# Security Issues Patched in Smash Balloon Social Post Feed Plugin

*Updated on October 29, 2021 - Marc Montpas*

During an internal audit of the Smash Balloon Social Post Feed plugin (also known as Custom Facebook Feed), we discovered several sensitive AJAX endpoints were accessible to any users with an account on the vulnerable site, like subscribers. Some of these endpoints could enable Stored Cross-Site Scripting (XSS) attacks to occur.

A successful Stored XSS attack could enable bad actors to store malicious scripts on every post and page of the affected site. If a logged-in administrator visits one of the affected URLs, the script may run on their browser and execute administrative actions on their behalf, like creating new administrators and installing rogue plugins.

We reported the vulnerabilities to this plugin's author via email, and they recently released version 4.0.1 to address them. We strongly recommend that you update to the latest version of the Smash Balloon Social Post Feed plugin and have an established security solution on your site, such as Jetpack Security.

## Details

**Plugin Name:** Smash Balloon Social Post Feed
**Plugin URI:** https://wordpress.org/plugins/custom-facebook-feed/
**Author:** Smash Balloon
**Author URI:** https://smashballoon.com/

## The Vulnerabilities

### Stored Cross-Site Scripting via Arbitrary Setting Update

**Affected versions:** < 4.0.1
**CVE-ID:** CVE-2021-24918
**CVSSv3.1:** 7.3
**CWSS:** 80.6

```
public function cff_save_settings() {
        $data = $_POST;
        $model = isset( $data[ 'model' ] ) ? $data['model'] : null;
        // return if the model is null
        if ( null === $model ) {
                return;
        }

        // (...)

        $model = (array) \json_decode( \stripslashes( $model ) );
        $general = (array) $model['general'];
        $feeds = (array) $model['feeds'];
        $translation = (array) $model['translation'];
        $advanced = (array) $model['advanced'];
        // Get the values and sanitize
        $cff_locale                                             = sanitize_text_field( $feeds['selectedLocale'] );
        $cff_style_settings                             = get_option( 'cff_style_settings' );
        $cff_style_settings[ 'cff_timezone' ]   = sanitize_text_field( $feeds['selectedTimezone'] );
        $cff_style_settings[ 'cff_custom_css' ] = $feeds['customCSS'];
        $cff_style_settings[ 'cff_custom_js' ]  = $feeds['customJS'];
        $cff_style_settings[ 'gdpr' ]                   = sanitize_text_field( $feeds['gdpr'] );
        $cachingType                                            = sanitize_text_field( $feeds['cachingType'] );
        $cronInterval                                           = sanitize_text_field( $feeds['cronInterval'] );
        $cronTime                                                       = sanitize_text_field( $feeds['cronTime'] );
        $cronAmPm                                                       = sanitize_text_field( $feeds['cronAmPm'] );
        $cacheTime                                                      = sanitize_text_field( $feeds['cacheTime'] );
        $cacheTimeUnit                                          = sanitize_text_field( $feeds['cacheTimeUnit'] );
        // Save general settings data
        update_option( 'cff_preserve_settings', $general['preserveSettings'] );
        // Save feeds settings data
        update_option( 'cff_locale', $cff_locale );

        // (...)

        // Save translation settings data
        foreach( $translation as $key => $val ) {
                $cff_style_settings[ $key ] = $val;
        }
        // Save advanced settings data
        $cff_ajax = sanitize_text_field( $advanced['cff_ajax'] );
        foreach( $advanced as $key => $val ) {
                if ( $key == 'cff_disable_resize' || $key == 'disable_admin_notice' ) {
                        $cff_style_settings[ $key ] = !$val;
                } else {
                        $cff_style_settings[ $key ] = $val;
                }
        }

        // (...)

        update_option( 'cff_ajax', $cff_ajax );
        // Update the cff_style_settings option that contains data for translation and advanced tabs
        update_option( 'cff_style_settings', $cff_style_settings );
        // clear cron caches
        $this->cff_clear_cache();
        new CFF_Response( true, array(
                'cronNextCheck' => $this->get_cron_next_check()
        ) );
}
```

The `wp_ajax_cff_save_settings` AJAX action, which is responsible for updating the plugin's inner settings, did not perform any privilege or nonce checks before doing so. This made it possible for any logged-in users to call this action and update any of the plugin's settings.

Unfortunately, one of these settings, `customJS` , enables administrators to store custom JavaScript on their site's posts and pages. Updating this setting is all it would've taken for a bad actor to store malicious scripts on the site.

## Timeline

2021-10-14 – Initial contact with Smash Balloon
2021-10-18 – We send them details about these vulnerabilities
2021-10-21 – Smash Balloon Social Post Feed 4.0.1 is released

## Conclusion

We recommend that you check which version of the Smash Balloon Social Post Feed plugin your site is using, and if it is less than 4.0.1, update it as soon as possible!

At Jetpack, we work hard to make sure your websites are protected from these types of vulnerabilities. We recommend that you have a security plan for your site that includes malicious file scanning and backups. Jetpack Security is one great WordPress security option to ensure your site and visitors are safe.

## Credits

Original researcher: Marc Montpas

Thanks to the rest of the Jetpack Scan team for feedback, help, and corrections.

*This entry was posted in **Security**, **Vulnerabilities** and tagged **Jetpack**, **Security**, **Vulnerabilities**. Bookmark the **permalink**.*

## Marc Montpas

Marc's interests led him to work in the trenches of cybersecurity for the better part of the last decade, notably at companies like Sucuri and GoDaddy. His journey led him to uncover several high-impact security issues while auditing open-source platforms, like WordPress. He's an avid Hacker Capture The Flag player and loves to hypothesize new attack vectors.

## Have a question?

Comments are closed for this article, but we're still here to help! Visit the support forum and we'll be happy to answer any questions.

View support forum

Search

## Get news & tips from Jetpack

Enter your email address to follow this blog and receive news and updates from Jetpack!

Email Address

Subscribe

Join 111,148 other subscribers

## Browse by Topic

- Affiliates (1)
- Analytics (6)
- Code snippets (32)
- Contribute (6)
- Customer Stories (6)
- Ecommerce (11)
- Events (5)
- Features (56)
- Grow (11)
- hosting (1)
- Innovate (6)
- Jetpack News (45)
- Learn (65)
- Meet Jetpack (14)
- Performance (24)
- Photos & Videos (9)
- Promotions (2)
- Releases (166)
- Search Engine Optimization (12)
- Security (75)
- Small Business (16)
- Social Media (13)
- Support Stories (3)
- Tips & Tricks (85)
- Uncategorized (5)
- Utilities & Maintenance (4)
- Vulnerabilities (18)
- Website Design (13)
- WordAds (1)
- WordCamp (3)

Jetpack

EN

**WordPress Plugins**

Akismet Anti-spam

Jetpack

Jetpack Boost

Jetpack CRM

Jetpack Protect

Jetpack Search

Jetpack Social

Jetpack VideoPress

VaultPress Backup

WP Super Cache

**Partners**

Recommended Hosts

For Hosts

For Agencies

**Developers**

Documentation

Beta Program

Contribute to Jetpack

**Legal**

Terms of Service

Privacy Policy

GDPR

Privacy Notice for California Users

**Help**

Knowledge Base

Forums

Security Library

Contact Us

Press

**Social**

**Mobile Apps**

GET IT ON Google Play

Download on the App Store

An airline

Work With Us