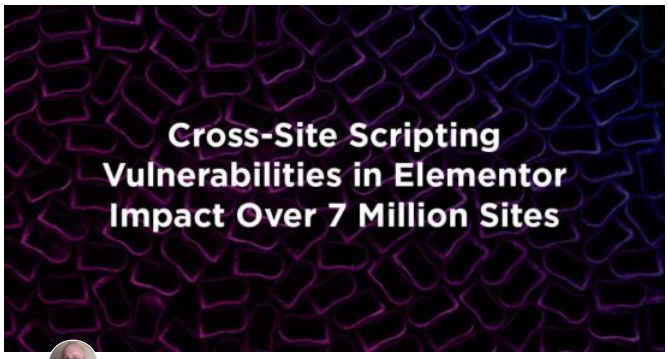


Ram Gall

March 17, 2021

Cross-Site Scripting Vulnerabilities in Elementor Impact Over 7 Million Sites

On February 23, 2021, the Wordfence Threat Intelligence team responsibly disclosed a set of stored Cross-Site Scripting vulnerabilities in Elementor, a WordPress plugin which "is now actively installed and used on more than 7M websites" according to a recent announcement on the Elementor blog. These vulnerabilities allowed any user able to access the Elementor editor, including contributors, to add JavaScript to posts. This JavaScript would be executed if the post was viewed, edited, or previewed by any other site user, and could be used to take over a site if the victim was an administrator.

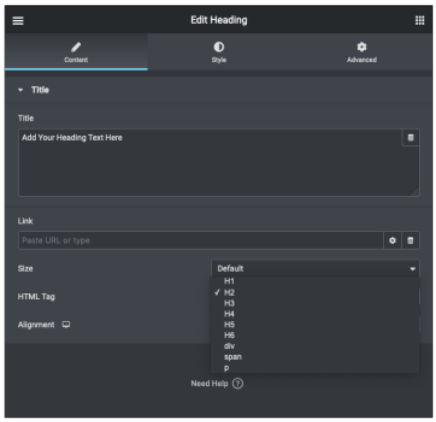
As Elementor has a contact method specifically for security reports, we were able to provide the full disclosure immediately. Elementor acknowledged the vulnerability the next day, on February 24, 2021. An initial patch was made available in version 3.1.2 on March 2, 2021. However, we recommend updating to at least Elementor version 3.1.4, the latest available at the time of this writing, as it contains additional fixes for the issue.

Wordfence Premium users received a firewall rule protecting against these vulnerabilities on February 23, 2021. Sites still running the free version of Wordfence will receive the same protection after 30 days, on March 25, 2021.

Description: Multiple Authenticated Stored Cross-Site Scripting (XSS)
Affected Plugin: Elementor
Plugin Slug: elementor
Affected Versions: < 3.1.2
CVE IDs: Pending
CVSS Score: 6.4 Medium
CVSS Vector: CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:C/C:L/I:L/A:N
Fully Patched Version: 3.1.4

Elementor is a wildly popular editor plugin that allows content creators, including contributors, the ability to visually design websites using "elements" that can be added to any location on the page being built.

Many of these elements offer the option to set an HTML tag for the content within. For example, the "Heading" element can be set to use H1, H2, H3, etc. tags in order to apply different heading sizes via the `header_size` parameter. Unfortunately, for six of these elements, the HTML tags were not validated on the server side, so it was possible for any user able to access the Elementor editor, including contributors, to use this option to add executable JavaScript to a post or page via a crafted request.



Since posts created by contributors are typically reviewed by editors or administrators before publishing, any JavaScript added to one of these posts would be executed in the reviewer's browser. If an administrator reviewed a post containing malicious JavaScript, their authenticated session with high-level privileges could be used to create a new malicious administrator, or to add a backdoor to the site. An attack on this vulnerability could lead to site takeover.

Depending on the element, the executable JavaScript could be added in multiple ways.

For instance, the "Column" element, one of the most basic Elementor components, accepts an `html_tag` parameter. This parameter was output without escaping, and could be set to an inline script, a script with a remote source, or could even be attacked using attribute-based XSS.

The Accordion, Icon Box, and Image Box elements were all vulnerable to this type of attack as well, though the vulnerable parameter names varied depending on the component.

Escaping output isn't always enough

Escaping the output of the chosen HTML tag might have been sufficient to prevent some of these components from being exploitable, and indeed, the "Section" element and the "Toggle" element suffered from similar flaws but could not be exploited because they escaped their chosen HTML tags, and because any additional content was wrapped inside several other levels of tags.

Unfortunately, however, escaping output is not always sufficient to prevent exploits from occurring. For instance, for the "Heading" element, escaping the output of the [illegible] parameter would not have been sufficient to prevent Cross-Site Scripting. To exploit this vulnerability, an attacker would set the `header_size` parameter to `script` and add the actual JavaScript to be executed to the heading text. The "Divider" element was also vulnerable to this type of attack via the `html_tag` parameter because the inner text was nested immediately inside the chosen `html_tag`.

This is an excellent example of why it is important to validate input in addition to escaping output. Enforcing a list of allowed HTML tags on the server side rather than only on the client side would prevent exploitation of this type of vulnerability. Indeed, this is the approach the patched version uses to correct the issue.

Timeline

February 23, 2021 – Wordfence Threat Intelligence releases a firewall rule to Premium users and provides full disclosure to the Elementor security contact.
February 24, 2021 – Elementor acknowledges the disclosure and begins to work on a fix.
March 2, 2021 – An initial patch becomes available in version 3.1.2.
March 8, 2021 – Additional fixes are put in place in version 3.1.4.
March 25, 2021 – The firewall rule becomes available to free users.

Conclusion

In today's article, we detailed stored Cross-Site Scripting(XSS) vulnerabilities present in Elementor, which could be exploited via the Column element as well as the Accordion, Icon Box, Image Box, Heading, and Divider components. These vulnerabilities have been patched in version 3.1.4, and we strongly recommend that all users of Elementor update to the latest version available, which is 3.1.4 at the time of publication.

Wordfence Premium users have been protected against these vulnerabilities since February 23, 2021. Sites still running the free version of Wordfence will receive the same protection 30 days later, on March 25, 2021.

If you know a friend or colleague who is using Elementor, we recommend forwarding this advisory to them, as these vulnerabilities can be used for site takeover. While these vulnerabilities require contributor-level permissions to exploit, the immense popularity of Elementor means that there are likely to be many vulnerable configurations in the wild. As such, we recommend treating these vulnerabilities with greater than normal urgency.
Did you enjoy this post? Share it!

Comments

12 Comments

cedial *
March 17, 2021
11:39 am

Something else to consider is the Elementor interactions with the OceanWP theme, which is used by huge numbers of Elementor users. I switched to Elementor from the WP block editor almost 2 years ago because at that time the block editor had some intolerable bugs. My site is text rather than graphics-based, which means that building blog pages with these inappropriate tools was a nightmare.

I can't prove there is a problem here, but the bugs and erratic behavior of both Elementor and OceanWP make me very nervous. I have programmed all my life, and more stinks here than you have found.

Felix *
March 17, 2021
12:34 pm

Hello,
does this also affect Elementor Pro?

Ram Gall *
March 17, 2021
2:33 pm

Hi Felix,

The vulnerabilities we found were present in the Free version of Elementor, which needs to be installed in order for Elementor Pro to function. That is, if you have Elementor Pro installed, you should still make sure to keep the underlying Elementor installation up to date.

Rubb *
March 17, 2021
11:49 pm

What I need to do if I have the pro version but the it is expired ?
I can't update only the free

Ram Gall *
March 18, 2021
11:29 am

Hi Rubb,

If you can't keep the Pro version of Elementor up to date I'd recommend purchasing a new license or uninstalling it, as you'll want to be protected from any future vulnerabilities discovered in the pro version.

Mushlih Almubarak *
March 18, 2021
1:22 am

Hi
I am using the free version of Elementor.
The question is, whether the free elementor plugin has fixed this vulnerability?
Thank you

Ram Gall *
March 18, 2021
11:27 am

Hi Mushlih,
The free version of elementor has fixed this vulnerability, so be sure to update as soon as you can.

Juice *
March 18, 2021
3:58 pm

Should this be detected as a critical issue in the Wordfence scan? None of my Elementor sites have shown this as an issue in the WF scan despite having the vulnerable Elementor versions.

Ram Gall *
March 19, 2021
8:07 am

Hi,

I recommend creating a support ticket (if you're a Premium user) or a post on our support forums (if you're using the free version of Wordfence). Our support team would be happy to help you out with this!

Max *
March 19, 2021
4:34 am

so, as far as I understand, this exploit can only take place in administration, right? with someone with editing capabilities injecting malicious code etc.

provided it's always better to keep plugins updated, i suppose this means that older versions can still be safely used in scenarios where site owner coincides with administrator or editor (i.e. someone not really that interested in injecting malicious code)?

Ram Gall *
March 19, 2021
6:53 am

Hi Max,

You're correct that this can only be exploited by users that can access the Elementor editor. If the only users on the site are those that are already allowed to add unfiltered HTML or JavaScript, such as administrators or editors, then yes, this doesn't add any additional risk. The primary risk is for sites that have users with fewer privileges, such as contributors and authors, as this creates a larger attack surface.

Max *
March 19, 2021
11:01 am

thanks for confirming (phew!)