

[Лучшие](#)

[Тематические](#)

[Корпоративные](#)

Информация о хэбе

[Перевод] Защищаем сайт с помощью ZIP-бомб

0 комментариев

Старые методы по-прежнему работают



[Обновление] Теперь я в каком-то списке спецслужб, потому что написал статью про некий вид «бомбы», так?

Если вы когда-нибудь хостили веб-сайт или администрировали сервер, то наверняка хорошо знаете о плохих людях, которые пытаются сделать разные плохие вещи с вашей собственностью.

Когда я в возрасте 13 лет впервые захостил свою маленькую Linux-коробочку

с доступом по SSH, я смотрел логи и каждый день видел IP-адреса (в основном, из Китая и России), которые пытались подключиться к моей сладенькой маленькой коробочке (которая на самом деле была старым ноутбуком ThinkPad T21 со сломанным дисплеем, жужжавшим под кроватью). Я сообщал эти IP их провайдерам.

На самом деле если у вас Linux-сервер с открытым SSH, то можете сами посмотреть, сколько попыток подключений происходит ежедневно:

```
grep 'authentication failures' /var/log/auth.log
```

```
Jul 5 03:36:27 sshd[15494]: Disconnecting: Too many authentication failures for invalid user support from 201.254.82.80 port 34159 ssh2 [preauth]
Jul 5 03:36:29 sshd[15496]: Disconnecting: Too many authentication failures for root from 201.254.82.80 port 34168 ssh2 [preauth]
Jul 5 03:36:34 sshd[15498]: Disconnecting: Too many authentication failures for root from 201.254.82.80 port 34181 ssh2 [preauth]
Jul 5 03:36:43 sshd[15500]: Disconnecting: Too many authentication failures for invalid user admin from 201.254.82.80 port 34216 ssh2 [preauth]
Jul 5 03:36:58 sshd[15502]: Disconnecting: Too many authentication failures for invalid user admin from 201.254.82.80 port 34271 ssh2 [preauth]
Jul 5 05:21:48 sshd[1250]: Disconnecting: Too many authentication failures for root from 180.214.7.231 port 48232 ssh2 [preauth]
Jul 5 07:38:44 sshd[2352]: Disconnecting: Too many authentication failures for invalid user admin from 58.48.178.200 port 42818 ssh2 [preauth]
Jul 5 08:23:49 sshd[31024]: Disconnecting: Too many authentication failures for invalid user admin from 122.55.47.210 port 53703 ssh2 [preauth]
Jul 5 08:27:14 sshd[3770]: Disconnecting: Too many authentication failures for root from 40.255.145.181 port 48215 ssh2 [preauth]
Jul 5 08:49:11 sshd[12817]: Disconnecting: Too many authentication failures for root from 186.47.222.84 port 37294 ssh2 [preauth]
Jul 5 10:10:30 sshd[1864]: Disconnecting: Too many authentication failures for invalid user admin from 222.47.26.198 port 36549 ssh2 [preauth]
Jul 5 10:15:42 sshd[1777]: Disconnecting: Too many authentication failures for root from 212.43.85.148 port 57774 ssh2 [preauth]
Jul 5 11:22:14 sshd[28759]: Disconnecting: Too many authentication failures for root from 201.174.19.78 port 37110 ssh2 [preauth]
Jul 5 11:22:15 sshd[28761]: Disconnecting: Too many authentication failures for root from 201.174.19.78 port 37112 ssh2 [preauth]
Jul 5 11:22:17 sshd[28763]: Disconnecting: Too many authentication failures for invalid user admin from 201.174.19.78 port 37118 ssh2 [preauth]
Jul 5 11:22:22 sshd[28765]: Disconnecting: Too many authentication failures for invalid user usuario from 201.174.19.78 port 37126 ssh2 [preauth]
Jul 5 11:22:29 sshd[28767]: Disconnecting: Too many authentication failures for root from 201.174.19.78 port 37148 ssh2 [preauth]
Jul 5 11:22:45 sshd[28769]: Disconnecting: Too many authentication failures for invalid user admin from 201.174.19.78 port 37151 ssh2 [preauth]
Jul 5 12:02:03 sshd[2952]: Disconnecting: Too many authentication failures for root from 5.239.96.130 port 60054 ssh2 [preauth]
Jul 5 12:38:16 sshd[8897]: Disconnecting: Too many authentication failures for invalid user support from 183.250.89.39 port 31638 ssh2 [preauth]
Jul 5 13:25:13 sshd[17028]: Disconnecting: Too many authentication failures for invalid user admin from 42.59.189.251 port 34105 ssh2 [preauth]
Jul 5 15:04:32 sshd[568]: Disconnecting: Too many authentication failures for invalid user admin from 81.174.255.65 port 38574 ssh2 [preauth]
Jul 5 15:28:36 sshd[4774]: Disconnecting: Too many authentication failures for root from 180.214.188.178 port 37887 ssh2 [preauth]
Jul 5 15:54:23 sshd[5036]: Disconnecting: Too many authentication failures for root from 222.180.67.177 port 35863 ssh2 [preauth]
Jul 5 15:56:17 sshd[5865]: Disconnecting: Too many authentication failures for invalid user service from 131.150.120.262 port 51524 ssh2 [preauth]
Jul 5 16:18:07 sshd[13212]: Disconnecting: Too many authentication failures for root from 37.76.148.61 port 44090 ssh2 [preauth]
Jul 5 17:07:58 sshd[21382]: Disconnecting: Too many authentication failures for root from 170.248.70.188 port 14625 ssh2 [preauth]
```

Сотни неудачных попыток авторизации, хотя на сервере вообще отключена авторизация по паролю и он работает на нестандартном порту

Wordpress нас приговорил

Ладно, признаем, сканеры веб-уязвимостей существовали и до Wordpress, но после того, как эта платформа стала настолько популярной, большинство сканеров начали проверять неправильно сконфигурированные папки wp-admin и непропатченные плагины.

Так что если маленькая начинающая хакерская банда хочет получить немного свеженьких учёток, они скачают [один из этих](#) сканерских инструментов и натравят его на кучу веб-сайтов в надежде получить доступ к какому-нибудь сайту и [дефейснуть](#) его.


```

if (ob_get_level()) ob_end_clean();
//send the gzipped file to the client
readfile('10G.zip');

```

Готово!

Теперь мы можем использовать её в качестве простой защиты:

```

<?php
$agent = filter_input(INPUT_SERVER, 'HTTP_USER_AGENT');

//check for nikto, sql map or "bad" subfolders which only exist on wordpress
if (strpos($agent, 'nikto') !== false || strpos($agent, 'sqlmap') !== false ||
startswith($url, 'wp-') || startswith($url, 'wordpress') || startswith($url, 'wp/'))
{
    sendBomb();
    exit();
}

function sendBomb(){
    //prepare the client to receive GZIP data. This will not be suspicious
    //since most web servers use GZIP by default
    header("Content-Encoding: gzip");
    header("Content-Length: ".filesize('10G.zip'));
    //Turn off output buffering
    if (ob_get_level()) ob_end_clean();
    //send the gzipped file to the client
    readfile('10G.zip');
}

function startswith($a, $b) {
    return strpos($a, $b) === 0;
}

```

Очевидно, этот скрипт не образец элегантности, но он может защитить нас от скрипт-кидди, упомянутых раньше, которые вообще понятия не имеют, что в сканерах можно изменять user-agent.

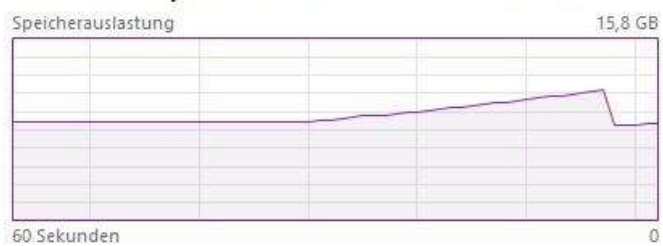
Итак... Что будет, если запустить этот скрипт?

Клиент	Результат
IE 11	Память расходуется, IE падает
Chrome	Память расходуется, демонстрируется ошибка
Edge	Память расходуется, утекает, грузится вечно
Nikto	Как будто нормально сканирует, но не выдаёт результат
SQLmap	Большой расход памяти, затем падает
Safari	Большой расход памяти, затем падает и перезагружается, затем опять большой расход памяти и так далее...
Chrome (Android)	Память расходуется, демонстрируется ошибка

(если вы проверяли бомбу на других устройствах/браузерах/скриптах, пожалуйста, [сообщите мне](#), и я добавлю результат в таблицу)

Arbeitsspeicher

16,0 GB Other



Результат загрузки скрипта в Chrome

Если вам нравится рисковать, [попробуйте сами!](#)

—

07/06/2017 23:32:06

0

habr Аноним

комментарии (0)