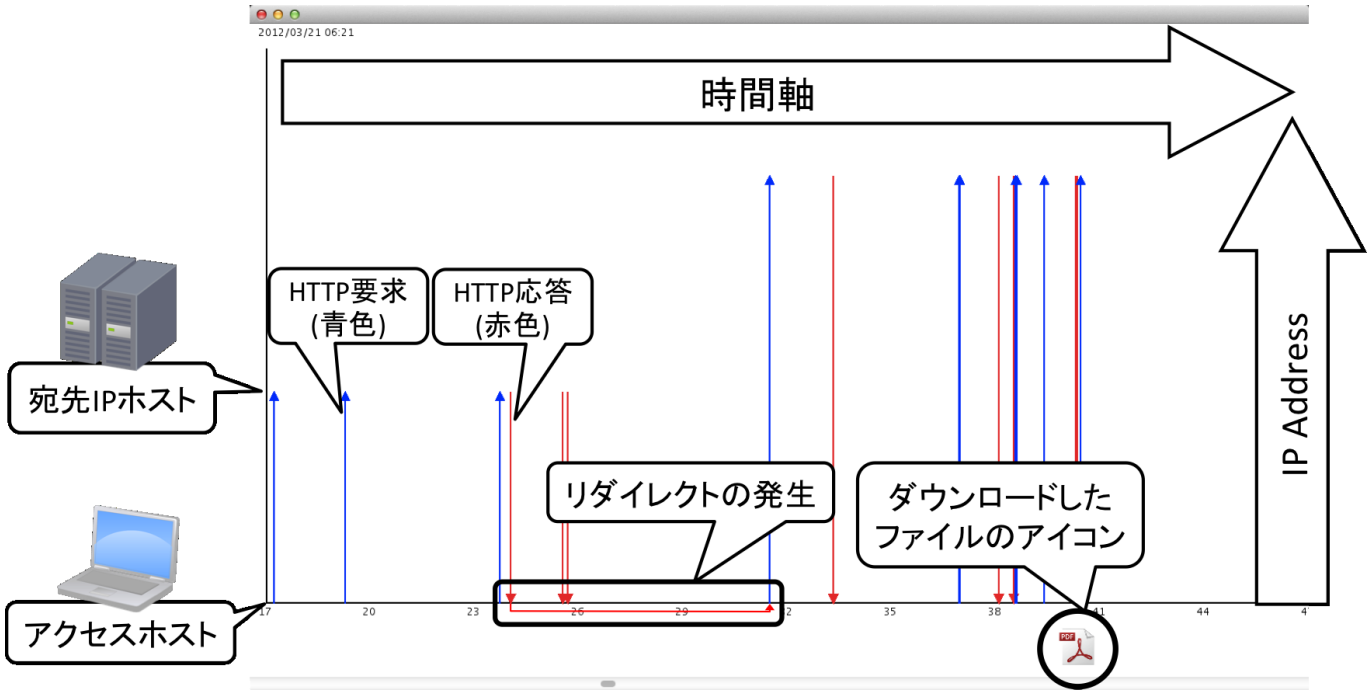

図 4 提案視覚化システムの視覚化画面の概要

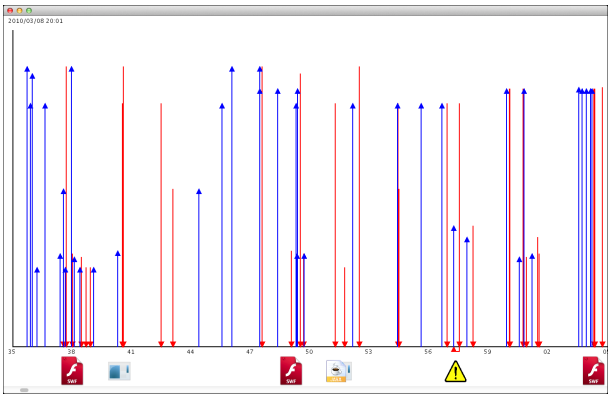

図 5 隠れの問題

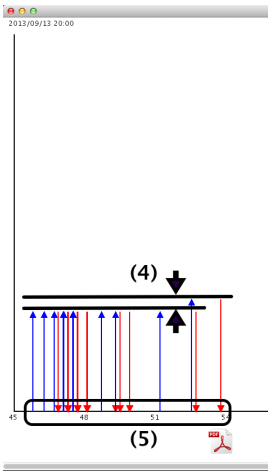

図 6 CiNii への正常なアクセストラフィックによる視覚化事例

図 6 の視覚化画面は，以下の理由から正常な通信であると判断できる．

- 図中の (4) において，アクセス先の IP アドレスが 2 つあることが見て取れる．またその 2 つの IP アドレスは，縦軸上へのマッピングの関係から IP アドレスの値を基にした距離で考えると，比較的近い値を持つ 2 つのサーバにアクセスしていると考えられる．
- 図中の (5) では，PDF のダウンロードに際し，リダイレクトが全く発生していないことが見て取れる．

以上のことから，図 6 の通信は，PDF ファイルのダウンロードはあるものの，これは Web 閲覧者による能動的な行為の結果であり，DbD 攻撃によるマルウェアの感染事例ではない可能性が高いことが視覚化表示から理解できる．

5.2 Malware Dataset を用いた視覚化事例

Drive-by Download Dataset by Mrionette(D3M)[10] は，DbD 攻撃に特化したデータセットである．NTT セキュアプラットフォーム研究所が開発した Marionette と呼ばれるアクティブ型ハニーポットによって，収集した DbD 攻撃サイトの巡回データとマルウェアが含まれているデータセットである．

図 7 に，D3M データを本提案視覚化システムで視覚化した事例を示す．図 7 中の (6) の表示部分では，横軸方向にわかりやすく補助線を引いている．20 時 02 分 09 秒頃に，ユーザが PDF ファイルがダウンロードされていることを示している．ダウンロードされた PDF ファイルを入手する直前に，不審なリダイレクトが発生していることも図 7 中 (7) の表示部分から見てとれる．

HTTP リダイレクトは，通常ドメイン名の変更であったり，サーバー内のフォルダ構成の変更でのアクセス到達性

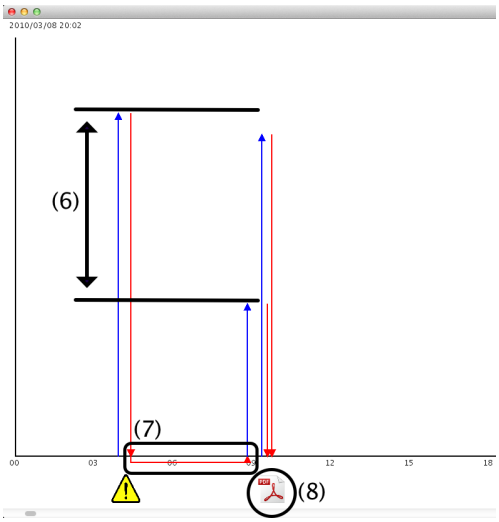

図 7 D3M データによる視覚化事例

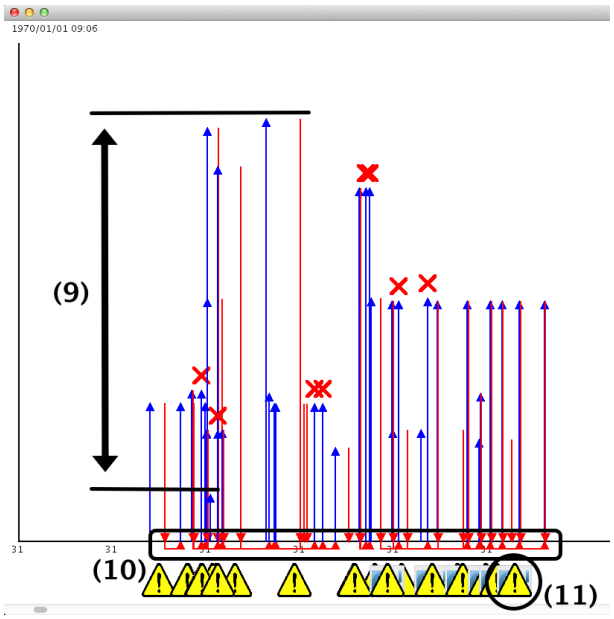

図 8 Web サイトの巡回による HTTP 通信データによる視覚化事例

を高めるために行われることが多い．したがって，通常のリダイレクト動作は，リダイレクト元とリダイレクト先の IP アドレスが，ネットワーク内において近傍である可能性が高く，つまり IP アドレスが近い可能性が高いと考えられる．しかし，図 7 の例では，HTTP リダイレクトにおいて (6) は IP アドレス空間での距離を表し，この場合において全く別のネットワークに存在していることを示している．

前述のような不審なリダイレクト (7) の直後に，PDF ファイルがダウンロード (8) されており，かつ前述のとおり過去 1 分間において PDF ファイル名が HTTP 応答内に存在していない．よって PDF ファイルは，リダイレクト後 JavaScript によって動的に URL が生成されアクセスがあったと推測できる．

5.3 収集データを用いた視覚化事例

本提案視覚化システムを用いて，セキュリティパッチの未適用である脆弱な OS とブラウザおよびそのプラグインを用いて，Web サイトを巡回し DbD 攻撃サイトと思われる HTTP 通信データを視覚化した事例を，図 8 に示す．

表 1 の仮想環境を構築し，urlquery[11] に投稿された DbD 攻撃サイトと思われる Web サイトにアクセスし，通信の内容をすべて収集した．urlquery[11] とは，Web でのマルウェアの検知および解析の為の Web サービスである．

視覚化画面を見ると，まず多数のリダイレクトが発生していることが，図 8 の (10) から読み取れる．また，多数のリダイレクトアイコンに隠れてしまっているが，実行ファイルのダウンロードが行われていることが，図 8 の (11) から読み取れる．リダイレクト先の IP アドレスは，図 8 の (9) から読み取れるように，広い IP アドレス帯域に分布している．

以上の視覚化画面から，DbD 攻撃であると認知し，キャプチャデータを詳細に調査を行った．結果，HTML ファイルに難読化された JavaScript コードを発見できたものの，マルウェアのダウンロードを見つけることは出来なかった．視覚化画面に実行ファイルのダウンロードが表示されていたファイルは，Content-Type が実行ファイルである “application/octet-stream” を示しているものの，HTTP プロトコルのリダイレクト応答という不自然なものであった (図 9)．

おそらく，攻撃コードが用いてる脆弱性を持っていなかったため攻撃が失敗したか，誘導先のサーバーがダウン

表 1 Web サイト巡回の仮想環境

OS	Ubuntu 12.04 LTS
CPU	Core i7-3770K
RAM	16GB
仮想マシンモニタ	Virtualbox 4.3.6
仮想マシン上 OS	Windows XP Professional SP3
ソフトウェア環境	InternetExplorer 6.0.2900.5512 Adobe Reader 9.0, Adobe Flash 11.1.102.55, Oracle Java Ver.6 Update 10 Apple QuickTime 7.6

```
HTTP/1.1 302 Moved Temporarily
Server: nginx
Date: Wed, 05 Feb 2014 09:55:17 GMT
Content-Type: application/octet-stream
Transfer-Encoding: chunked
Connection: keep-alive
P3P: CUR ADM OUR NOR STA NID
Location: http://xx.xxxxxxx.jp/yie/ld/gcs?v=zZH...

0
```

図 9 不自然な HTTP 応答の観測

していた為にマルウェアがダウンロードされなかったと思われる．

6. 考察と今後の課題

本提案視覚化システムでは，次の点を視覚化することが出来た．

- マルウェアの可能性があるファイルのダウンロード通信の非表示
- Drive-by Download 攻撃に特徴的なリダイレクト
- HTTP 通信における IP アドレスの偏在性

しかし一方で，次のような課題が残っている．まず，JavaScript 難読化によるリダイレクトの追跡と視覚化することは，本提案システムでは実現できていない．JavaScript の難読化を解析しその処理を把握し，リダイレクトを追跡するには，JavaScript の構文解析や仮想環境での仮想実行などの技術的手段を用いる必要性がある．また本提案システムでは，システム管理者やネットワーク管理者等比較的小規模なネットワーク上での使用を想定しているが，隠れの問題も通信量の増加によって，より顕著に問題となると考えられる．4.3 節で隠れの問題に，JavaScript の難読化によって表示量を低減することで対処したが，さらに通信量が増えると対処できなくなる可能性がある．

7. おわりに

DbD 攻撃は，ユーザに意図せずマルウェアを感染させる手法として，急速に被害を拡大しておりその対策が望まれている．

本稿では，DbD 攻撃の特徴の 1 つである「リダイレクト」に着目し，視覚的表現への変換を試み，DbD 攻撃と思われる通信を認知支援するシステムを提案した．本提案視覚化システムによって，システム管理者やネットワーク管理者は，これまでに加えてマルウェアの感染認知に気づきやすくなったと考えられる．

今後は，リダイレクト動作の関連付けをより明確すべく，JavaScript の難読化によるリダイレクトの追跡が出来ない問題や隠れの問題に対して，さらなる DbD 攻撃の特徴の調査を進め，マルウェアの感染をより認知しやすく改良を続けていきたい．

参考文献

[1] 2013 上半期 Tokyo SOC 情報分析レポート, IBM Security Services, 入手先〈http://www-935.ibm.com/services/jp/its/pdf/tokyo_soc_report2013_h1.pdf〉(参照 2014-02-13)
[2] 義則隆之, 伴拓也, 宮嵜仁志, ほか：通信可視化と動的解析の連携による攻撃解析支援, コンピュータセキュリティシンポジウム 2012 論文集, p.224-231,2012.
[3] 金子博一, 松木隆宏, 新井悠：通信トラフィックの分析による Gumbler 感染 PC の可視化, IEICE Technical Report, IA2010-1, ICSS2010-1, 2010.
[4] Van Lam Le, Ian Welch, Xiaoying Gao, Peter Komisarczuk, “Anatomy of drive-by download attack”, in *Proc.* AISC, 2013.
[5] 寺田剛陽, 古川忠延, 東角芳樹, 鳥居 悟, 検知を目指した不正リダイレクトの分析, 情報処理学会シンポジウム論文集, p.765-770, 2010.
[6] Google Safe Browsing, 入手先〈https://www.google.com/transparencyreport/safebrowsing/〉(参照 2014-02-13)
[7] Mcfee SiteAdvisor, 入手先〈http://www.siteadvisor.com/〉(参照 2014-02-13)
[8] JSOC INSIGHT 2013 vol.2, 入手先〈http://www.lac.co.jp/security/report/2013/11/06_jsoc_01.html〉(参照 2014-02-13)
[9] CiNii Articles - 日本の論文をさがす - 国立情報学研究所, 入手先〈http://ci.nii.ac.jp/
[10] 神薗雅紀, 畑田充弘, 寺田真敏, ほか：“マルウェア対策のための研究用データセット 〜MWS Datasets 2013〜”, CSS2013, 2013.
[11] URLQuery, 入手先〈https://urlquery.net/〉(参照 2014-02-13)