CMS に対する攻撃情報の収集方式の実装と評価

松本 悦宜 †‡　　三木 剛 †　　力宗 幸男 ‡

† 株式会社神戸デジタル・ラボ
650-0033 兵庫県神戸市中央区江戸町 93 番 栄光ビル 2F
y-matsumoto@kdl.co.jp

‡ 兵庫県立大学大学院 応用情報科学研究科
650-0047 兵庫県神戸市中央区港島南町 7 丁目 1 番 28 計算科学センタービル 5-7 階

あらまし CMS を使用した Web サイトが増加する一方で，ユーザ数の多い CMS を標的とした攻撃も増加しており，多くの国内外の Web サイトが改ざんされる事例が確認されている．これらの攻撃は，CMS の構造を使用したものや，CMS やプラグインなどの関連ソフトウエアの脆弱性を狙ったものなど，従来の攻撃と異なったものになっている．またネットワークにおける攻撃と異なり，ある程度アクセスを期待する Web サイトにおいて多く攻撃を受ける可能性がある．本稿では，特定の CMS において，収集を行う Web サイトと収集した攻撃を分析するシステムを構築する．また，実際に運用していくなかで収集したデータを分析することで，これらの攻撃への対策を提案する．

Implementation of The System for Gathering Attack Vectors In Websites Using CMS

Yoshinori Matsumoto†‡　　Tsuyoshi Mikii†　　Yukio Rikiso‡

†Kobe Digital Labo Inc.
Eikou Bldg. 2F, 93 Edo-machi, Chuo-ku, Kobe, Hyogo, 650-0033, JAPAN
y-matsumoto@kdl.co.jp

‡Graduate School of Applied Informatics University of Hyogo
Computational Science Center Building 5-7F,
7-1-28 Minatojima-minamimachi, Chuo-ku, Kobe, Hyogo, 650-0047, JAPAN

Abstract There are many websites using CMS, but It is increasing attacks against websites using a popular CMS. Some of them are defaced. These attackers use features and vulnerabilities of the CMS or their related software such as plugins. Although servers such as honeypot are very useful for gathering network attacks, they cannot gather attacks against websites using CMS. In this paper, we develop the system gathering and analytic attacks for some websites using a popular CMS. Furthermore, we propose the solution of gathered attacks.

1　はじめに

Web サイトを開発するうえで，CMS (Contents Management System) を使用して開発する例が増加している．CMS を使用することにより，開発時や更新時のコストを抑えることができる．また，他のベンダーや開発者が作成したものを拡張機能として追加することができ，自由度の高い Web サイトを開発することができる．

一方で， CMS を使用した Web サイトに対する攻撃が増加している．Netcraft 社によると，マルウエアをホスティングしていたサイトのうち約 19%の IP アドレスが WordPress と呼ばれる CMS を使用しているサイトであったとの調査結果を発表した [1]．また，日本国内の CMS を使用した Web サイトにも攻撃が増加しており，2013 年には，IPA が古い CMS を使用した Web サイトの攻撃に対する注意喚起を発行し [2]，2014 年には，JPCERT/CC が古い Movable Type と呼ばれる CMS を使用した Web サイトの改ざん件数が増加していると注意喚起を発行した [3].

CMS に対する攻撃は，従来の Web サイトに対する攻撃だけでなく，使用している CMS の構造や付属している拡張機能の脆弱性，サードパーティベンダーや開発者が作りこんでしまった脆弱性などを使用した攻撃が考えられる．これらの攻撃に備えるために，具体的な攻撃情報を把握することが必要になる．従来では，攻撃情報を把握するために，ハニーポットのようなシステムをインターネット上に設置し，システムが受け取るパケットを収集し分析する方式がよく用いられてきた．しかし，CMS に対する攻撃を収集する場合は，特定の CMS に対する攻撃は十分に収集できないと考えられる．これは，特定の CMS に対する攻撃は，攻撃者がそれぞれの CMS を使用した Web サイトであると判断していると考えられるためである．

本稿では，これらの問題を踏まえ，実際に CMS を使用した攻撃を収集するため，攻撃を収集するシステムを開発し，評価を行った．

2　取り扱う CMS について

W3Techs の調査 [4] によれば，ユーザ数の多い CMS は表 1 の通りである．

表 1: 主な CMS

CMS 名	市場占有率 (%)
WordPress	60.6
Joomla	8.0
Drupal	5.2

本稿では，世界的に最もユーザ数の多いとされる WordPress を対象とした．

2.1　攻撃例

WordPress を使用した Web サイトにおいて，考えられる攻撃手法は以下のものである．本章では，それぞれの攻撃手法について解説する．

- ブルートフォース攻撃
- 拡張機能の脆弱性を使用した攻撃
- コアファイルの脆弱性を使用した攻撃
- アカウント管理の不備を使用した攻撃

本稿では，この中から「ブルートフォース攻撃」および「拡張機能の脆弱性を使用した攻撃」を対象にした．

2.1.1　ブルートフォース攻撃

WordPress には，記事の編集や拡張機能の管理などを行うため，ユーザを認証するログイン画面が用意されている．初期設定において，ログイン画面の構造および URL はすべての Web サイトに共通となる．この構造は，主に ID とパスワードを POST リクエストに送るだけであり，リクエスト構造は非常に単純なものになっている．また，初期設定においてトークンなどによる送信元の確認や，ログイン試行回数の上限がないことから，ログイン時の POST リクエストを大量に試行するブルートフォース攻撃が可能になっている．

2.1.2 拡張機能の脆弱性を使用した攻撃

WordPress には，トップページ，各記事，その他の静的ページの外観を設定するテーマや，機能を追加するプラグインなど，Web サイトごとに自由度の高い開発が可能になる拡張機能がサポートされる．また，これらのテーマやプラグインなどを配布または販売する場合もあり，1つの Web サイトで複数のサードパーティ製の製品が使用されていることがある．

しかしながら，セキュリティ対策はそれぞれの拡張機能の開発者に委ねられており，一部の製品では，深刻な脆弱性を作り込んだまま配布され続けているものも見られる．

2.1.3 コアファイルの脆弱性を使用した攻撃

WordPress のコアファイルの脆弱性を使用した攻撃も見られる．WordPress はコアファイルのアップデート時に脆弱性が修正されることもあり，常に最新版を使用することが望まれている．

WordPress のバージョン 3.7 からは，マイナーアップデートについては自動的にアップデートが行われるようになったものの，当該バージョンより古い WordPress を使用している Web サイトや，自動的にアップデートする機能を使用していない Web サイトのうち更新が行われていない Web サイト，またはメジャーアップデートを行っていない Web サイトなど，古いバージョンの WordPress を使用している Web サイトも見られる．

2.1.4 アカウント管理の不備を使用した攻撃

Web サイトの更新を行う際に FTP を使用する場合もあるが，FTP のアカウントが適切に管理されていないことにより，FTP サーバに不正にログインされ，Web サイトが改ざんされる事例も見られる．

3 収集方法

本稿で使用したシステムの概要を，図 1 に示す．収集は以下の方法で行う．

1. ログ管理システムを設置する
2. 収集サーバを設置する
3. このサイトに来るアクセスをリバースプロキシで取得する
4. ログ管理システムに送信する
5. ログ管理システムにおいて，ログの確認または分析を行う

3.1 収集サーバ

収集サーバは，WordPress を使用したサーバおよび通常の Web サーバの 2 種類の方法で運用した．それぞれのサーバの概要を表 2 に示す．但し，表 2 において，対象期間内にサーバの障害やメンテナス等の理由で停止していた期間も含んでいる．

3.1.1 WordPress を使用したサーバ

ログの収集を行うための Web サーバとして，3 つのサーバを設置した．これらのサーバは WordPress をインストールし，表 2 のサーバ A からサーバ C のように運営した．

WordPress のバージョンは，WordPress 3.9.1 および 3.9.2 を使用した (2014 年 8 月現在)．バージョンは随時アップデートを行い最新版を使用するようにした．

これらのサーバに対するアクセスを増やすために，コンテンツを定期的に更新するようにした．ここで更新するコンテンツは，外部の Web サイトから情報収集を行い，これらの情報を元にコンテンツを作成するスクリプトを開発した．この時，それぞれのサーバごとに異なるコンテンツが作成されるようにした．作成したコンテンツは，WordPress の XMLRPC を取り扱う機能を使用して更新した．

それぞれのサーバにはリバースプロキシを設置し，アクセスログを収集した．ここで収集す

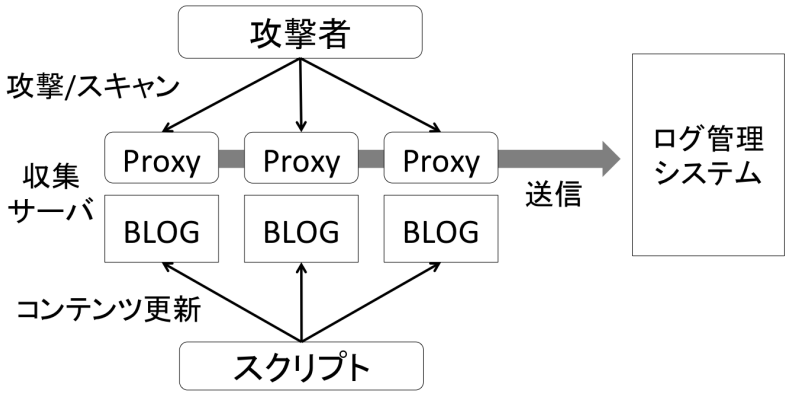

図 1: システム概要

る情報は，アクセス元の IP アドレス，タイムスタンプ，アクセス先 URL，POST リクエストのボディ部分である．

3.1.2 通常の Web サーバ

また，通常の Web サーバに対するアクセスと，WordPress に対するアクセスを比較するために，Web サーバのみのサーバも設置した．このサーバは，サーバ D として表 2 に示す．サーバ D は Apache 2.2 を使用し，コンテンツを作成せず運用した．

3.2 ログ管理システム

収集したログを一元的に管理するシステムを開発した．本システムは，それぞれの収集サーバから送信されたログをデータベースに格納し，Web インタフェース経由で検索・閲覧できるようにした．また，アクセス先 IP アドレスから割り当てられている国名を調べてデータに追加した．

4 収集結果

収集の結果，以下に示すような，攻撃と考えられるアクセスを収集した．

- ログインページに対するアクセス
- テーマディレクトリ以下に対するアクセス
- プラグインディレクトリ以下に対するアクセス

4.1 ログインページに対するアクセス

ログインページ (wp-login.php) に対するアクセスをサーバごとにまとめると表 3 のようになった．WordPress を使用しているサーバ A から C にはアクセス数に差はあるものの，複数回のアクセスを確認した．一方，サーバ D はアクセスを確認できなかった．

表 3: ログインページに対するアクセス

サイト名	アクセス数
サーバ A	104,866
サーバ B	18,537
サーバ C	1,927
サーバ D	0

4.2 テーマディレクトリに対するアクセス

テーマディレクトリ (wp-content/themes) 以下に対するアクセスをテーマごとにまとめると表 4 にようになった．表 4 において，テーマ A から C は，それぞれ異なる，本サーバでは使用していないテーマである．「その他のテーマ」と

表 2: 収集サーバ一覧

サーバ名	運用方法	対象期間	概要
サーバ A	WordPress	2014 年 5 月から 8 月まで	メディアに関連する情報を発信するサイト
サーバ B	WordPress	2014 年 5 月から 8 月まで	話題となっている情報を発信するサイト
サーバ C	WordPress	2014 年 5 月から 6 月まで	技術的な情報を発信するサイト
サーバ D	Web サーバ	2014 年 5 月から 8 月まで	Web サーバを起用し外部からのアクセスを受け付けているものの，外部に公開するコンテンツは作成しない

してまとめたテーマは，運営中に使用したテーマやなど，攻撃とは確認できなかったもの指している．この中で 3 種類のテーマに対して攻撃と思われるアクセスが確認できた．

表 4: テーマに対するアクセス

テーマ	アクセス数
テーマ A	1
テーマ B	1
テーマ C	1
その他のテーマ	4193

4.3 プラグインディレクトリに対するアクセス

プラグインディレクトリ (wp-content/plugins) 以下に対するアクセスをプラグインごとにまとめると表 5 のようになった．表 5 において，プラグイン A から K は，それぞれ異なる，本サーバでは使用していないプラグインである．また，「その他のプラグイン」としてまとめたプラグインは先述と同様，攻撃とは確認できなかったものを指している．この中で，11 種類のプラグインに対して攻撃と思われるアクセスが確認できた．

表 5: プラグインに対するアクセス

プラグイン	アクセス数
プラグイン A	2
プラグイン B	2
プラグイン C	1
プラグイン D	1
プラグイン E	1
プラグイン F	1
プラグイン G	1
プラグイン H	1
プラグイン I	1
プラグイン J	1
プラグイン K	1
その他のプラグイン	907

5 考察

これらのアクセスの中から以下のスキャンまたは攻撃と考えられるアクセスについて分析を行った．

- ログイン試行
- SQL インジェクション
- 遠隔からのファイルアップロード

本章では，それぞれのアクセスについて行った分析を解説する．また，運用しているサイトや製品の名前を伏せるため，一部リクエストのデータから変更していることがある．

5.1 ログイン試行

以下のアクセスは，ログイン画面 (wp-login.php) に対して行われたログイン試行のブルー

トフォース攻撃の POST リクエストの中から，1 件のリクエストのボディ部分のみを抽出したものである．ログイン時には，POST データの log パラメータにユーザ ID，pwd パラメータにパスワードを入力して送信するようになっており，以下のアクセスは ID を admin，パスワードを football としてログイン試行を行っていると考えられる．

WordPress の初期設定ではログイン試行回数に制限はないため，同一 IP から大量のログイン試行が行われることが確認された．

ブルートフォース攻撃として確認できたアクセスの中から，多く使用されたパスワードを表 6 に集計した．表 6 から，パスワードは password など，一般的な英単語に対し多くログイン試行されており，Web サイトのドメインや，ドメインと英単語の組み合わせなどが使用されることもあった．

また，ログイン試行される ID は admin が多く見られた．これは以前 WordPress の管理者権限の初期 ID が admin であったことから，admin が管理者権限を持ったユーザで使用されている場合が多いと考えられる．パスワードと同様，ID にドメイン名が含まれることも多く見られた．

仮に，ユーザが簡単なパスワードを設定しているとログインが成功し，記事の投稿や Web サイトの改ざんなどが行われる可能性がある．

```
log=admin&pwd=football&wp−submit=
%D0%92%D0%BE%D0%B9%D1
%82%D0%B8&redirect_to=http%3A
%2F%2F(domain)%2Fwp−admin%2F
&testcookie=1
```

表 6: 試行されたパスワード

パスワード	出現回数
(domain)123456	102
admin	80
12345678	75
password	73
123456	72

5.2 SQL インジェクション

リクエストの中に，配布されているプラグインの SQL インジェクションの脆弱性を使用した攻撃が確認された．

WordPress は MySQL と連携しコンテンツを管理しているが，テーマやプラグインが SQL 文を発行する際に，SQL インジェクションの脆弱性があった場合，攻撃者は，パラメータを細工することにより，任意の SQL 文を発行し，データベースの閲覧や改ざんを行う可能性がある．

SQL インジェクションの脆弱性をもったプラグインに対する攻撃の一例を以下に示す．

ここでは，以下の URL でアクセスすると，当該プラグインの使用により，URL パラメータの文字列がデコードされ SQL 文として解釈されるようになっている．このとき，パラメータを細工することで運営者の意図しない SQL 文を発行する可能性がある．

本脆弱性は当該プラグインの最新バージョンでは修正されており，実証コードは一般に公開されていた．

```
/wp−content/plugins/(plugin name)/
library/(vulnerable).php?track=
LTEgVU5JT04gU0VMRUNUIHZlcn
Npb24oKSwxLDEsMQ==
```

5.3 遠隔からのファイルアップロード

WordPess のテーマとプラグインは，それぞれのユーザによって変更しやすくするために，ファイルをアップロードをする機能をもつ場合がある．遠隔からのファイルアップロードが可能である問題をもつテーマおよびプラグインに対して，ファイルをアップロードしようとするアクセスを確認した．

本攻撃の概要を図 2 に示す．図 2 の通り，通常はこれらの機能は，ログイン後の管理画面からのアクセスに制限する必要があるが，直接アクセスするなどにより遠隔の攻撃者からのアップロードが可能になる場合があった．

攻撃者がアップロードを行うと，通常はアップロード用のディレクトリに保存され，ファイルに認証無しでアクセスが可能になる．

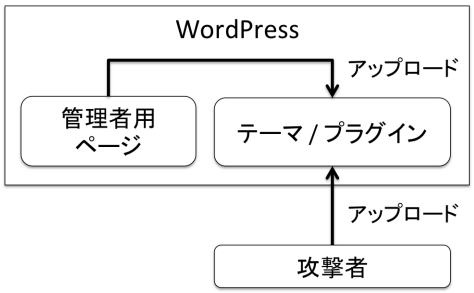

図 2: 遠隔からのファイルアップロード

以下に，アップロード時のリクエストの一部を示す．このテーマおよびプラグインには PHP ファイルが送信されている．これは，アップロード後にこのファイルにアクセスすることにより，攻撃者からの指示に従い，WordPress が設置しているサーバ内で，コマンド実行などの操作を行うためのバックドアとして使用されると考えられる．

本脆弱性は当該テーマおよびプラグインの最新バージョンでは修正されており，実証コードは一般に公開されていた．

```
Content−Disposition: form−data;
name="Filedata"; filename="ifire.php
    "
Content−Type: text/plain

<?php eval(gzinflate(base64_decode("DZZ
HDqwIEkTv0qv/xQIovEa9wHtXeDYtvPee0
09dIJX5FBkRxZn0f6q3Gcs+2Ys*******AII
gUYKX/u8/f//+/d//AQ=="))); ?>
```

5.4 通常のサーバとの比較

表 3 の通り，ログインに対する攻撃は，サイトごとに数は異なるものの，複数回のアクセスを確認できた．対して，通常の Web サーバとして設置しているだけのサイト D では確認できなかった．また，サイト D では，複数の他ソフトウエアの脆弱性を使用した攻撃と考えられるようなアクセスを確認した．

このため，攻撃者は何らかの方法でサイト A から C に関しては WordPress を使用しているサイトであると認識している可能性が高いと考えられる．

6 結論

以上のことから，WordPress のサイトを使用して攻撃情報を収集することによって，WordPress のサイトを狙った攻撃またはスキャンを収集することができた．

この中には，サードパーティ製品の脆弱性を使用すると考えられる攻撃もみられ，実装した方式は攻撃の傾向を明らかにするうえで有用であることがわかった．

7 今後の課題

収集サーバの数を増やし，攻撃情報を集め体系的な分析ができるようにする予定である．収集したデータは機械学習などを使用して分析の効率をあげる予定である．

また，ほかの CMS でも同様のことを行い，CMS ごとの比較を行う予定である．

参考文献

[1] WordPress hosting: Do not try this at home!
http://news.netcraft.com/archives/2014/03/24/wordpress-hosting-do-not-try-this-at-home.html

[2] WordPress や Movable Type の古いバージョンを利用しているウェブサイトへの注意喚起
https://www.ipa.go.jp/security/topics/alert20130913.html

[3] 旧バージョンの Movable Type の利用に関する注意喚起
https://www.jpcert.or.jp/at/2014/at140024.html

[4] Usage of content management systems for websites (18 August 2014)
http://w3techs.com/technologies/overview/content_management/all