

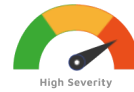
[← Back to all zero days](#)

## Reflected Cross-Site Scripting (XSS) in Thembay

AFFECTED  
VENDOR  
Thembay

STATUS  
Fixed

DATE  
Jul 17, 2020



[Description](#) [Proof of concept \(POC\)](#) [Impact](#) [Remediations](#) [Timeline](#)

### Description

A cross-site scripting (XSS) attack can cause arbitrary code (JavaScript) to run in a user's browser while the browser is connected to a trusted web site. The application targets your users and not the application itself, but it uses your application as the vehicle for the attack. XSS payload was executed when the user loads a malicious link generated using the ajax call back in Greenmart autocomplete search.

### Proof of concept: (POC)

The following vulnerability was tested on the Greenmart theme on WordPress with version 5.4.2.

#### Issue 01: Reflected cross-site scripting.

1. Install the Greenmart theme on WordPress with version 5.4.2.

Figure-01: The view-source of the WordPress application, which confirms the theme is Greenmart.

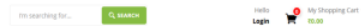


Figure-02: Greenmart search functionality

GET request to the administration page	Name	Value
term	admin	admin
term	admin	admin
term	admin	admin

Figure-03: The search action related backend ajax call

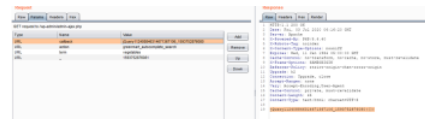


Figure-04: The ajax call to "greenmart\_autocomplete\_search" action and the response from the server

Request	Response
GET request to the administration page	term
term	admin
term	admin
term	admin

Figure-05: Call-back request parameter with payload and the response from the server.

2. Click on the following link [http://localhost/wordpress/wp-admin/admin-ajax.php?callback=-->%27><svg/onload=alert\(document.cookie\)>&action=greenmart\\_autocomplete\\_search&term=defaultText&\\_wpnonce=1593737670196](http://localhost/wordpress/wp-admin/admin-ajax.php?callback=-->%27><svg/onload=alert(document.cookie)>&action=greenmart_autocomplete_search&term=defaultText&_wpnonce=1593737670196)



Figure-06: The call-back parameter is vulnerable to Reflected XSS, and it's getting executed in the user browser context.



Figure-07: Wp-config configuration related to protecting XSS.

### Impact

When the user input from a URL or POST data is reflected on the page without being stored, thus allowing the attacker to inject malicious content. This means that an attacker has to send a crafted malicious URL or post form to the victim to insert the payload.

### Remediations

Replaced and protected the relevant code from the vendor.

July 18, 2020: Vendor Released Fixed

July 29, 2020: CVE Assigned

## Discovered by

Cyber Security Works Pvt. Ltd.

**Talk to CSW's team of experts to secure your landscape.**

[Schedule free consultation](#)



Cyber Security Works helps reduce security debt and inherent vulnerabilities in an organization's infrastructure and code. We work with large public, private, and start-up companies and help them prioritize their vulnerabilities.



[Sitemap](#) [Privacy Policy](#) [Customer Agreements](#)  
© 2022 - Cyber Security Works

## Resources

[Ransomware](#)  
[Cyber Risk Series](#)  
[Blogs](#)  
[Patch Watch](#)  
[Data Sheets](#)  
[White Papers](#)  
[Zero Days](#)  
[Glossary](#)  
[Events](#)  
[CISA-KEY](#)

## Partner

[Become a Partner](#)

## Quick Links

[About Us](#)  
[Contact Us](#)  
[Careers](#)  
[Services](#)  
[Media Coverage](#)  
[Cybersecurity month](#)  
[Predictions for 2022](#)  
[Cybersecurity for govt](#)  
[Hackathon](#)

## Cookies.

This site uses cookies to give you a better experience. By using our site you agree to the use of cookies. See our [cookie policy](#) for more details.