Mikey Veenstra

February 27, 2020

Site Takeover Campaign Exploits Multiple Zero-Day Vulnerabilities

Early yesterday, the Flexible Checkout Fields for WooCommerce plugin received a critical update to patch a zero-day vulnerability which allowed attackers to modify the plugin's settings. As our Threat Intelligence team researched the scope of this attack campaign, we discovered **three additional zero-day vulnerabilities** in popular WordPress plugins that are being exploited as a part of this campaign. The targeted plugins were Async JavaScript, Modern Events Calendar Lite, and 10Web Map Builder for Google Maps. At this time, we have reached out to each plugin's development team in hopes of getting these issues resolved quickly.

This attack campaign exploits XSS vulnerabilities in the above plugins to inject malicious Javascript that can create rogue WordPress administrators and install malicious plugins that include backdoors. It is important that site administrators using these plugins urgently take steps to mitigate these attacks.

We have released firewall rules to protect Wordfence users from these attacks. These rules are already available to Wordfence Premium users and will be available to sites still on the free version in thirty days. Fortunately, because these vulnerabilities are being exploited to inject XSS payloads, those attacks have been blocked by the built-in XSS protection available to all Wordfence users, free or Premium. However, the nature of each vulnerability allows other disruptive activity that would not be blocked by these protections, necessitating the additional firewall rules.

Today's post gives an overview of these vulnerabilities to inform the community of their current risk. More details of this campaign will be considered in a forthcoming blog post.

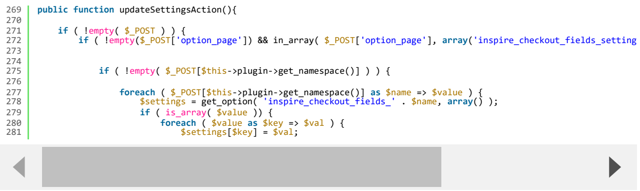
Unauthenticated Stored XSS in Flexible Checkout Fields For WooCommerce

Description: Unauthenticated Stored XSS via Plugin Settings Change
Affected Plugin: Flexible Checkout Fields for WooCommerce
Affected Versions: <= 2.3.1
CVSS Score: 9.3 (Critical)
CVSS Vector: CVSS:3.0/AV:N/AC:L/PR:N/UI:R/S:C/C:H/I:H/A:N
Patched Version: 2.3.2

In a report released yesterday by NinTechNet, researchers alerted the community to the presence of this vulnerability and the attacks against it. Unauthenticated attackers are capable of modifying the plugin's options, which can be leveraged to inject XSS payloads that can be triggered in the dashboard of a logged-in administrator.

Flexible Checkout Fields versions up to 2.3.1 are vulnerable to these attacks. The plugin's developers, WP Desk, issued a patch with version 2.3.2 quickly after they were made aware of the issue. Since then, they've issued two more updates to implement some additional security measures. The WordPress.org repository reports an install base of more than 20,000 sites with the plugin. We urge all of the plugin's users to update to the latest available version as quickly as possible to reduce their risk of exploitation.

This vulnerability was due to a lack of capabilities checks on the plugin's settings update function.
In `classes/settings.php`, the function `updateSettingsAction()` is hooked into the WordPress `admin_init` hook. This hook fires when any `/wp-admin/` endpoint is accessed, including those that don't require authentication.



```
public function updateSettingsAction(){

    if ( !empty( $_POST ) ) {
        if ( !empty($_POST['option_page']) && in_array( $_POST['option_page'], array('inspire_checkout_fields_setting


            if ( !empty( $_POST[$this->plugin->get_namespace()] ) ) {

                foreach ( $_POST[$this->plugin->get_namespace()] as $name => $value ) {
                    $settings = get_option( 'inspire_checkout_fields_' . $name, array() );
                    if ( is_array( $value )) {
                        foreach ( $value as $key => $val ) {
                            $settings[$key] = $val;
```

The snippet above shows the first several lines of the function, which makes some checks for certain `$_POST` values but no security checks. By crafting an array of expected settings, attackers can inject JavaScript payloads into the elements that render onscreen.

This vulnerability was patched by the plugin developers by implementing a capabilities check to ensure only administrators can modify these settings.

Subscriber+ Stored XSS in Async JavaScript

Description: Subscriber+ Stored XSS via Plugin Settings Change
Affected Plugin: Async JavaScript
Affected Versions: <= 2.19.07.14
CVSS Score: 7.6 (High)
CVSS Vector: CVSS:3.0/AV:N/AC:L/PR:L/UI:R/S:C/C:L/I:H/A:N
Patched Version: 2.20.02.27

A similar vulnerability exists in the popular Async JavaScript Plugin, which is currently active on more than 100,000 WordPress sites. We notified the plugin's developer, Frank Goossens, who quickly released a patch for this issue. Because the update was made available so recently, we are providing limited details about the vulnerability at this time.

Async JavaScript's settings are modified via calls to `wp-admin/admin-ajax.php` with the action `aj_steps`. This AJAX action is registered only for authenticated users, but no capabilities checks are made. Because of this, low-privilege users including Subscribers can modify the plugin's settings.

Similar to Flexible Checkout Fields above, certain setting values can be injected with a crafted payload to execute malicious JavaScript when a WordPress administrator views certain areas of their dashboard.

Unauthenticated Stored XSS in 10Web Map Builder for Google Maps

Description: Unauthenticated Stored XSS via Plugin Settings Change
Affected Plugin: 10Web Map Builder for Google Maps
Affected Versions: <= 1.0.63
CVSS Score: 9.3 (Critical)
CVSS Vector: CVSS:3.0/AV:N/AC:L/PR:N/UI:R/S:C/C:H/I:H/A:N
Patched Version: 1.0.64

A third XSS via settings change vulnerability is present in 10Web Map Builder for Google Maps. This plugin is active on over 20,000 sites. Unlike Async JavaScript, this vulnerability can be exploited by unauthenticated attackers.

We have reached out to establish contact with the plugin's developers and are awaiting their response at this time. As with the previous vulnerability, because it's under attack in the wild we are providing limited detail.

The vulnerability in 10Web Map Builder exists in the plugin's setup process. The plugin's setup functions are called during `admin_init` which, like Flexible Checkout Fields, is accessible to unauthenticated users. If an attacker injects malicious JavaScript into certain settings values, that code will execute for administrators in their dashboard as well as front-of-site visitors in some circumstances.

Update 02/28/2020

10Web Map Builder for Google Maps has now been patched to resolve this issue. We urge users to update to version 1.0.64 as soon as possible.

Multiple Subscriber+ Stored XSS Vulnerabilities In Modern Events Calendar Lite

Description: Multiple Subscriber+ Stored XSS Vulnerabilities
Affected Plugin: Modern Events Calendar Lite
Affected Versions: <= 5.1.6
CVSS Score: 7.6 (High)
CVSS Vector: CVSS:3.0/AV:N/AC:L/PR:L/UI:R/S:C/C:L/I:H/A:N
CVE ID: CVE-2020-9459
Patched Version: 5.1.7

The last vulnerability in this report affects Modern Events Calendar Lite, with over 40,000 installs.

We have reached out to establish contact with the developer and are awaiting response. Again, as this issue is known to malicious actors, we are making the community aware of it.

Modern Events Calendar Lite registers a number of AJAX actions for logged-in users. Some of these actions allow low-privileged users like subscribers to manipulate settings and other stored data. When exploited in this way, the affected data can be injected with various XSS payloads.

Depending on where the attackers inserted code, these scripts can be executed in the WordPress dashboard to affect administrators, or on the front of the victim's site to affect their visitors. Current attacks in this campaign are targeting administrators in order to create rogue accounts for the attackers.

Update 02/29/2020

Modern Events Calendar Lite has now been patched to resolve this issue. Update to version 5.1.7 as soon as possible.

Conclusion

Today we disclosed three new zero-day vulnerabilities affecting the WordPress ecosystem. We are working to assist these developers to quickly resolve these vulnerabilities, but some remain unpatched at this time. We take the security disclosure process very seriously, and we would not publish these details if it wasn't necessary to alert the WordPress community about their risk in the midst of this campaign.

The XSS attacks used in this campaign are reliably blocked by the Wordfence firewall's built-in protections, which are available to Wordfence Premium users as well as sites still on the free version. New WAF rules to prevent other disruptive activity are also available to Premium users at this time.

Because these attacks are ongoing, research into this campaign is still underway. We will publish a follow-up post with complete details on these attacks as soon as this research is complete. Make sure you are informed as soon as possible by subscribing to our mailing list.

This work would not be possible without the combined efforts of the Wordfence team. Special thanks to Director of Threat Intelligence Sean Murphy, QA Lead Matt Rusnak, and QA Engineer Ramuel Gall for their contributions to the discovery and research of these attacks, analysis and disclosure of the vulnerabilities, and assistance in editing this post.

Did you enjoy this post? Share it!

Comments

3 Comments

Kadigan *
February 28, 2020
1:36 am

Is it just me, or is there a very noticeable trend here? Rather than bypassing security measures put in place, 99% of the attacks I've seen so far seem to revolve around there being no security at all, for certain actions!

Maybe it's time we went a different path altogether? Automatically fence settings pages to admin-only UNLESS the plugin author specifies a lesser privilege requirement, for instance? Automatically elevate and re-verify required privileges if certain functions are called or registered?

susan.berdinka *
March 3, 2020
6:03 pm

ThemeRx pushed out a patch. Does it solve the 0-day vulnerability?

Chloe Chamberland *
March 4, 2020
8:05 am

At this time, please reach out to the ThemeREX team to verify that your particular theme installation is secured. We'll have more details in the coming days.

Our business hours are 9am-8pm ET, 6am-5pm PT and 2pm-1am UTC/GMT excluding weekends and holidays.
Response customers receive 24-hour support, 365 days a year, with a 1-hour response time.