Portswigger: DOM XSS attack using reflected input

Sam Newman

April 2025

1 Website in Question

This section of the website contains a vulnerability with the search function. The function allows users to search for items on the webpage and matches you with searches matching your query

2 Business Impact

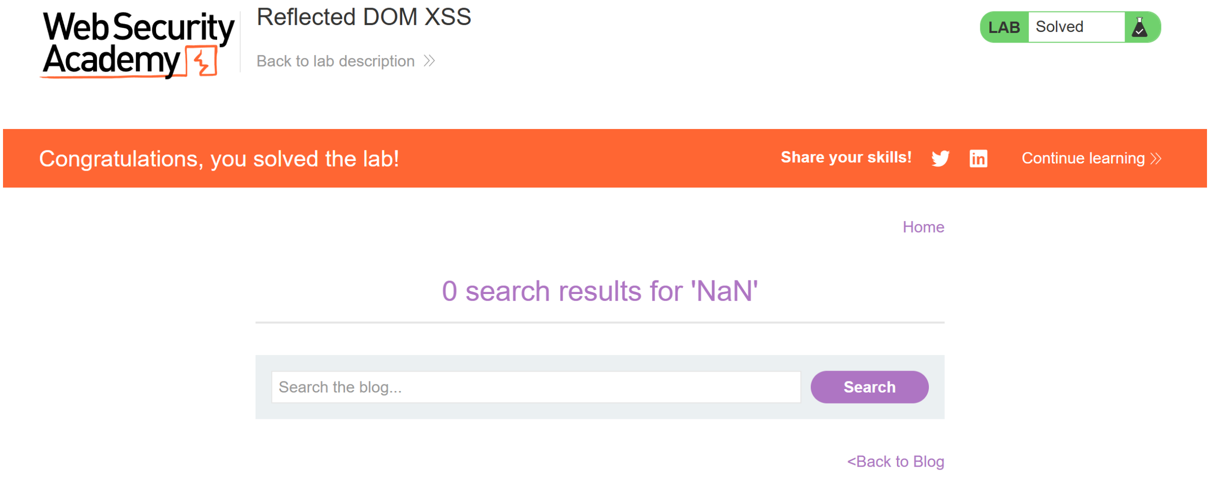
This vulnerability could allow arbitrary JS code execution on a victim machine, simply by clicking on a link which could be supplied from a social media page, text message, or email. Arbitrary code execution on a browser is a terrible scenario, and could be used to dump cookies, passwords, and user data, and even change the HTML of a site and redirect a user to a malicious page.

3 How It Works

First, I explored the website using Portswigger, checking out what different buttons did and seeing where their requests went. By looking at website pages and responses, and exploring what the site did with queries into the search bar, I could see that the website took the user query and injected it into a JSON object, which was then used in conjunction with the unsafe eval() function to execute the search for the item. By using escape characters for the quotation marks and a - instead of the encoded +, we could escape the quotes and attack an alert() to the end of the JSON string, then comment out the rest to make the object valid. The input I used was \ ”-alert(1)// in the search parameter of the URL. This escapes the string in the JSOn object, injecting the malicious payload into the site

4 Why It Works + Mitigation

The exploit works via use of the unsafe eval() function. This function executes string input as code, and should never be used in a website interface precisely for its vulnerability to XSS attack.



Reflected DOM XSS

Back to lab description

LAB Solved

Congratulations, you solved the lab!

Share your skills!

Continue learning

Home

0 search results for 'NaN'

Search the blog...

Search

<Back to Blog

Figure 1: Completed Lab