Vol.2013-DPS-154 No.29
Vol.2013-CSEC-60 No.29
2013/3/14

Drive By Download 攻撃に対する HTTP ヘッダ情報に基づく検知手法の提案

酒井裕亮[†1]　　佐々木良一[†1]

近年，WEB サイトを閲覧したユーザ PC にマルウェアを感染させることを目的とした Drive By Download 攻撃が様々な技術を複合的に用いることで高度化している．中でもスクリプトコードの難読化技術は，Drive By Download 攻撃の潜行化を実現しており，コードの特徴に基づいた検知は難しい可能性がある．そこで，難読化技術の影響を受けない HTTP ヘッダ情報に着目し調査を行い，Drive By Download 攻撃における検知の判断要素の一つとしての可能性を示し，検知手法の提案を行った．NTT セキュアプラットフォーム研究所より得た D3M の実験データを用い，実験を行った結果，検知率が 88%以上であることを確認した．

Proposal of detection method based on HTTP headers against Drive By Download Attack

HIROAKI SAKAI[†1]
RYOICHI SASAKI[†1]

Recently, Drive By Download Attacks to infect the user program which is used for browsing the Website have been more sophisticated by introducing complex techniques. Among them, script code obfuscation enables the attack undetectable. Therefore, conventional detection methods based on the script code becomes useless. Thus we have focused on the HTTP header which is not affected by the script code obfuscation. We conducted a survey of the characteristics of the HTTP header of Drive By Download Attack. In addition, we made a proposal the detection method based o the characteristics. Moreover, experimentation of the detection method using D3M data obtained from NTT Secure Platform Laboratory made appear that the detection rate is more than 88%.

1. はじめに

近年，WEB サイトを閲覧したユーザ PC にマルウェアを感染させることを目的とした Drive By Download 攻撃（以下 DBD 攻撃とする）が様々な技術を複合的に用いることで隠匿性や可用性が向上し高度化している．DBD 攻撃とは，主に Web ブラウザを通じて利用者に気づかれないように不正なプログラムをダウンロードさせる攻撃手法である．

この攻撃に対し，WEB サイトの管理者のみならず，ユーザ側においても何らかの対策を行う必要性がある．しかし、下記の様な理由からユーザ側での対策が困難になっている．

- マルウェア感染源が正規サイトである
- サイトを閲覧しただけでマルウェアに感染する可能性がある
- 視覚的な情報による改ざんの認識が難しい

このため先の研究で，実際の改ざんされた正規 WEB サイト及び挿入された不正スクリプトを調査・分析し，そこから得られた特徴に対して数量化理論２類を用い，改ざんサイト・不正スクリプトを判別するためのシステム，DICE（Detection of Injected Site using Cyber search Engine）を提案・試作開発し，精度や速度を実験により評価した[1][2].

しかし，近年の DBD 攻撃はコードの難読化技術を利用している場合が多く，先の研究を含めたコードの特徴に基づいた手法のみでは対応できない可能性がある．

本研究では，コードの難読化の影響を受ける可能性が低い HTTP ヘッダ情報に着目し，実際の DBD 攻撃の通信を調査することで得られた特徴を用いて，先の研究で提案されたシステムにおける追加機能として検知手法を提案し評価する．本稿では，2 章で DBD 攻撃及び WEB サイト改ざん攻撃について説明し，3 章で既存対策手法とその問題点を考察する．さらに 4 章で先行研究について述べ，5 章で先行研究の問題点と研究目的を説明し，6 章では DBD 攻撃の HTTP ヘッダ情報の調査とその結果を報告する．7 章では調査から得られた特徴を用いた提案方式の概要を説明し，8 章では提案方式による検証実験について述べる．

2. DBD 攻撃と WEB サイト改ざんについて

2.1 DBD 攻撃について

DBD 攻撃の流れを図 1 に示す．DBD 攻撃とは，一般の WEB サイトが SQL インジェクション等の方法で改ざんされ，そのページにアクセスすると（図 1，①），悪意のあるサイトに接続がリダイレクトされることにより誘導され（図 1, ②），マルウェアがダウンロードされ感染に至る（図 2，③）という攻撃手法である．

†1 東京電機大学
Tokyo Denki University.

Vol.2013-DPS-154 No.29
Vol.2013-CSEC-60 No.29
2013/3/14

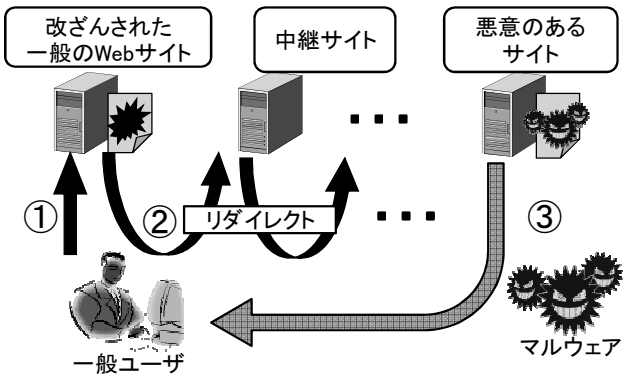

図 1　Drive By Download 攻撃
Figure 1　Drive By Download Attack

2.2 WEB サイト改ざんについて

2000 年に発生した中央省庁の WEB サイト改ざんに代表されるように, WEB サイトへの改ざん攻撃は以前から行われており，2012 年においても最高裁判所の WEB サイトが改ざんされる等近年でも発生している．これらの改ざん攻撃は，サイトデザインを改ざんすることで攻撃者自身の主張・メッセージを訴えることが目的であったといえる．しかしこのような目的とは異なり, WEB サイトを閲覧した一般ユーザの PC にマルウェアを感染させることを目的として，サイトに不正なスクリプトを挿入する WEB サイト改ざん攻撃が増加している．また，調査文献[3]によると近年の WEB サイト改ざん攻撃の検知件数に占める DBD 攻撃の割合が高い数値で推移しており, WEB サイト改ざんの目的が DBD 攻撃に起因していることがわかる．割合を表 1 に示す.

表 1　サイト改ざんの検知件数に占める DBD 攻撃の割合
Table 1　Percentage of the number of Drive By Download Attack that occupy the site tampering

2011 年 8 月	2011 年 9 月	2011 年 10 月	2011 年 11 月	2012 年 12 月	2012 年 1 月
22.8%	31.7%	40.7%	28.9%	31.5%	94.2%

ユーザを標的とした WEB ページの改ざんは，主に SQL インジェクションと呼ばれる手法を用いて，無差別かつ広範囲に行われている．SQL インジェクションとは，正規サイト上でデータベースと連携して運用されている WEB アプリケーションの脆弱性を突き，データベースを不正に操作することで，データベース内の情報の不正取得や改ざん等を行う行為である．ユーザを標的とした WEB ページの改ざんでは，この手法を用いて JavaScript や iframe 等のスクリプトが不正に WEB ページ内に挿入されている．

3. 既存対策手法について

3.1 ユーザ側における対策の必要性

DBD 攻撃及びそれに起因する WEB サイト改ざん攻撃が発生する根本的原因は, WEB サイト側の脆弱性にあるといえる．すなわち，WEB サーバの管理側で WEB サイトの脆弱性を排除することで，改ざん攻撃のリスクを軽減させることが可能である．しかし，全ての WEB サーバの管理側がこうした対策を行っているとは限らないため，ユーザ側でも何らかの対策をとる必要がある. WEB におけるユーザ側の簡易なマルウェア対策として「怪しい WEB ページにアクセスしない」,「怪しいファイルをダウンロードしない」といった，人の手による対策方法が挙げられる．しかし，ユーザを標的とした攻撃に対しては 1 章で述べた理由から効果が無くなる可能性がある.

このようなことから，ユーザ側において「機械的に DBD 攻撃及びそれに起因する改ざんサイト・不正サイトへのアクセスを防止する」という対策が必要となる.

3.2 ユーザ側における既存対策手法

「DBD 攻撃及びそれに起因する改ざんサイト・不正サイトへのアクセスを防止」という対策の具体的方法の例として, 危険なサイトに対して事前に警告を出す Google のセーフブラウジング機能[4]の活用や，不正サイトや不正スクリプトのブラックリストによるアクセス制限等が挙げられる. しかし，これらの対策方法には問題点がある．セーフブラウジング機能の警告は，Google のクローラが巡回した際に WEB サイトの危険性を識別して表示される. このことから, 改ざんが行われてからクローラが巡回するまでの期間は警告を出すことができない．不正スクリプトのブラックリストによるアクセス制限では，この問題に対処することが可能ではあるが，リストに未掲載の未知の不正スクリプトに対しては対処できないという問題点が挙がる.

そこで先の研究では，この「未知の不正スクリプト」に対処するための方法として，不正スクリプトにおける共通の特徴に対して数量化理論を用いる判定方法を考案し，そのシステムである「DICE」を提案・試作開発した．次章では先の研究について述べる.

4. 先行研究について

4.1 改ざんサイト自動検知システム DICE

先の研究で，一部組織内ネットワークのプロキシにおいて改ざんサイトの判定・検知を行い，通信の遮断もしくは警告を行うシステムである「改ざんサイト自動検知システム DICE」を提案した[2]．改ざんサイト自動検知システム DICE のシステム概略について図 2 に示す.

Vol.2013-DPS-154 No.29
Vol.2013-CSEC-60 No.29
2013/3/14

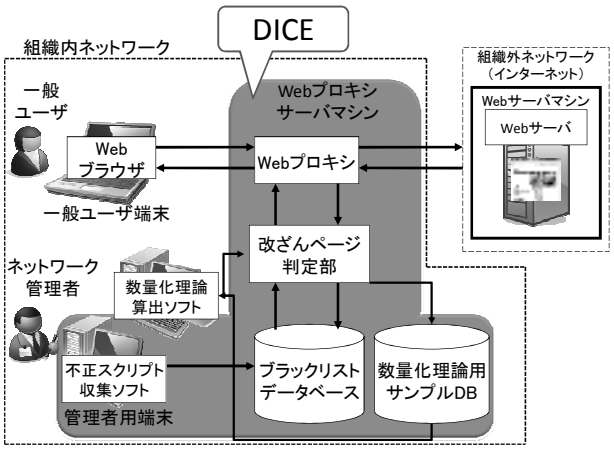

図 2　改ざんサイト自動検知システム DICE

Figure 2　Defaced Sites Automatic Detection System DICE

図 2 の組織内ネットワークの Web プロキシサーバマシン内におけるシステムの総称を DICE と名付けている．図 2 における改ざんページ判定部及び数量化理論 2 類についての説明, そして検証実験の結果について次節以降で述べる．

4.2 不正スクリプトの特徴

先の研究における調査では 1 次調査，2 次調査を通して下記の不正スクリプトにおける 5 つの特徴に注目すべきであるといった．

(1) **タイトルタグの改ざん**
(2) **スクリプトの多重挿入**
(3) **スクリプト名の偏り**
(4) **不正誘導先 URL の TDL の偏り**
(5) **スクリプト内の属性の設定・未設定**

先の研究で提案・試作開発された DICE は，これらの特徴に対して数量化理論 2 類を用いた検知方式となっている．

4.3 数量化理論 2 類

数量化理論とは，統計数理研究所出身の林知己夫教授らにより開発された日本独自のデータ分析手法である[5].

数量化理論 2 類は，2 つのタイプ A と B が不規則にあるとき，パラメータの係数を適切に設定することで，A は A 同士，B は B 同士で近い値を取るようにし，かつ A と B は遠い値を取るようにする．これにより，A と B の境界線の設定が可能となる．また，未知のデータに対しては，パラメータの係数の値を用いて，A，B どちらかに属する可能性が強いかを推定することが可能である．

4.4 数量化理論 2 類を用いた検知方式

数量化理論 2 類の適応にあたり，株式会社エスミ社のソフトウェア EXCEL 数量化理論[6]を利用している．

また，ここでは 4.2 で述べた 1 次調査，2 次調査で利用した調査プログラムを用いて収集した 350 件の URL の WEB ページを利用しており，Google Safe Browsing[5]を用いて調査したところ，350 件の URL の内 121 件の URL が危険な WEB ページであり，229 件の URL が安全な WEB ページであることがわかっている．

先の研究[2]ではパラメータの設定実験により最適なパラメータ数を 3 として設定し，検証実験により 4.2 で述べた(1), (3), (5), のパラメータの組み合わせを採用した． 設定実験と検証実験を組み合わせた結果を表 2 に示す．

表 2　パラメータ設定実験と検証実験の結果

Table 2　Result on survey of parameter setting experiment and verification experiment

		実際の結果		
		全体	悪性	良性
最適パラメータ	全体	350	121	229
	悪性	121	93	28
	良性	229	28	201

表 2 より，検知率は 84.0%，誤検知率は 16.0%としている．

5. 先行研究における問題点

先の研究における有効性の確認結果は，独自に収集した 2010 年以前の悪性ページを基としたデータに対するものである．最新データを用いて DBD 攻撃及びそれに含まれる悪性ページへの有効性を確認するため検証実験を行った．実験対象データは D3M 2012[7]及び D3M 2011 の攻撃通信データを使用した．

5.1 D3M について

D3M とは Drive-by-Download Data by Marionette の略称であり，NTT セキュアプラットフォーム研究所の高対話型 WEB クライアントハニーポット(Marionette)が収集した WEB 感染型マルウェアの観測データ群である．

D3M に含まれる攻撃通信データは, 公開ブラックリストに登録されている URL に対して WEB クライアントハニーポットが巡回を行った通信のフルキャプチャデータである．

5.2 DBD 攻撃に対する検証実験

DBD 攻撃では一般的に以下の様な 4 種類のファイルのダウンロードが危険とされている．

- PDF ファイル：Adobe Reader の脆弱性を利用
- SWF ファイル：Flash Player の脆弱性を利用
- Java ファイル：JRE や JDK の脆弱性を利用
- 実行ファイル

DBD 攻撃を防ぐ, または被害を最小限に抑えるためには上記の危険なファイルがユーザにダウンロードされる以前の段階での検知及び通信の遮断が求められる．そのため, 検証実験では攻撃通信データの各 URL の通信において, 危険なファイルが現れる以前の HTML ファイル等のデータに対して先の研究における提案方式を施行している．

D3M 2012 及び D3M 2011 における計 120 件の URL に対して検証実験を行ったところ，4.2 で述べた 5 つの特徴の

うち(1), (2), (3), (4)の 4 つの特徴が得られなかったため，数量化理論 2 類を適応することができず，有効な検知結果を得ることが出来なかった．

5.3 コードの難読化

特徴が得られなかった原因として，ほぼ全ての実験対象データにおいてコードの難読化が行われていたためである．

コードの難読化とは，プログラムのソースコードを人間にとって理解しにくいコードとすることであり，近年の DBD 攻撃において非常に多く利用されている．本来，コードの難読化は例として企業やプログラマ等が，自身が権利を持つソースコードにおける知的財産保護を目的とした手法の一つである．しかし DBD 攻撃等を画策する攻撃者にとっては目的が異なり，悪意のあるスクリプトコードをセキュリティ製品による検知を回避するためや，悪意のあるソースコードの目的を隠蔽するために利用されている．攻撃者側におけるコードの難読化の例を図 3 に示す．

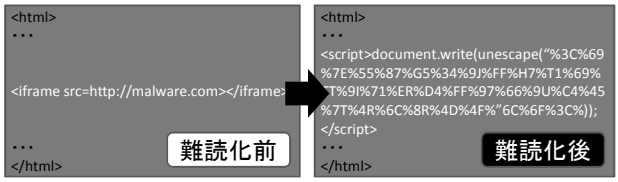

図 3　コードの難読化
Figure 3　Code Obfuscation

図 3 における難読化前のコードの文字列がブラックリストに登録されていたとしても，難読化によって全く異なる文字列となることで有効に機能しない可能性がある．

5.4 研究目的

先の研究で利用している特徴は全てコードの文字列に依存している．しかし，コードの難読化によって文字列に依存した検知は困難である可能性が高い．すなわち，先の研究における提案方式を含むコードの文字列に依存した検知手法は DBD 攻撃に対応出来ない可能性が高いといえる．そのため，コードの文字列に依存しない検知手法が必要であり，本研究の研究目的としている．

6. 調査

先の研究の問題点から，コードの難読化の影響を受け難い HTTP ヘッダ情報に着目した．本章では DBD 攻撃における HTTP ヘッダ情報の調査結果について述べる．

6.1 ヘッダ情報の特徴

DBD 攻撃の研究用データセットである D3M 2012 より得られた 40 件の URL の通信に対して HTTP ヘッダ情報の調査を行い，D3M 2012 の通信はレスポンスヘッダ内の X-Powered-By header に PHP のバージョン情報を高い割合で含んでいるという特徴を確認した．本来，X-Powered-By header は WEB サーバの管理側が非表示設定にすべきヘッダ情報であり，PHP は頻繁に脆弱性が報告されている．X-Powered-By header は利用する WEB サーバソフトウェアによって出力される文字列が異なり，例を以下に示す．

- Microsoft-IIS :『ASP.NET』に統一
- Apache や NginX 等 :『PHP/4.4.9』,『PHP/5.3.10』等

D3M 2012 において X-Powered-By header に PHP のバージョン情報を含む割合を表 3 に示す．表下段において含む割合を百分率で表し, 続いて(該当件数/対象件数)としている．

表 3　PHP のバージョン情報を含む割合
Table 3　Percentage containing the version of PHP

全体 URL 数（40 件）	Apache,NginX のみ（31 件）
72.5% (29/40)	90.3% (28/31)

一般の WEB サイト[8]の上位 300 件の URL に対して比較を行い，PHP のバージョン情報を含む割合に大きな差を確認した．含む割合の比較を表 4 に示す．

表 4　一般の WEB サイトとの比較
Table 4　Comparison with the general web sites

D3M 2012 の URL(40 件)	一般の WEB サイト	
	上位 100 件	上位 300 件
72.5% (29/40)	6.0% (6/100)	9.0%(27/300)

6.2 エクスプロイトキットのヘッダ情報調査

近年の DBD 攻撃はエクスプロイトキットの利用が主流となっている．エクスプロイトキットとは，様々な脆弱性攻撃を行う為の悪性ツールであり，コードの難読化の実現や専門知識を必要とせずとも利用できる等の特徴がある．

D3M 2012 の URL (40 件) を調査したところ, 65%の URL においてエクスプロイトキットを利用しており，該当するヘッダ情報を調査したところ 97%と非常に高い割合で PHP のバージョン情報が含まれていることを確認した．

そこで, D3M 以外におけるエクスプロイトキットの調査を行った．調査対象として，ブラックリスト[9]に 2012 年の 10 月から 11 月期に掲載された「Black Hole v2」,「Cool」の 2 種類のエクスプロイトキットの URL(各 10 件)を選択し，該当 URL を WEB ページ解析サービス[10]を用いてレスポンスヘッダの調査を行い，高い割合で X-Powered-By header に PHP のバージョン情報が含まれることを確認した．調査結果を表 5 に示す．

表 5　エクスプロイトキットの通信における PHP のバージョン情報を含む割合
Table 5　Percentage containing the version of PHP in communication to use the Exploit kit

調査対象	**Black Hole v2**	**Cool**
含む割合	90% (9/10)	100% (10/10)

6.3 PHP のバージョン情報が出力される原因

レスポンスヘッダに PHP のバージョンが含まれる原因としては，基本的にサーバの実質的管理者でないと出力されるヘッダを変更できないという点が挙げられる．ヘッダ

情報の出力設定は php.ini と呼ばれる設定ファイルからのみ変更が可能である．また，PHP のバージョンが 5.4 以降の場合には他の方法でも変更が可能ではあるが，D3M 2012 のバージョン情報を含む通信において 5.4 よりも古いバージョンが必ず含まれることを確認した．

そして，文献[11][12]によると，マルウェア等の危険なコンテンツをホストしている WEB サーバには下記の様な特徴がある．

- 古いバージョンの PHP やサーバソフトウェアを利用
- 攻撃活動が可能なフリーホスティングサービス

したがって，セキュリティの低下により攻撃を受ける可能性の高い WEB サーバが利用されている点が原因として考えられる．

7. 提案方式

本章では調査により得られた HTTP ヘッダ情報の特徴を利用した検知手法について述べる．

また，DBD 攻撃における先の研究と本研究の関係性を図 4 に示す．

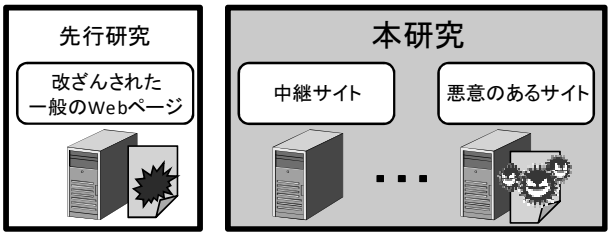

図 4　先行研究と本研究の関係

Figure 4　Relationship between The present study and preceding studies

先の研究では「改ざんされた一般の WEB ページ」を対象とし，本研究では「中継サイト」から「悪意のあるサイト」そして，マルウェアのダウンロードに至るまでを対象としている．

7.1 3 種類のヘッダ情報の利用

本提案手法は危険なファイルへのリクエストに対して検知及び遮断を目的としており，X-Powered-By header と Content-Type header を組み合わせ，Date header を用いてヘッダ情報の特徴と危険なファイルを関連付けることでリダイレクト等に対応している．利用するこれら 3 種類のレスポンスヘッダ情報を表 6 に示す．

表 6　利用するレスポンスヘッダ

Table 6　List of Response headers to be used

ヘッダ項目	含まれる情報
X-Powered-By header	PHP のバージョン情報等
Content-Type header	ファイルタイプ情報
Date header	レスポンスの発行時刻情報

DBD 攻撃では 5.2 で述べた 4 種類のファイルのダウンロードが危険とされている為，Content-Type header における判断要素として利用する．

また，DBD 攻撃の特徴として危険なファイルのダウンロードの通信において，リダイレクト元を特定するための Referrer header が含まれていない場合が多いが，Date header の時間間隔を用いることでリダイレクトの関連付けが可能であると考えた．

7.2 プロキシサーバの実装と遮断定義

先の研究との連携を考慮し，追加機能として同一環境上でヘッダ情報判定部をプロキシサーバ上に Perl 言語を用いて実装した．WEB プロキシにはフリーで公開されている Squid[13]を使用し，Squid の「url_rewrite_program」と呼ばれるオプションを利用している．このオプションにプログラムを指定すると，要求された URL を書き換え，WEB ブラウザに渡すことができる．ヘッダ情報判定部のシステムを図 5 に示す．

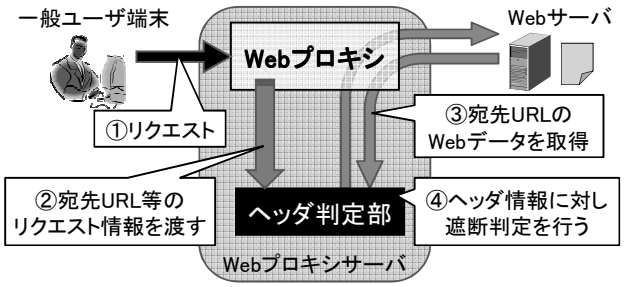

図 5　ヘッダ情報判定部

Figure 5　Subsystem performing the determination for HTTP headers

ヘッダ情報判定部の遮断定義を以下に示す．

A)　バージョン情報を含むファイルに対する遮断

同一レスポンスヘッダ内において，X-Powered-By header に PHP のバージョン情報が含まれており，かつ Content-Type header に危険とされているファイルタイプ情報が含まれている場合に危険とみなし URL の書き換えを行う．

B)　バージョン情報を含まないファイルに対する遮断

近年の DBD 攻撃に多く見られる Java ファイルは，同一レスポンスヘッダ内に X-Powered-By header を含まない傾向にある．また，その他の危険とされているファイルタイプにおいても，同様な傾向を調査段階で複数確認した．このようなファイルに対して遮断定義 A)の方法では対応できない．そこで，PHP のバージョン情報を含む HTML 等からの Date header の時間間隔を利用する．時間間隔以内に危険とされるファイルタイプ情報が確認された場合に危険とみなし URL の書き換えを行う．

時間間隔は D3M 2011 と D3M 2012(計 120 件の URL)より調査することで得られた傾向から，提案システムでは 10

秒以内として設定した. D3M 2011 と D3M 2012 から得られた Date header における時間間隔の傾向のグラフを図 6 に示す．縦軸は時間間隔（秒），横軸は対象の URL 数となっている．

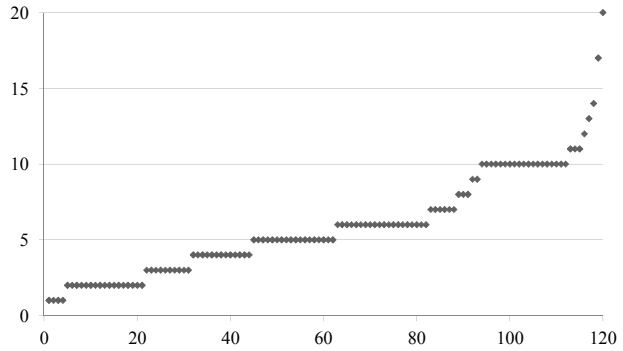

図 6　Date header における時間間隔の傾向
Figure 6　Time interval tendency of Date header

8. 検証実験

ヘッダ情報判定部における検知率の検証実験を行った．ここでは，本研究で提案する HTTP ヘッダ情報に基づいた検知手法のみにおける検証実験を行っている．

実験対象は悪性データとして D3M 2011 と D3M 2012 の通信データから計 120 件の URL，良性データとして一般の WEB サイト[8]から上位 230 件の URL を使用した．悪性データは，危険とされるファイルタイプのダウンロードを含んでおり攻撃が成功したとされる通信のみを対象とした．また，良性データとしての一般の WEB サイトは必ずしも良性であるという確証は無いが，ここでは全て良性データとして扱い，ヘッダ情報判定部が検知した場合は誤検知としている．検証実験の結果を表 7 に示す．

表 7　検証実験結果
Table 7　Result of verification experiment

			実際の結果	
		全体	悪性	良性
	全体	350	120	230
ヘッダ情報判定部	悪性	120	96	24
	良性	230	17	213

結果は検知率が 88.2%，誤検知率が 11.8%となった．

悪性データにおいて，Date header における時間間隔を 10 秒以上に設定することで検知可能である通信は増加する．しかし，仕様上厳密にリダイレクトを認識している訳ではない為，ユーザのアクセス（例としてユーザの故意の操作によるファイルのダウンロード等）による誤検知が発生する可能性がある．

良性データにおいて誤検知された原因を下記に示す．

(A)　対象 URL の HTML ファイルや，WEB ページ内に埋め込まれた外部のアクセス解析ツールに PHP のバージョン情報を含む

(B)　WEB ページ内に埋め込まれた外部のショッピングサイト等の広告 SWF ファイル

誤検知された 17 件の URL の内 16 件は，原因(A)から派生する原因(B)の通信であった．これらはアクセス解析ツール及び広告 SWF ファイルのホワイトリストにより解消される可能性がある．

9. おわりに

本稿では，近年の DBD 攻撃の特徴の一つであるコードの難読化の影響を受け難い HTTP ヘッダ情報に着目した．DBD 攻撃における通信の調査を行い，HTTP レスポンスヘッダに特徴を有していることを確認し，その特徴が検知の判断要素として有効である可能性を示すと共に検知手法を提案し，実験により基本的有効性を確認した．検出率を高めるため今後，さらに改良を図っていく．

参考文献

1) 田村佑輔, 甲斐俊文, 佐々木良一 : ユーザ標的型 Web サイト改ざんに対する検索エンジンを用いた検知手法の提案, 情報処理学会論文誌, Vol.51, p191-198
2) 田中達哉, 田村佑輔, 甲斐俊文, 佐々木良一 : 改ざんサイト自動検知システム DICE の開発と評価, 情報処理学会シンポジウム論文集, 2010, p531-536
3) セキュアブレイン : セキュアブレイン gred セキュリティレポート Vol.31
4) Google セーフブラウジング
http://www.google.com/safebrowsing/diagnostic?site=
5) 数量化理論 2 類とは
http://bstat.f7.ems.okayama-u.ac.jp/statedu/hbw2-book/nobel155.html
6) 株式会社エスミ
http://www.esumi.co.jp/
7) MWS 実行委員会 : 研究用データセット MWS 2012 Datasets について
http://www.iwsec.org/mws/2012/about.html
8) 日本の人気サイトランキング
http://akimoto.jp/japan/
9) Malware Domain List
http://www.malwaredomainlist.com/
10) urlQuery
http://urlquery.net/
11) Google : Google Technical Report Proves 2008 – All Your Iframes Point to Us
12) Microsoft : Microsoft Security Intelligence Report Vol.13
13) squid : Optimising Web delivery
http://www.squid-cache.org/