2021

Basic WordPress security

ARTICLE I

SUHAIRY BIN SUBORI

Contents

Introduction

A month ago, I had this idea of self-hosting my own WordPress website because I do not really like the idea of getting nickel and dimed by VPS companies when I had my own unused Server at home.

Was this a great idea? Absolutely. But am I prepared for the aftermath? Not in the slightest sense.

My initial security was lacklustre. I went ahead with the thought that all that matters was just a strong password and I am done for the day. But nope :D

Woke up from my short nap at 11pm and I saw my router LEDs blinking non-stop. Thought it might be due to someone in my family downloading some huge files but nope. Curious I logged into my web server and did:

```
lsof -i
```

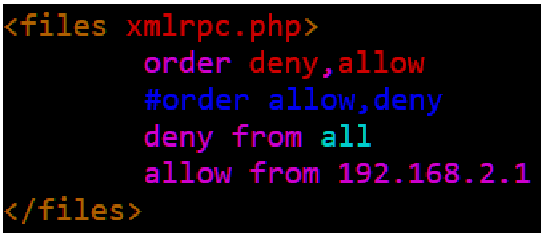
Why are there like several connections to my server from the same IP?? Okay, it might be from someone reading my blog and I will give him the benefit of doubt.

Several minutes passed and my router's light blinks as if it is on a 24km route march. Really… I head over to **/var/www/html** and checked **access.log**

Right…. So much for the `benefit of doubt`. It dawned upon me that someone is trying to hack into my web server! Pretty sure that multiple post to **xmlrpc.php** is the work of **wpscan**. I was anxious and excited at the same time.

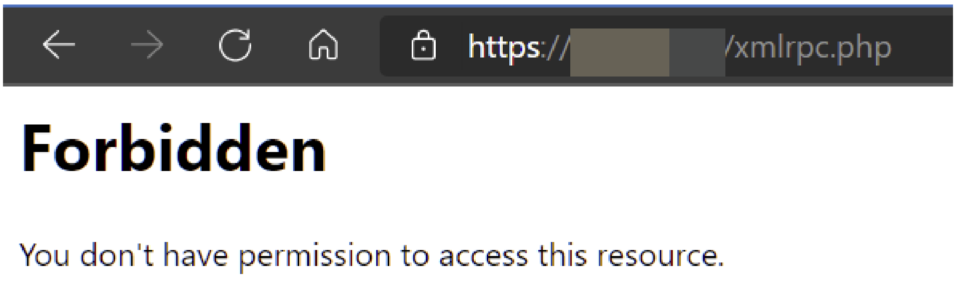
Blocking xmlrpc.php

Ok first course of action, lets block **xmlrpc.php**.



```
<files xmlrpc.php>
        order deny,allow
        #order allow,deny
        deny from all
        allow from 192.168.2.1
</files>
```

Figure 1 – blocking xmlrpc.php from public



https://[redacted]/xmlrpc.php

Forbidden

You don't have permission to access this resource.

Figure 2 – Resulting error when public tries to access xmlrpc.php

Was thinking that this will stop the attack on its tracks. Several minutes passed…

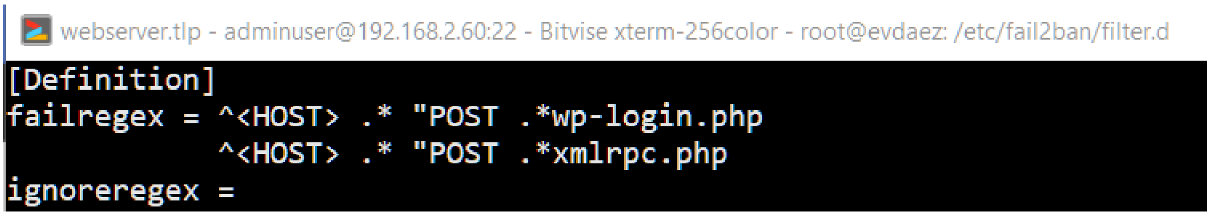
Why are my router LEDs still blinking like a disco ball?

Went to check **access.log** again and found that **wp-login.php** is being targeted. My mind's racing at this moment trying to think of a countermeasure.

Implementing fail2ban

And then an idea came across my mind, **fail2ban**. I have read many posts on **r/homelab** on reddit of people using that software.

Installed fail2ban, head to **/etc/fail2ban/filter.d** and proceed to create a filter named **wordpress.conf**

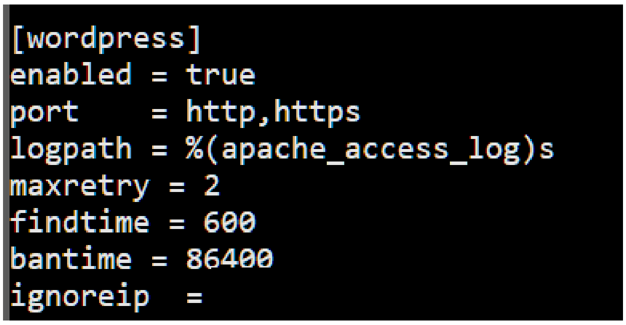


webserver.tlp - adminuser@192.168.2.60:22 - Bitvise xterm-256color - root@evdaez: /etc/fail2ban/filter.d

```
[Definition]
failregex = ^<HOST> .* "POST .*wp-login.php
            ^<HOST> .* "POST .*xmlrpc.php
ignoreregex =
```

Figure 3 – Criteria used to determine ban

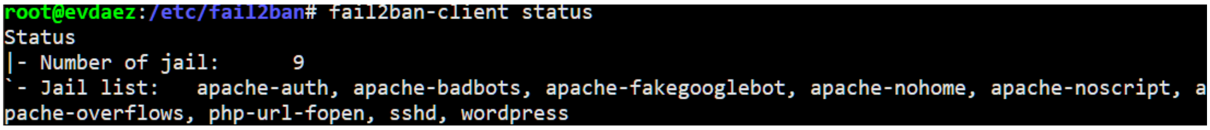
I create a jail entry in **/etc/fail2ban/jail.local** and restarted fail2ban.



```
[wordpress]
enabled = true
port     = http,https
logpath = %(apache_access_log)s
maxretry = 2
findtime = 600
bantime = 86400
ignoreip  =
```

Figure 4 – WordPress jail

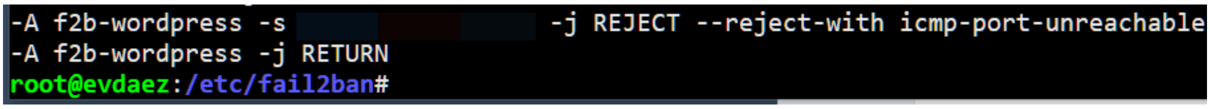
Then I checked if the WordPress jail is active.



```
root@evdaez:/etc/fail2ban# fail2ban-client status
Status
|- Number of jail:      9
`- Jail list:   apache-auth, apache-badbots, apache-fakegooglebot, apache-nohome, apache-noscript, a
pache-overflows, php-url-fopen, sshd, wordpress
```

Figure 5 – Checking whether WordPress jail is active.

Less than 15 seconds later, the hacker was banned.

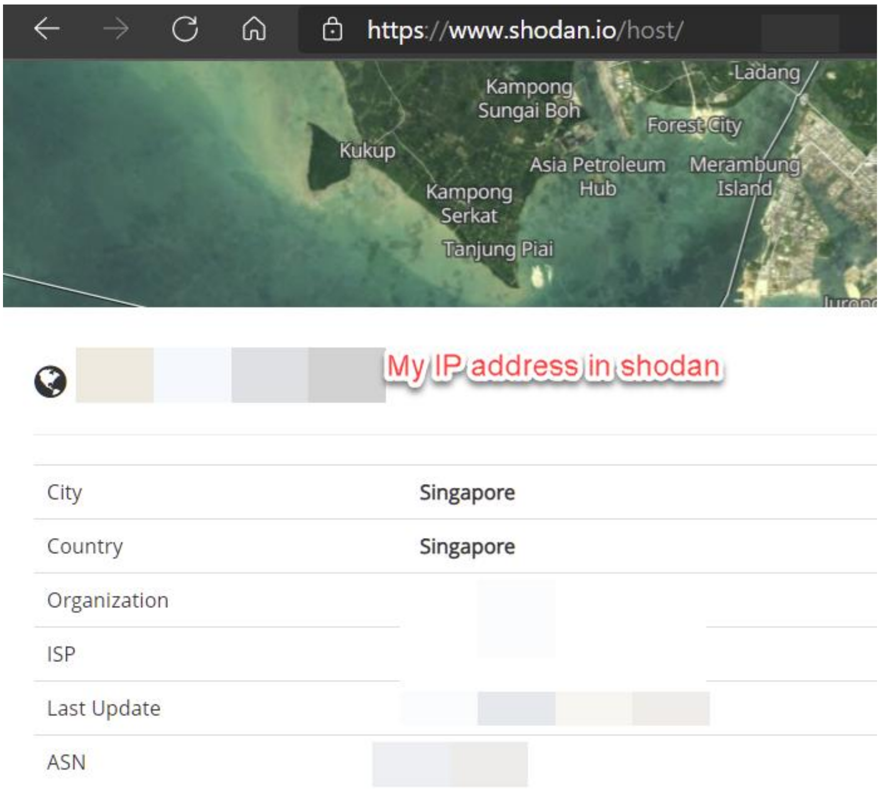


```
-A f2b-wordpress -s [redacted] -j REJECT --reject-with icmp-port-unreachable
-A f2b-wordpress -j RETURN
root@evdaez:/etc/fail2ban#
```

Figure 6 – Banned after multiple failed logins.

Webserver appearing on Shodan

By this time onwards, there are no more `blinking` LEDs but then my IP address decides that it wants to be famous.



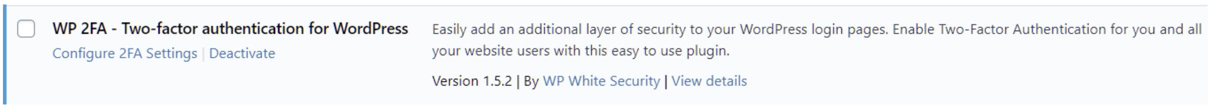
[redacted] My IP address in shodan

City	Singapore
Country	Singapore
Organization	[redacted]
ISP	[redacted]
Last Update	[redacted]
ASN	[redacted]

Figure 7 – IP address in Shodan

Countermeasures implemented

Great… things are starting to get serious. I installed **2FA** plugin as a 2nd line of defence.



☐ WP 2FA - Two-factor authentication for WordPress

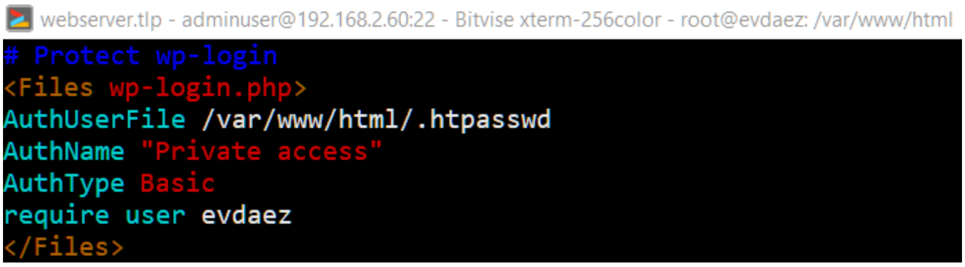
Configure 2FA Settings | Deactivate

Easily add an additional layer of security to your WordPress login pages. Enable Two-Factor Authentication for you and all your website users with this easy to use plugin.

Version 1.5.2 | By WP White Security | View details

Figure 8 – Installed 2FA plugin in WordPress

On top of that, i protect my login page using apache's basic authentication. It is a hassle. Yes, but at times, you must do what it takes to make yourself sleep better at night.

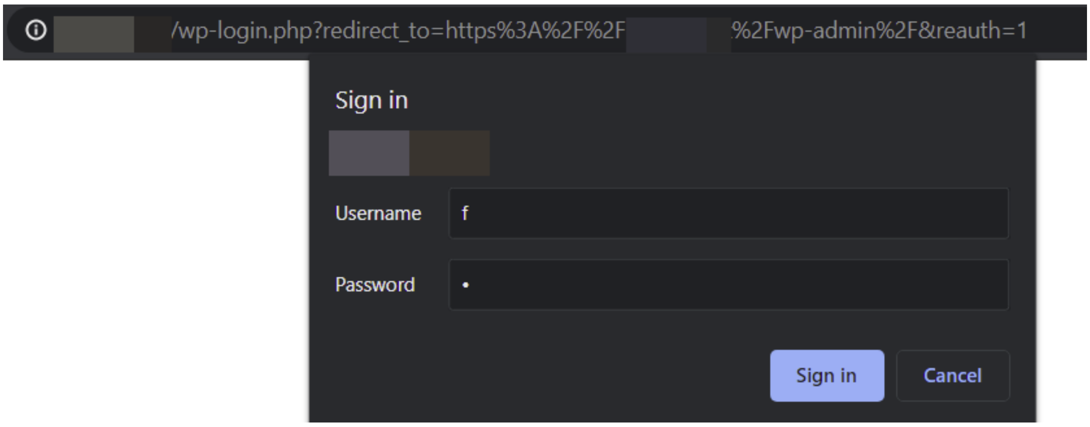


webserver.tlp - adminuser@192.168.2.60:22 - Bitvise xterm-256color - root@evdaez: /var/www/html

```
# Protect wp-login
<Files wp-login.php>
AuthUserFile /var/www/html/.htpasswd
AuthName "Private access"
AuthType Basic
require user evdaez
</Files>
```

Figure 9 – Apache basic authentication configuration

Implementing the above configuration ensures that there is a popup when someone tries to access my login page. This is the 3rd layer of defence.



ⓘ [redacted] /wp-login.php?redirect_to=https%3A%2F%2F [redacted] %2Fwp-admin%2F&reauth=1

Sign in

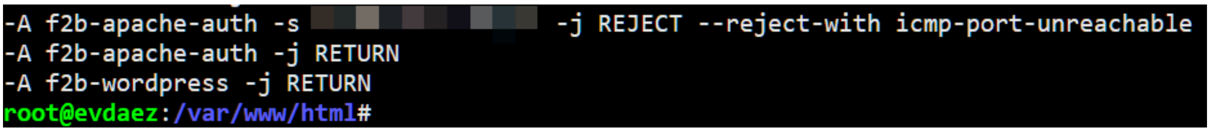
[redacted]

Username: f

Password: •

Figure 10 – Apache login prompt

This complements fail2ban in a way because if someone tries to brute force apache's basic authentication, they will get banned.

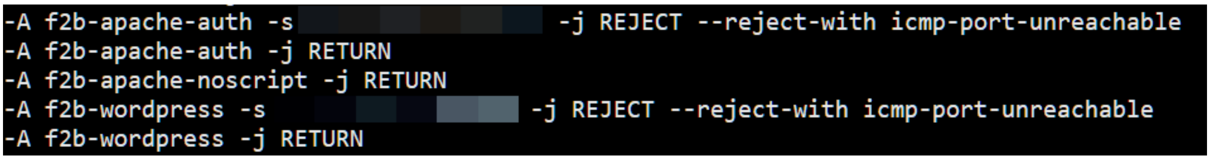


```
-A f2b-apache-auth -s [redacted]  -j REJECT --reject-with icmp-port-unreachable
-A f2b-apache-auth -j RETURN
-A f2b-wordpress -j RETURN
root@evdaez:/var/www/html#
```

Figure 11 – Fail2ban doing its job

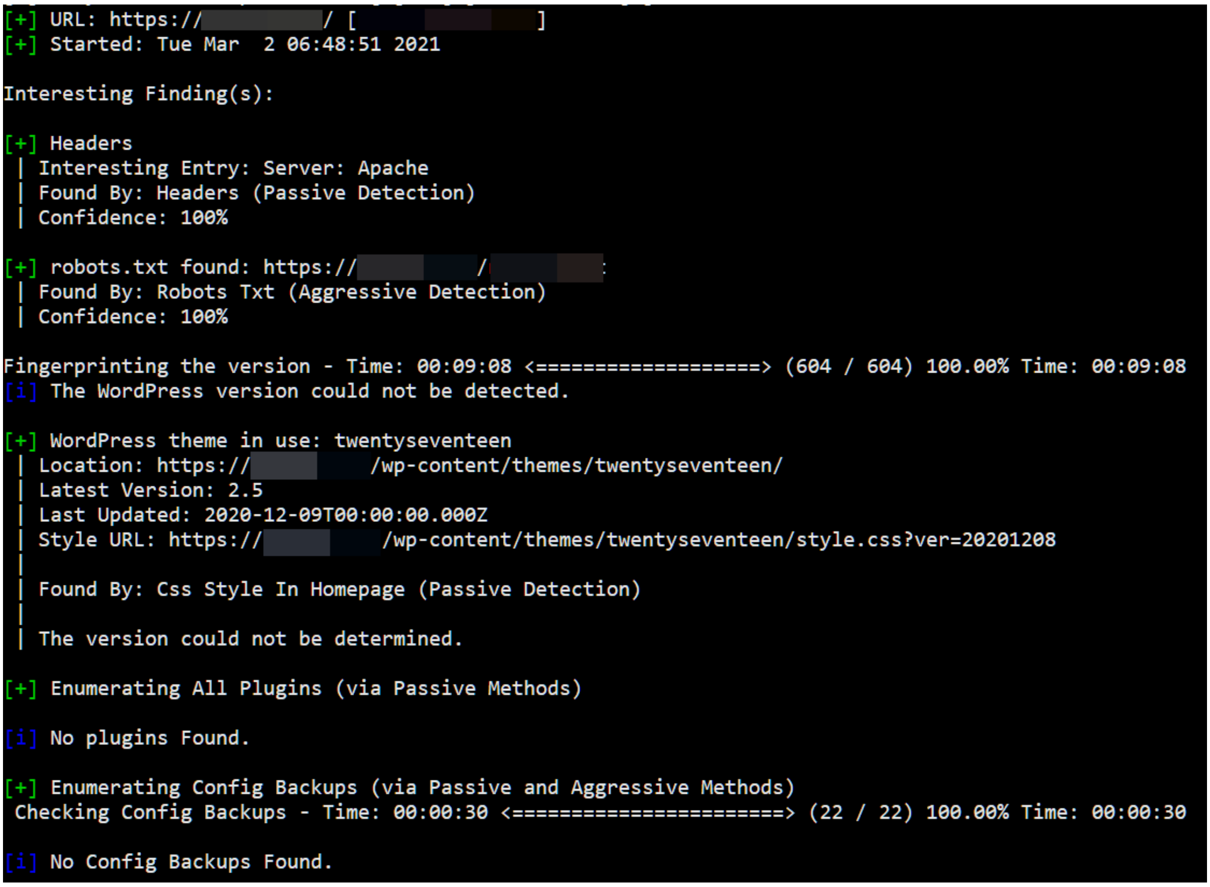
Using wpscan to look for holes

Although fail2ban is a bit overzealous by banning my VPS IP(had to unban my IP multiple times) when I conducted a scan over the internet. I could say that the result is pretty much satisfactory when there is not much information to derive from wpscan results.



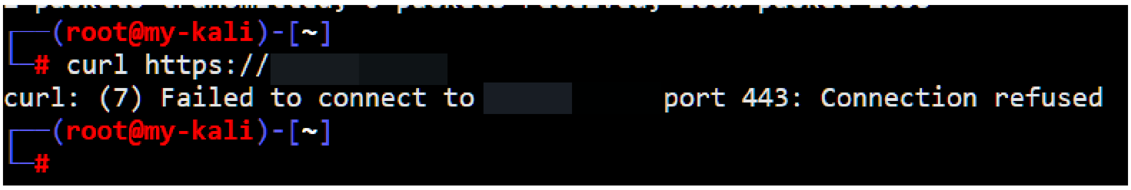
```
-A f2b-apache-auth -s [redacted] -j REJECT --reject-with icmp-port-unreachable
-A f2b-apache-auth -j RETURN
-A f2b-apache-noscript -j RETURN
-A f2b-wordpress -s [redacted] -j REJECT --reject-with icmp-port-unreachable
-A f2b-wordpress -j RETURN
```

Figure 12 – Running aggressive wpscan will result in attacker's IP getting blocked.



```
[+] URL: https://[redacted]/ [[redacted]]
[+] Started: Tue Mar  2 06:48:51 2021

Interesting Finding(s):

[+] Headers
 | Interesting Entry: Server: Apache
 | Found By: Headers (Passive Detection)
 | Confidence: 100%

[+] robots.txt found: https://[redacted]/[redacted]
 | Found By: Robots Txt (Aggressive Detection)
 | Confidence: 100%

Fingerprinting the version - Time: 00:09:08 <===================> (604 / 604) 100.00% Time: 00:09:08
[i] The WordPress version could not be detected.

[+] WordPress theme in use: twentyseventeen
 | Location: https://[redacted]/wp-content/themes/twentyseventeen/
 | Latest Version: 2.5
 | Last Updated: 2020-12-09T00:00:00.000Z
 | Style URL: https://[redacted]/wp-content/themes/twentyseventeen/style.css?ver=20201208
 |
 | Found By: Css Style In Homepage (Passive Detection)
 |
 | The version could not be determined.

[+] Enumerating All Plugins (via Passive Methods)

[i] No plugins Found.

[+] Enumerating Config Backups (via Passive and Aggressive Methods)
 Checking Config Backups - Time: 00:00:30 <=======================> (22 / 22) 100.00% Time: 00:00:30

[i] No Config Backups Found.
```

Figure 13 – wpscan results



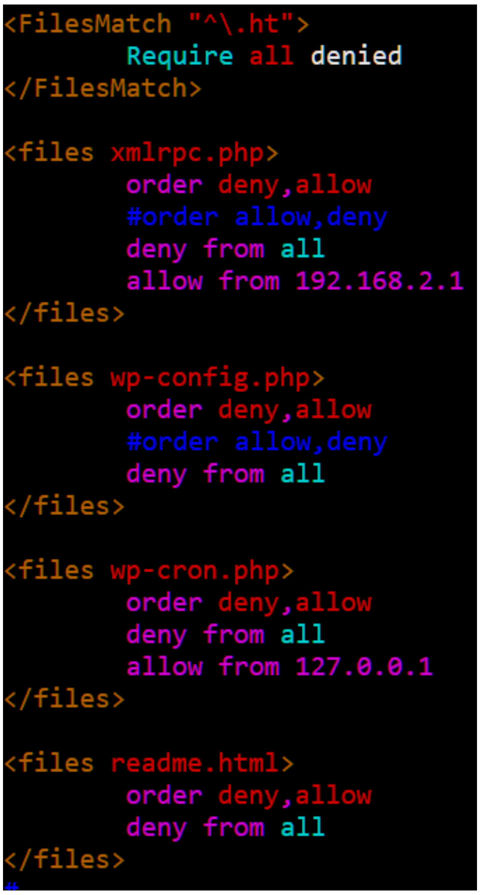
```
┌──(root㉿my-kali)-[~]
└─# curl https://[redacted]
curl: (7) Failed to connect to [redacted]       port 443: Connection refused
┌──(root㉿my-kali)-[~]
└─#
```

Figure 14 – Blocked IP for aggressive scan(On VPS)

Wrapping it up

My IP address is still in Shodan. Guess it loves being in the spotlight… However, with all the steps taken and implementing **WAF**(will be discussed in another article), I do think that I provide my webserver with a basic degree of security.

Apache configuration file working in tandem with fail2ban works wonders in my opinion.



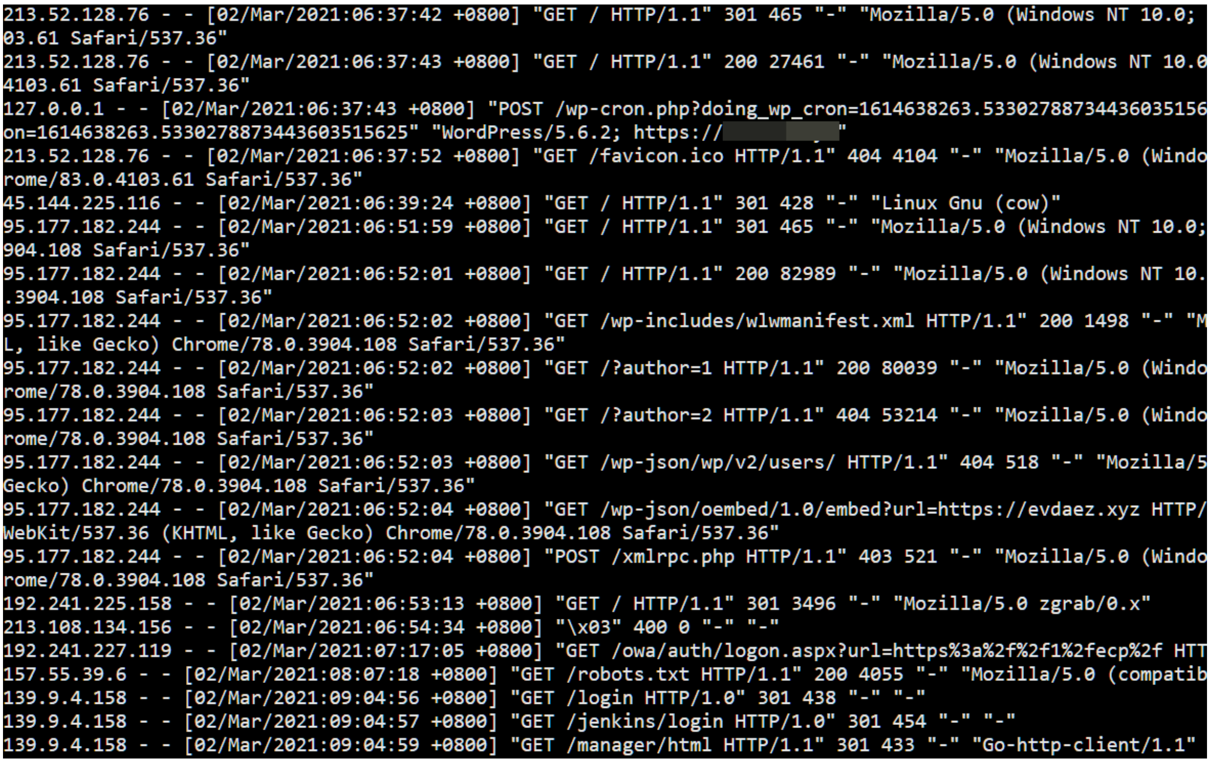
```
<FilesMatch "^\.ht">
        Require all denied
</FilesMatch>

<files xmlrpc.php>
        order deny,allow
        #order allow,deny
        deny from all
        allow from 192.168.2.1
</files>

<files wp-config.php>
        order deny,allow
        #order allow,deny
        deny from all
</files>

<files wp-cron.php>
        order deny,allow
        deny from all
        allow from 127.0.0.1
</files>

<files readme.html>
        order deny,allow
        deny from all
</files>
```

Figure 15 – Apache configuration file

Conclusion

Are things so much better after doing all those countermeasures?

A little, I guess. There are always random bots trying random stuff here and there but hey no more blinking lights!



```
213.52.128.76 - - [02/Mar/2021:06:37:42 +0800] "GET / HTTP/1.1" 301 465 "-" "Mozilla/5.0 (Windows NT 10.0;
03.61 Safari/537.36"
213.52.128.76 - - [02/Mar/2021:06:37:43 +0800] "GET / HTTP/1.1" 200 27461 "-" "Mozilla/5.0 (Windows NT 10.0
4103.61 Safari/537.36"
127.0.0.1 - - [02/Mar/2021:06:37:43 +0800] "POST /wp-cron.php?doing_wp_cron=1614638263.5330278873443603515
on=1614638263.5330278873443603515625" "WordPress/5.6.2; https://[redacted]"
213.52.128.76 - - [02/Mar/2021:06:37:52 +0800] "GET /favicon.ico HTTP/1.1" 404 4104 "-" "Mozilla/5.0 (Windo
rome/83.0.4103.61 Safari/537.36"
45.144.225.116 - - [02/Mar/2021:06:39:24 +0800] "GET / HTTP/1.1" 301 428 "-" "Linux Gnu (cow)"
95.177.182.244 - - [02/Mar/2021:06:51:59 +0800] "GET / HTTP/1.1" 301 465 "-" "Mozilla/5.0 (Windows NT 10.0;
904.108 Safari/537.36"
95.177.182.244 - - [02/Mar/2021:06:52:01 +0800] "GET / HTTP/1.1" 200 82989 "-" "Mozilla/5.0 (Windows NT 10.
.3904.108 Safari/537.36"
95.177.182.244 - - [02/Mar/2021:06:52:02 +0800] "GET /wp-includes/wlwmanifest.xml HTTP/1.1" 200 1498 "-" "M
L, like Gecko) Chrome/78.0.3904.108 Safari/537.36"
95.177.182.244 - - [02/Mar/2021:06:52:02 +0800] "GET /?author=1 HTTP/1.1" 200 80039 "-" "Mozilla/5.0 (Windo
rome/78.0.3904.108 Safari/537.36"
95.177.182.244 - - [02/Mar/2021:06:52:03 +0800] "GET /?author=2 HTTP/1.1" 404 53214 "-" "Mozilla/5.0 (Windo
rome/78.0.3904.108 Safari/537.36"
95.177.182.244 - - [02/Mar/2021:06:52:03 +0800] "GET /wp-json/wp/v2/users/ HTTP/1.1" 404 518 "-" "Mozilla/5
Gecko) Chrome/78.0.3904.108 Safari/537.36"
95.177.182.244 - - [02/Mar/2021:06:52:04 +0800] "GET /wp-json/oembed/1.0/embed?url=https://evdaez.xyz HTTP/
WebKit/537.36 (KHTML, like Gecko) Chrome/78.0.3904.108 Safari/537.36"
95.177.182.244 - - [02/Mar/2021:06:52:04 +0800] "POST /xmlrpc.php HTTP/1.1" 403 521 "-" "Mozilla/5.0 (Windo
rome/78.0.3904.108 Safari/537.36"
192.241.225.158 - - [02/Mar/2021:06:53:13 +0800] "GET / HTTP/1.1" 301 3496 "-" "Mozilla/5.0 zgrab/0.x"
213.108.134.156 - - [02/Mar/2021:06:54:34 +0800] "\x03" 400 0 "-" "-"
192.241.227.119 - - [02/Mar/2021:07:17:05 +0800] "GET /owa/auth/logon.aspx?url=https%3a%2f%2f1%2fecp%2f HTT
157.55.39.6 - - [02/Mar/2021:08:07:18 +0800] "GET /robots.txt HTTP/1.1" 200 4055 "-" "Mozilla/5.0 (compatib
139.9.4.158 - - [02/Mar/2021:09:04:56 +0800] "GET /login HTTP/1.0" 301 438 "-" "-"
139.9.4.158 - - [02/Mar/2021:09:04:57 +0800] "GET /jenkins/login HTTP/1.0" 301 454 "-" "-"
139.9.4.158 - - [02/Mar/2021:09:04:59 +0800] "GET /manager/html HTTP/1.1" 301 433 "-" "Go-http-client/1.1"
```

Figure 16 - Bots out in full force