Vol.2012-IOT-16 No.51
2012/3/16

単純なハニーポットによるウエブアクセス動向の予備的調査

長谷川明生

ウエブリクエストのみを収集するPerlで実装した非常に単純なウエブハニーポットを実行し，予備実験として1ヶ月間で179件のリクエストを記録した．そのデータを解析したところ不正アクセスと解釈されるアクセスは半数を超えた．

Preliminary Analysis on the Monitoring data Captured by a Simple WEB Honeypot

Akiumi Hasegawa

A Perl script was developed to record WEB requests. Running the script for a month, 179 WEB requests were captured. It seems that half of the recorded requests were attempts of probes or exploits to security holes.

1. はじめに

インターネットのセキュリティ状況が悪化の一途をたどり，サーバ等への攻撃も巧妙化の一途をたどっている．多くのアプリケーションがウエブコンテンツとして提供されるようになってきている状況で，ウエブサーバへのアクセスの状況を把握することは重要である．高機能なハニーポットは多様な情報が取得可能であるが，運用や保守には高度な知識が必要である．

本論文では，学生でも簡単に設置でき，データ収集や解析が可能なように，小飼1のウエブサーバ・プログラムに着目し，ウエブリクエストを記録する機能だけを持った単純なウエブ・ハニーポットを作成した．そして，これを1ヶ月間走らせてアクセスを記録した．1ヶ月の間に，動作確認のためのアクセスを含めて179件のアクセスがあり，その半数が不正アクセスと結論つけられた．

2. 単純なハニーポットとデータ収集

Perlで記述したウエブ・ハニーポットを仕掛けたホストを研究室のプライベートネットワーク上に置き，ウエブ・ポートへのアクセスだけをブロードバンド・ルータで当該ホストに振り向けるように設定した．これは，設定ミス等によるセキュリティリスクを下げるためである．このIPアドレスはDNSには登録してあるが，ウエブサーバを連想させるような名前ではない．

ここで使ったウエブ・ハニーポットは，ウエブリクエストをブラウザにオウム返しする小飼のプログラムを，オウム返しをせずリクエストを記録するだけにしたものである．この簡単なプログラムを2011年12月13日から2012年1月13日の1ヶ月間データを収集した．

2.1 アクセス状況とデータ処理手順

図1に収集したデータの一部を示す．この例は，プログラムを開始した直後のログである．煩雑さを避けるために一部省略した．

記録されている時刻はGMTで，先頭の記録は，動作確認のためにプログラム起動直後に隣接するセグメントからFirefoxでアクセスしたことによるものである．図1のデータから，プログラム起動から24時間以内に外部からアクセスされていることが見てとれる．

1ヶ月に記録されたリクエストは179件で，この記録をPerlのスクリプトでCSVファイルに変換し，Accessデータベースとして分析を実施した．

中京大学情報理工学部

```
HTTP::Request=HASH(0x65b33f0)150.42.xx.yy - - [Tue, 13 Dec 2011
09:04:32 GMT] "GET /favicon.ico HTTP/1.1" 200 0
GET /favicon.ico HTTP/1.1

HTTP::Request=HASH(0x61d5810)67.135.aaa.bb - - [Wed, 14 Dec 2011
04:17:20 GMT] "HEAD / HTTP/1.0" 200 0
HEAD / HTTP/1.0
```

図 1　ハニーポットのログの例

3. アクセス記録の解析

3.1 日毎のアクセス数の変動

アクセス数を日毎に集計して図 2 に示す.

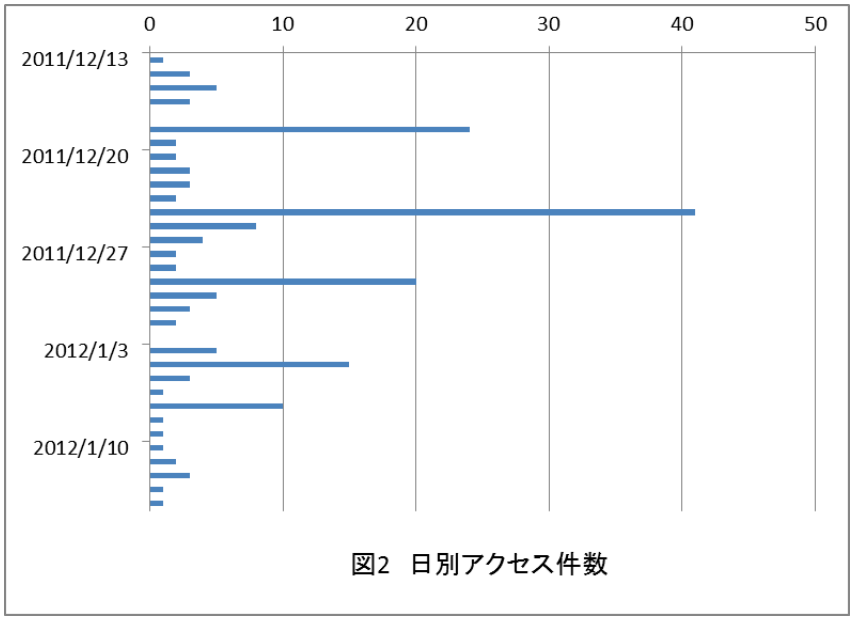

図2　日別アクセス件数

特定の日に集中してアクセスを受けていることが見てとれる．また，集中したアクセスの多くは PHP 等の脆弱性を探しているものと思われる．

3.2 リクエストメソッドとリクエスト対象

つぎにウエブのリクエストメソッドと何を要求しているかを解析した．表 1 には，リクエストメソッドと，各メソッドの件数を示す．POST を使ったものは，phpmyadmin の setup を直接操作する試みであった．

表 1　リクエストメソッドと件数

メソッド	出現件数
GET	112
HEAD	62
POST	5
合計	179

表 2 には，何を要求してきたかを示す．

表 2　リクエスト対象

要求対象	出現件数
ドキュメントルート	67
PHP 関係	62
SQL/DB 関係	5
Horde フレームワーク	8
ウエブメール	6
Google 他 URL	4
その他	14
合計	179

ふたつの表を突き合わせると，ドキュメントルートの要求数と HEAD メソッドの要求数が極めて近いことがわかる．実際，メソッドが HEAD の場合は，例外なくドキュメントルートを要求している．ドキュメントルート要求の残りは GET である．また，Google 等へのアクセスは，オープンなキャッシュを探す行動と考えられる．これら以外のアクセスは，ソフトウエアの脆弱性の検出もしくは攻撃を狙ったものと解釈できる．

3.3 IP アドレスから見たリクエスト

アクセスを IP アドレスで集約して表 3 に示す．この表では，4 回以上アクセスしてきたアドレスのみ具体的にホスト名も含めて記載した．

表 3　複数回アクセスを試みたホスト一覧

IP アドレス	ドメイン	回数	エージェント	目的
118.123.240.xxx		6		phpmyadmin
140.113.86.yyy	nctu.edu.tw	6	ZmEU	phpmyadmin
178.124.131.zzz	belarusby.com	5		blog 等
193.85.145.nnn	asysijd.cz	8		pmwiki
202.179.8.mmm		15	mozilla4.0	horde,mail
213.184.47.kkk		10		phpmyadmin
218.29.115.jjj	kd.ny.adsl	17	ZmEU	phpmyadmin
38.113.185.pp		39		phpmyadmin
94.23.45.ww	kimsufi..com	4		/
合計		110		

179 回のアクセスを IP アドレスで見ると，ユニークな IP アドレス数は 69 で，DNS 逆引きが設定されているものは 38 個であった. これらのアドレスのうち 9 個が Amazon AWS 上のものであった．なお、ホスティングサービスも 1 個記録されていた。

表 1〜3 を比較すると，単発でドキュメントルートをアクセスするホストと，なんらかの悪意を持って複数回アクセスを試みるホストが存在することが推察される．

3.4 エージェント情報の解析

リクエストのエージェント情報を表 4 に示す.

表 4　リクエスト・エージェントの分類

エージェント	回数
ZmEU	29
Mozilla/4.0(compatible,MSIE6.0,Windows98)	20
Opera らしいもの	10
Firefox らしいもの	3
その他（Pythonlib, スキャナ等）	6
なし	111
合計	179

記録を見るとドキュメントルートへのアクセスの多くは HTTP/1.0 で，エージェント情報を伴っていないのが普通である．

表から ZmEU とよばれる phpmyadmin 攻撃ツールの利用が目立つ．また，類似のツールの”Morfeus Fucking Scanner”を使ったアクセスも 1 件記録されていた．

3.5 IP アドレス割当からみた傾向

アクセス元の IP アドレスを割当国から見ると, アドレスの個数からは中国およびアメリカが目立つ．これらの国について AWS およびホスティングからの 10 個というのは時代を示している．なお，集中してアクセスを試みているアドレスの所在は東欧およびアメリカである．

4. おわりに

単純なプログラムが不正アクセスの傾向の調査に役立つことを示せた．現在のプログラムは，不備が多く資料採取には、もう今少しの改善が必要である．プログラムの改善とともに, データ採取のために運用している Argus2や Dionaea3との連携も視野に入れての調査を計画している．

参考文献

1 小飼弾，perl - HTTP::Daemon できみにも書ける Web サーバ，
ttp://blog.livedoor.jp/dankogai/archives/50686715.html，（2012 年 2 月 8 日確認）
2 http://www.qosient.com/argus/ (2012 年 2 月 8 日確認)
3 http://dionaea.carnivore.it/　(2012 年 2 月 8 日確認)