CVE-2022-25020 - DOM-based + Stored XSS

February 25, 2022

Chapter 1

Thumbnail Vulnerability

1 Application

https://github.com/pluxml/PluXml

2 Introductory Remarks

Note that an XSS vulnerability was disclosed before in https://www.cvedetails.com/cve/CVE-2021-38602/ and also in https://www.cvedetails.com/cve/CVE-2021-38603/ (regarding the fields headline, content and information). However, these vulnerabilities are fundamentally different from our security issue regarding the thumbnail path in a blog post.

3 Description of the Vulnerability

In the PluXml PHP blogging platform (v5.8.7), unsanitized input from the (optional) thumbnail path of a blog post is stored on the webserver, where it is used to render the corresponding HTML page returned to an arbitrary reader. This results in a stored Cross-Site Scripting attack (stored XSS). Indeed, any blog user can write Javascript code within the thumbnail path of a blog post, which is executed locally and secretly in the blog reader's browser. Additionally, local Javascript code tries to load the thumbnail at the provided path at run time. While doing so, it changes the HTML page in such a way that malicious Javascript code can be injected (DOM-based XSS). As a consequence of both, blog readers or higher privileged users could be the victim of involuntary crypto mining or credential theft. Here is an unlisted (non-public) youtube video that shows the exploit: https://youtu.be/TsGp-QB5XWI.

4 Steps to Reproduce the Exploit

Create a blog post with `" onerror="alert(1)` as thumbnail. This will trigger both: a DOM-based XSS during creation and a stored XSS when publishing or previewing the post.

5 Technical Description of the Vulnerability

In the file `core/admin/article.php`, the `thumbnail` attribute is not sufficiently sanitized before storing the value. Additionally, when displaying the blog post (previewing or publishing) no sanitization is done as well. Moreover, the file `core/admin/article.php` does insert the Javascript code `refreshImg`, which loads the thumbnail at run time and in this process inserts an `<img >` tag in the the innerHTML. The user controlled `dta` attribute is not sufficiently sanitized before changing the HTML code, such that arbitrary Javascript can be inserted.

Chapter 2

Thumbnail Alternative Text Vulnerability

6 Description of the Vulnerability

In the PluXml PHP blogging platform (v5.8.7), unsanitized input from the (optional) alternative text of a thumbnail in a blog post is stored on the webserver, where it is used to render the corresponding HTML page returned to an arbitrary reader. This results in a stored Cross-Site Scripting attack (stored XSS). Indeed, any blog user can write Javascript code within the alternative text of the thumbnail of a blog post, which is executed locally and secretly in the blog reader's browser. As a consequence, blog readers could be the victim of involuntary crypto mining or credential theft. Here is an unlisted (non-public) youtube video that shows the exploit: `https://youtu.be/TNNzl4g2ufE`.

7 Steps to Reproduce the Exploit

Create a blog post with `"><iframe src="javascript:alert(1)">` as alternative text of the image and non-empty thumbnail path.

8 Technical Description of the Vulnerability

In the file `core/admin/article.php`, the `thumbnail_alt` attribute is not sufficiently sanitized before storing the value. Additionally, when displaying the blog post (previewing or publishing) only insufficient sanitization is done as well.