ユーザ参加型 Web セキュリティ観測システムにおける収集情報の網羅性に関する一考察

松中 隆志 †　　山田 明 †　　窪田 歩 †

†(株)KDDI 研究所
356-0003 埼玉県ふじみ野市大原 2-1-15
{ta-matsunaka, ak-yamada, kubota}@kddilabs.jp

あらまし ユーザの Web ブラウザにおいてマルウェアをダウンロード・実行させる Drive-by Download 攻撃が問題となっている．著者らは，多様なユーザの Web セキュリティを網羅的に観測するために，ユーザ参加型の対策フレームワーク (FCDBD: Framework for Countering Drive-By Download) を提案している．現在，本フレームワークの有効性を検証するために，約 100 人の承諾を得たユーザから Web アクセス履歴を収集している．本稿では，収集した Web アクセス履歴の分析結果を報告する．まず，ユーザ数が 100 人であっても主要 Web サイトの 91%が観測できることがわかった．また，全アクセス履歴の約 1/4 は，従来のクローリングが網羅しきれない Web サイトであった．さらに，本フレームワークによって，ユーザのブラウザおよびプラグインの更新実態を把握できることがわかった．

A Study on the Coverage of Obtained Data by the User-Participating Framework for Monitoring Web Security

Takashi Matsunaka†　　Akira Yamada†　　Ayumu Kubota†

†KDDI R&D Laboratories, Inc.
2-1-15 Ohara, Fujimino, Saitama 356-0003, JAPAN

Abstract Drive-by Download attack, which forces a web browser to download and execute a malware, is one of the most popular thread on the Web. The authors proposed the user-participating framework for countering drive-by download (FCDBD) which monitors a web security from several users' aspects. The authors are now collecting web access data from 100 users with the agreement for evaluating the feasibility of our framework. In this paper, we report the result of the analysis of collected web access data. First, we found that our framework can monitor 91 % of the major websites by 100 users. We found 1/4 of our collected data can monitor the websites which are hard to monitor by the existing web crawlers. We also realize that our framework can monitor the development of updated softwares (web browsers and plug-ins).

1　はじめに

Drive-by Download 攻撃は，Web 上における主要な脅威の一つである．この攻撃は，Web を利用してマルウェアを拡散する攻撃であり，ユーザは，攻撃が仕掛けられた Web ページにアクセスするだけでマルウェアに感染させられてしまう．Provos らの報告 [1] によると，Drive-by

Download 攻撃に係る悪性サイト (Exploit サイト，Distribution サイト) は生存期間が短く，発見や解析が非常に困難である．

Drive-by Download 攻撃の早期発見，検出および防御を目的として，著者らはフレームワーク (FCDBD: Framework for Countering Drive-By Download) を提案，実装した [2]，[4]．このフレームワークでは，ユーザが使用するブラウザおよび Web プロキシに観測センサを設置することで広域な観測網を構築し，ユーザの Web アクセスに関する情報を提供してもらうことで Web 上の Drive-by Download 攻撃に係る脅威をリアルタイムに把握する．そして検出された脅威の情報を観測センサに適宜フィードバックすることで，ユーザが攻撃の被害にあうのを未然に防ぐ．

著者らは本フレームワークの有効性を検証するために，約 100 人の承諾を得たユーザから Web アクセス履歴を収集した．本稿では，収集した Web アクセス履歴の分析結果を報告する．まず，ユーザ数が 100 人であっても主要 Web サイトの 91%が観測できることがわかった．また，全アクセス履歴の約 1/4 は，従来のクローリングが網羅しきれない Web サイトであった．さらに，本フレームワークによって，ユーザのブラウザおよびプラグインの更新実態を把握できることがわかった．

2 関連研究

Drive-by Download 攻撃サイトを発見，検出する手法の一つとして，Web クローラ (honeyclient) を用いた Web サイトの巡回 (クローリング) がある [5]，[6]．honeyclient で効率的に悪性サイトを検知するためには，クローリングの起点となる seed を適切に与える必要がある．また，攻撃者が，自身の悪性サイトの検出を防ぐために，セキュリティ関連企業，研究機関によるクローリングと思われるアクセスに対して正常の Web サイトのようにふるまう (cloaking) ような対策を行うこともあり，クローリングによる悪性サイトの発見，検知は非常に困難である．さらに，悪性サイトの生存期間は数時間程度と短命なため，その実態をリアルタイムに把握することは困難である．

悪性サイトを検出する手法として，Web ページ間の遷移関係の構造 (リンク構造) に着目して，未知の悪性サイトを検出する方法が提案されている．Zhang ら [7] の手法は，Drive-by Download 攻撃事例の HTTP トラヒック情報から悪性サイトのリンク遷移元元をたどり，URL の類似性などを考慮して複数の悪性サイトに共通のハブとなるサイト (central server) を検出し，そのサイトをもとに MDN(Malware Distribution Network) を検出する．そして，その central server の URL の特徴をシグネチャとして，既知の MDN に属する未知の悪性サイトを検出するものである．Stringhini ら [8] の手法は，リンク構造上の特徴に加えてユーザのマシンの環境 (OS，ブラウザ，プラグインなど) も加味して，特定のマシン環境のユーザのみが到達するリンクのパスを抽出することで，膨大な Web アクセスログから Drive-by Download 攻撃の悪性サイトを検出するものである．Wand ら [9] の手法は，[7] による MDN の検出の後に，Landing サイトの HTML の内容，URL の特徴を抽出して，未知の悪性サイトを検出するものである．進藤ら [10] は Drive-by Download 攻撃のリンク遷移におけるファイルタイプの変化から特徴を抽出して攻撃の有無を検知する手法を提案している．これらの手法は，新たな MDN の検出，URL，コンテンツなどの特徴の抽出のためには膨大な攻撃事例のデータが必要である．そのため，未知の MDN に属する Landing サイトなど悪性サイトの検出には即時的に対応できない．また，MDN の検出，特徴の抽出のための攻撃事例のデータ収集をいかに行うか，も重要な課題となる．

3 Drive-by Download 攻撃対策フレームワーク

3.1 概要

Drive-by Download 攻撃対策フレームワーク (FCDBD: Framework for Countering Drive-By

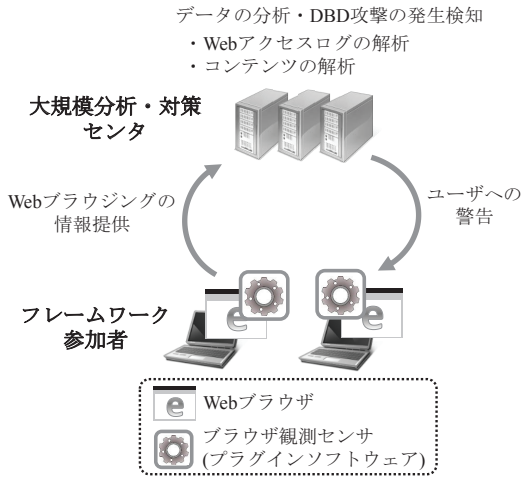

図 1: Drive-by Download 攻撃対策フレームワーク

Download) は，Drive-by Download に係る悪性サイトの早期発見，検出および防御を目的としたフレームワークである [2]，[4]．図 1 に FCDBD フレームワークの構成を示す．FCDBD フレームワークはユーザ側に配置される観測センサと，観測センサから提供された情報を解析する解析センタからなる．

観測センサは，ユーザの Web ブラウザ (ブラウザセンサ) および Web プロキシサーバ (Web プロキシセンサ) に設置され，ユーザの Web アクセスに関する情報を観測し，得られた情報を解析センタに送信する．ブラウザセンサは Web ブラウザのプラグインソフトウェアとして実装され，表 1 に記載した内容を含む Web アクセスログを解析センタに送信する．その際，個々のブラウザセンサは ID で識別されるが，この ID は Web ブラウザが起動されるごとにランダムに変更されるため，解析センタ側で同一ユーザの Web アクセスログを追跡できない．ブラウザセンサはまた取得した Web サイト上のコンテンツをセンタに送信する．その際，解析センタ側で悪性が疑われると判断されたサイトのコンテンツのみをセンタからの要求に応じて送信する．

解析センタは，観測センサから送信された情報を解析し，悪性と思われるサイトを検出する．そして，観測センサに検出された悪性サイトの情報を送信し，観測センサを利用するユーザが

表 1: Web アクセスログの主な内容

・観測センサの ID
・ユーザがアクセスした URL
・ダウンロードしたコンテンツのハッシュ値
・Web ページ遷移時のマウスイベントの有無
・HTTP Request/Response ヘッダ

悪性サイトにアクセスするのを防ぐ．図 2 に解析センタでの処理フローの概要を示す．観測センタから Web アクセスログを受信すると (1)，解析センタはまず Web アクセスログ内のアクセス URL とコンテンツのハッシュ値を，センタ内で保持しているブラックリストと照合する (2)．当該ブラックリストは，解析センタで過去に悪性と判定したコンテンツの URL およびハッシュ値，外部機関から取得したブラックリストにより構成される．次に解析センタは Web アクセスログをもとに Web サイトのリンク構造を解析して，当該サイトが悪性かどうかを判定する (3)．(3) では後述するコンテンツのダウンロードに至るページ遷移の振る舞いにもとづいて，当該アクセスが Drive-by Download 攻撃におけるマルウェアのダウンロードかどうか判定する [3]．解析センタは (2)，(3) の結果をセンサに返答する (4)．センサでは，この結果にもとづいて適宜サイトへのアクセスを遮断する．さらに解析センタは，観測センサに対して (2)，(3) で悪性と判定したサイトのコンテンツの送信を要求する (5)．またコンテンツの送信を要求するターゲットとして，(3) で採用した検出手法の他に，遷移先サイトの変化に着目して改ざんが疑われるサイトを検出する手法 [4] を用いて疑わしいコンテンツを収集し，検査する．観測センサは要求されたコンテンツを解析センタに送信する (6)．解析センタでは，送信されたコンテンツを解析し (7)，悪性と判定したコンテンツの URL およびハッシュ値をブラックリストに登録する (8)．また良性と判定したコンテンツがブラックリストに記載されている場合は，当該コンテンツをブラックリストから削除する．

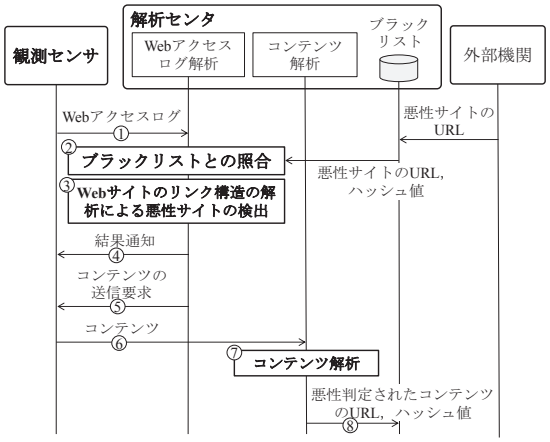

図 2: 解析センタでの処理フロー概要

4 実験によるデータ収集

開発したFCDBDシステムを用いて，実際に100人程度のユーザにブラウザセンサを配布してデータの収集を実施した．データ収集は2014年11月4日〜2015年3月31日の期間で実施した．参加ユーザは個別にご協力をお願いしてご賛同いただいた大学，研究室および企業に所属されている方々である．以降，実験にて得られたデータについて解析結果を記載する．

4.1 収集されたWebアクセスデータ

図3に2014年12月1日から2015年3月31日までにおいて1日あたりに収集されたURLとデータの提供があったセンサIDの数の遷移を示す．センサIDは，3節で述べたとおり，ブラウザの起動ごとにランダムに生成される．そのため，3のセンサIDの数は，必ずしも実際にセンサを起動したユーザの数と一致しないことに注意されたい．参加ユーザは2014年11月4日の実験開始から徐々に増加し，2015年1月下旬頃に100人に到達した．そのため，2014年12月頃と2014年1月下旬以降では，センサIDおよびURLの数に差が生じている．上記期間で平均すると，1日あたり55センサからのデータ提供があり，4,581URLのデータが収集されていた．

表2に，期間内にアクセスされた全URLにおけるユニークなURL，ホスト名，ドメイン名

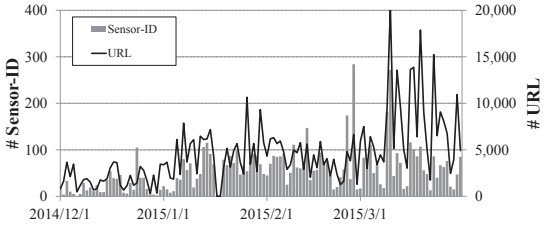

図 3: アクセスURL，センサIDの数の遷移

表 2: ユニークなURL，ホスト名，ドメイン名の数

	全アクセス	初期ページ
URL	390,081	28,359
ホスト名	11,696	6,319
ドメイン名	5,419	3,733

の数，および初期ページにおける各集計結果を示す．ここで初期ページとは，ユーザがリンクのクリック，URLの手動入力，ブックマークの選択などにより明示的にアクセス先URLを指定してアクセスされたWebページを指す．期間内における全アクセスについて1日あたり平均して3,224URL，97ホストずつユニークなURLおよびホスト名が新たに増加し続けていた．また，初期ページは1日あたり平均して236URL，52FQDN，31ドメインずつ増加していた．全アクセスについては，初期ページへアクセスした際に付随してアクセスされる広告などへのアクセスも含まれており，これらのURL(ホスト名)はパラメータなどが付与された一時的なURLであることが多い．そのため，初期ページと比較してURL数が約13.7倍となっている．

収集されたWebアクセスデータの網羅性を検証すべく，Alexa[11]から取得した日本でのドメイン別アクセスランキングをもとに，アクセスされた初期ページが上位100ドメイン，500ドメイン上のページである割合について調べた．

表3に，期間内にアクセスされた初期ページに含まれるAlexaの上位100ドメイン，500ドメインの数を示す．表のとおり，4か月間で上位100ドメインのうち91ドメイン，上位500ドメインのうち319ドメインがアクセスされていた．このことより，今回の100人程度の規模においても，FCDBDフレームワークにてほぼ

表 3: 主要ドメインへのアクセス

Alexa ランク	アクセスされたドメイン数
上位 100(日本)	91 (91.0%)
上位 500(日本)	319 (63.8%)

表 4: 初期ページへのアクセス全体における主要ドメインへのアクセスの割合 ($N = 118,564$)

Alexa ランク	アクセス数
上位 100(日本)	44,918 (37.9%)
上位 500(日本)	56,514 (47.7%)

表 5: 月日ごとの初期ページへのアクセス全体における主要ドメインへのアクセスの割合

月日	全アクセス数	上位 500(日本)
2014/12	7,242	3,624 (50.0%)
2015/3	18,185	8,067 (44.4%)

表 6: 初期ページのアクセス数の分布

アクセス数	1	< 10	≥ 10
初期ページ数	14,665	12,910	1,089
($N = 28,664$)	51.2%	45.0%	3.8%

主要なドメイン，サイトへのアクセスに関するデータが収集できていることが確認できた．

表 4 に初期ページへのアクセス全体に Alexa ドメイン上のサイトが含まれる割合を示す．表より，初期ページへのアクセスのうち 47.7%が Alexa の上位 500 位のドメインへのアクセス，37.9%が上位 100 位に属するドメイン上へのアクセスであった．このようにユーザは主要なドメイン以外にも広範囲にアクセスしているため，FCDBD フレームワークのようにユーザの実際の Web アクセスにもとづきデータを収集し Web の観測を実施することが，ユーザがアクセスしうるサイトを網羅的に観測する上で重要であると考えられる．

表 5 に，2014 年 12 月と 2015 年 3 月それぞれにおける初期ページへのアクセス全体に対する Alexa 上位 500 ドメイン上のサイトへのアクセスの割合を示す．2014 年 12 月と 2015 年 3 月で主要ドメインへのアクセスの割合を比較すると，全アクセス数が多い 2015 年 3 月の方が，Alexa 上位ドメインへのアクセスの割合が小さい．これより，参加ユーザ数が増えてアクセス数が増加すると，アクセスされる URL の多様性が増加し，総体的に主要ドメインへのアクセスの割合が減少すると推測できる．

次に初期ページごとのアクセス数を集計した結果を示す．表 6 にアクセス数ごとの初期ページ数の割合を示す．集計結果より，初期ページの約 51.2%が 1 回しかアクセスされていなかったことがわかる．このような Web サイトについては，過去のアクセスデータが参照できないため，時系列変化による改ざん検出の効果は期待できない．しかし，これらの URL を調査すると，大多数は一時的なパラメータと思われる内容がパスもしくはパラメータ部に含まれるものであった．実際に，アクセス回数が 1 回のみであったサイトのうち，URL のパスの部分が'/'で終わるもの，もしくは `index`，`top` を含むものといった Web サイトのトップページと思われるものは 3612URL であり，アクセス数が 1 回のみの初期ページの 24.6%程度であった．

表 7 に各初期ページへアクセスしたセンサ ID 数の分布を示す．表より約 54.1%の初期ページが 1 人のユーザからしかアクセスされていないことがわかる．このようなサイトにおいては，FCDBD フレームワークにおける複数ユーザからの監視による攻撃防止は期待できない．しかしながら，1 人の参加ユーザが複数回同じページにアクセスしている場合は，以前のアクセスとの比較によって攻撃の有無を検知できる可能性がある．

表 8 に初期ページごとにアクセスされた日数を集計した結果を示す．表より，約 72.8%のページが 1 日しかアクセスされないページであることがわかる．このようなサイトでは，長期的な観測は難しいが同日に複数の参加ユーザからアクセスされていた場合は，即応的に攻撃を検知し被害拡大を抑えられる可能性がある．また同表より，約 5.4%の初期ページが 4 日以上，平均して 1 か月に 1 回アクセスされていることがわかる．

以上より，収集された Web アクセスデータより，Alexa 上位ドメインに属するような主要な Web サイトにおいては十分な網羅性を確保でき

表 7: 各初期ページへアクセスしたセンサの ID 数の分布

センサ ID 数	1	< 10	≥ 10
初期ページ数	15,499	12,484	681
$(N = 28,664)$	54.1%	43.5%	2.4%

表 8: 初期ページごとのアクセスされた日数の分布

アクセスされた日数	1	< 4	≥ 4
初期ページ数	20,867	6,244	1,553
$(N = 28,664)$	72.8%	11.8%	5.4%

ることがわかった．反面，初期ページの 51.4%においてはアクセス回数が 1 回のみであり，FCDBD フレームワークによる観測が十分に行えない．さらに初期ページの 54.1%は参加ユーザ 1 人からしかアクセスされていない，初期ページの 72.8%は 1 日しかアクセスされていない．これらのサイトに関しては FCDBD フレームワークによる観測による効果が限定的である．しかしながら，これらの Web サイトへのアクセスが全アクセス数 (118,564 回) に占める割合は，アクセス数が 1 回のみのサイトへのアクセスは 12.4%(14,665 回)，アクセス日数が 1 日のみのサイトへのアクセスは 26.5%(31,474 回) であり，参加ユーザの全アクセスの 73.5%に関しては FCDBD フレームワークでの観測による攻撃防止効果が期待できることとなる．以上を図に示すと図 4 となる．図内の黒塗り部分が FCDBD フレームワークによる効果が期待できる参加ユーザのアクセス，灰色の部分が限定的ながら観測による攻撃防止効果が期待できるアクセスとなる．上述したとおり，全アクセスの 47.7%は従来のクローリング技術において観測されている主要なドメインの Web サイトに関するものである．残りの 25.8%が本フレームワークにより新たに観測が可能となる Web サイトである．

4.2 収集されたユーザ環境のデータ

参加ユーザのマシン環境の集計結果を以下に示す．今回収集されたマシン環境は Web ブラ

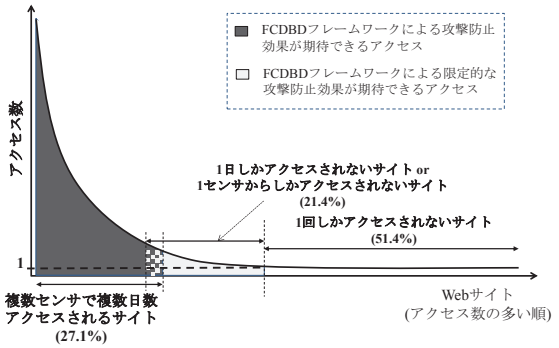

図 4: FCDBD フレームワークによって観測される範囲

表 9: 参加ユーザのマシン環境 (Web ブラウザ)

Firefox: 2,528 台			
Ver	台数	Ver.	台数
36	731	30	7
35	1,172	29	2
34	503	28	1
33	76	26	1
32	2	20	32
31	1		
Internet Explorer: 1,662 台			
Ver	台数	Ver.	台数
11	1,418	9	141
10	87	8	16

ウザ (種類×バージョン) が 17 通り，Web ブラウザの設定 (オプション) が 4 通り，プラグイン (種類×バージョン) が 416 種類であった．

全センサ ID 4,190ID における Web ブラウザの内訳について表 9 に示す．今回の実証実験ではブラウザセンサとして Firefox，InternetExplorer の 2 種類に対応したプラグインを提供していたが，両者の内訳は Firefox : Internet Explorer = 6:4 であった．表 10 に，各月ごとの Web ブラウザのバージョンの分布を示す．Firefox では，ほぼ最新バージョンのリリースに追随している様子が観測された．しかし，バージョン 36 のリリースが 2015 年 2 月にもかかわらず，2015 年 3 月で約 15.6%のセンサが 35 以前のバージョンを利用しているなど，多少最新版へのアップデートの対応に時間差がある様子が観測された．Internet Explorer では，約 14.7%のユーザが旧バージョン (10 以前) を利用していた．

Web ブラウザのオプションの分布を表 11 に示

表 10: Web ブラウザのバージョンの分布

Firefox				
Ver.	2014/12	2015/1	2015/2	2015/3
36			10	721
35		414	662	96
34	202	291	5	5
33	61	14	1	
32	1	1		
31 以前	3	8	1	32
Internet Explorer				
11	103	486	393	436
10	21	1	42	23
9		80	51	10
8	3		6	7

表 11: 参加ユーザのマシン環境 (Web ブラウザのオプション (1:有効，0:無効))

ProtectMode	SmartScreen	台数
0	0	316
0	1	249
1	0	793
1	1	1,632

す．Web ブラウザのオプションについては，セキュリティ関連の設定項目である ProtectMode，SmartScreen について有効/無効を取得した．なお，表 11 では Internet Explorer のオプションの集計のみを示す．Firefox については，Firefox のアップデートにともなう設定項目名の変更などが影響し，オプションの取得ができなかった．表 11 より，約 45.5%のユーザはデフォルトの設定 (すべて有効) から一部設定を変更している状況がみられた．

参加ユーザのブラウザにインストールされていたプラグインに関して，プラグインの種類は Internet Explorer で 105 個，Firefox で 91 個であった．表 12 にインストールされた台数の多いプラグインを示す．また表 13 に Shockwave Flash のバージョンの分布を示す．表 13 より，多少最新版へのアップデートの対応には個人差があるようであった．

以上より，今回取得したデータにおいては，ほぼ大多数の参加ユーザがブラウザおよびプラグインのアップデートをこまめに実施している様子が観測された．しかし，アップデートの対応に多少の時間差がみられることから，例えば

表 12: 参加ユーザのマシン環境 (プラグイン)

Internet Explorer	
プラグイン	台数
Office Document Cache Handler	1,036
Adobe PDF Reader	791
Shockwave Flash Object	788
Java(TM) Plug-in SSV Helper	632
Google Toolbar	539
Firefox	
プラグイン	台数
Shockwave Flash	2,392
Adobe Acrobat	2,377
Microsoft Office	2,040
Google Update	1,936
Java(TM) Platform SE	1,859

表 13: Flash のバージョンの分布

Ver.	2014/12	2015/1	2015/2	2015/3
17.0.0.134				282
16.0.0.305		414	671	746
16.0.0.296		222	222	151
16.0.0.287		70		
16.0.0.257		328	2	
16.0.0.235	119	193	2	
15.0.0.246	125	91	3	
15.0.0.239	81			
15.0.0.223	2	6	2	
15.0.0.152		3		2
14 以前	1	23	36	32

[12] のようなユーザ環境の違い，特にプラグインのバージョンの違いによるリダイレクトパスの違いを観測できる可能性がある．

5 まとめ

本稿では，実装した FCDBD フレームワークを用いて実際に 100 人のユーザにブラウザセンサを配布して収集したデータを解析し，FCDBD フレームワークの有効性について検証した．今後，さらに参加者を増やして参加者の増加による観測可能な Web サイトの広がりなど網羅性の変化を評価するとともに，Drive-by Download 攻撃対策技術の研究開発に有益となる Web アクセスデータの収集を引き続き実施し，当該データを利用した悪性 Web サイトの検出手法の評価を実施する．

現在，当該フレームワークの実験協力者を公募している．興味のある方はFCDBDフレームワークのポータルサイト (`http://www.fcdbd.jp`) を参照されたい．

謝辞

本研究成果は国立研究開発法人 情報通信研究機構 (NICT) の委託研究「ドライブ・バイ・ダウンロード攻撃対策フレームワークの研究開発」により得られたものである．ここに深謝する．

参考文献

[1] N. Provos, P. Mavrommatis, M. A. Rajab and F. Monrose, *All Your iFRAMEs Point to Us*, Proc. the 17th USENIX Security Symposium, 2008.

[2] 笠間貴弘, 井上大介, 衛藤将史, 中里純二, 中尾康二,「ドライブ・バイ・ダウンロード攻撃対策フレームワークの提案」, コンピュータセキュリティシンポジウム 2011(CSS2011), 2011.

[3] T. Matsunaka, A. Kubota and T. Kasama, *An Approach to Detect Drive-by Download by Observing the Web Page Transition Behaviors*, Proc. of 9th Asia Joint Conference on Information Security (AsiaJCIS2014), 2014.

[4] T. Matsunaka, J. Urakawa and A. Kubota, *Detecting and Preventing Drive-by Download Attack via Participative Monitoring of the Web*, Proc. of 8th Asia Joint Conference on Information Security (AsiaJCIS2013), 2013.

[5] M. Akiyama, M. Iwamura, Y. Kawakoya, K. Aoki and M. Itoh, *Design and Implementation of High Interaction Client Honeypot for Drive-by-Download Attack*, IEEE Trans. of Communication, Vol. E93–B, No. 5, pp. 1131–1139, May. 2010.

[6] Y–M. Wang, D. Beck, X. Jiang, C. Verbowski, S. Chen and S. King, *Automated Web Patrol with Strider HoneyMonkeys: Finding Web Sites That Exploit Browser Vulnerabilities*, Proc. 13th Annual Network & Distributed System Security Symposium (NDSS2006), 2006.

[7] J. Zhang, C. Seifert, J. W. Stokes and W. Lee, *ARROW: GenerAting SignatuRes to Detect DRive-By DOWnloads*, Proc. 20th International World Wide Web Conference (WWW2011), 2011.

[8] G. Stringhini, C. Kruegel and G. Vigna, *Shady Paths: Leveraging Surfing Crowds to Detect Malicious Web Pages*, Proc. 20th ACM Conference on Computer and Communications Security (CCS2013), 2013.

[9] G. Wand, J. W. Stokes, C. Herley and D. Felstead, *Detecting Malicious Landing Pages in Malware Distribution Networks*, Proc. 43rd Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN2013), 2013.

[10] 進藤康孝, 佐藤彰洋, 中村豊, 飯田勝吉,「マルウェア感染ステップのファイルタイプ遷移に基づいたDrive-by Download攻撃検知手法」, コンピュータセキュリティシンポジウム 2014(CSS2014), 2014.

[11] *Alexa – Actionable Analytics of the Web*, `http://www.alexa.com`.

[12] 笠間貴弘, 衛藤将史, 神薗雅紀, 井上大介,「クライアント環境に応じたリダイレクト制御に着目した悪性Webサイト検出手法」, 信学技報 Vol.114, No.71, ICSS2014-5, 2014.