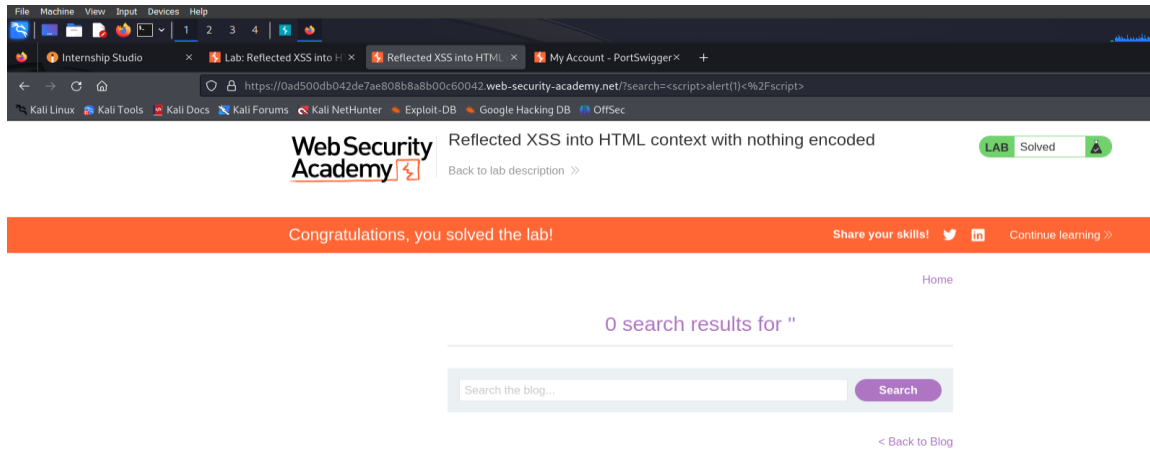
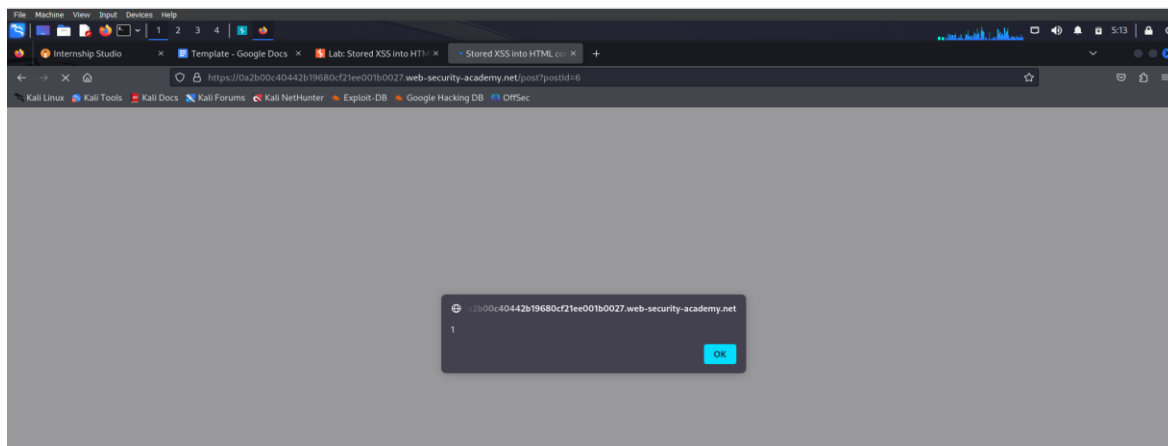


## TASK 1

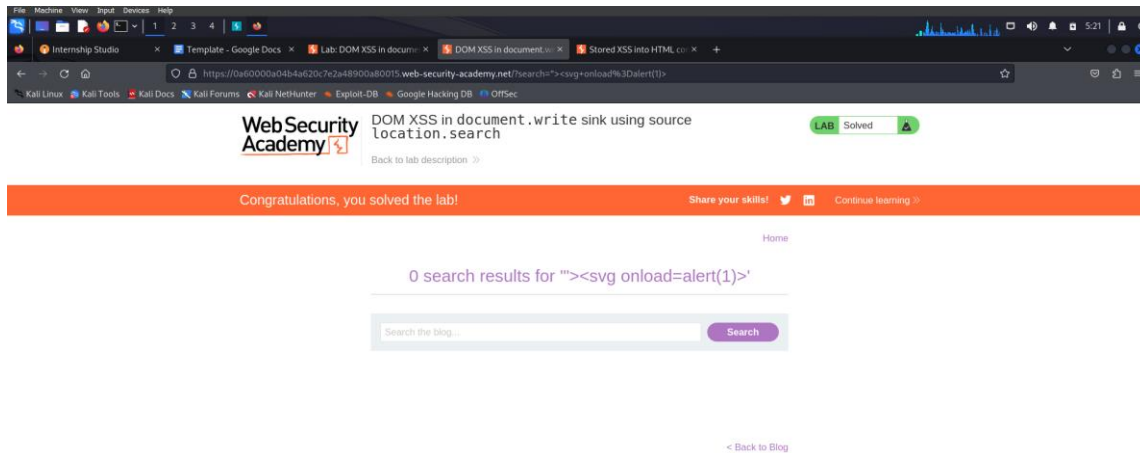
### LAB 1: Reflected XSS into HTML context with nothing encoded



### LAB 2: Stored XSS into HTML context with nothing encoded

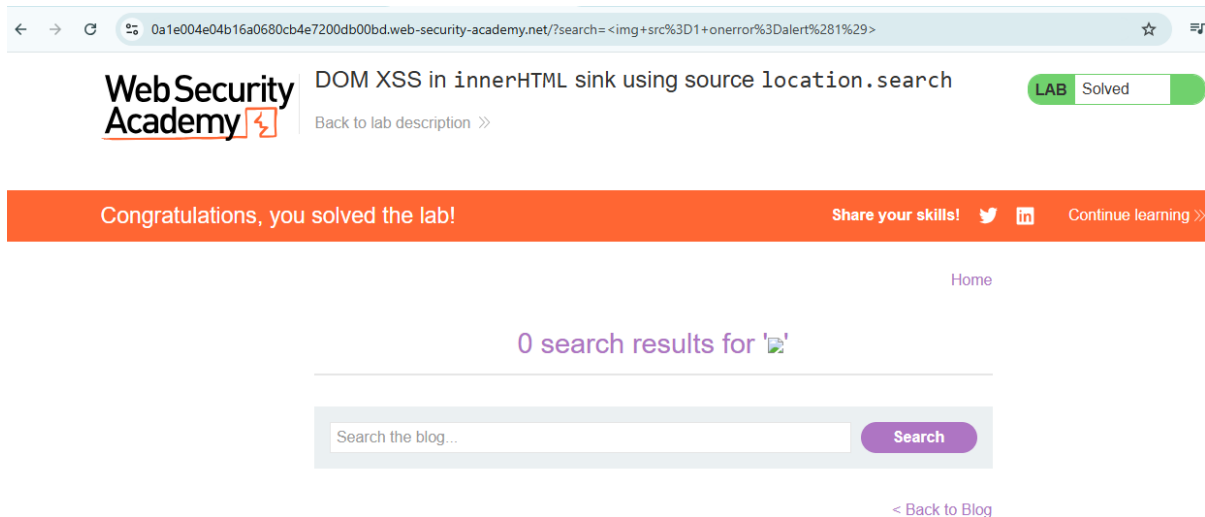


### LAB 3: DOM XSS in `document.write` sink using `source.location.search`



The screenshot shows a web browser window with multiple tabs. The active tab is titled "DOM XSS in document.write sink using source.location.search". The address bar shows the URL: `https://0a60000a04b4a629c7c2a48900a80015.web-security-academy.net/?search=<svg+onload=alert(1)>`. The page header includes the WebSecurity Academy logo and a "LAB Solved" badge. Below the header, an orange banner reads "Congratulations, you solved the lab!". The main content area shows "0 search results for '<svg onload=alert(1)>'" and a search bar with the placeholder "Search the blog...". A link to "Back to lab description" is visible. At the bottom, there is a "Home" link and a "< Back to Blog" link.

### LAB 4: DOM XSS in `innerHTML` sink using `source.location.search`



The screenshot shows a web browser window with a single tab titled "DOM XSS in innerHTML sink using source.location.search". The address bar shows the URL: `0a1e004e04b16a0680cb4e7200db00bd.web-security-academy.net/?search=<img+src%3D1+onerror%3Dalert%281%29>`. The page header includes the WebSecurity Academy logo and a "LAB Solved" badge. Below the header, an orange banner reads "Congratulations, you solved the lab!". The main content area shows "0 search results for '<img src=1+onerror=alert(1)>'" and a search bar with the placeholder "Search the blog...". A link to "Back to lab description" is visible. At the bottom, there is a "Home" link and a "< Back to Blog" link.

### LAB 5 : DOM XSS using web messages

WebSecurity Academy

DOM XSS using web messages

LAB Solved

Back to lab description

Congratulations, you solved the lab!

Share your skills! [Twitter](#) [LinkedIn](#) [Continue learning](#)

This is your server. You can use the form below to save an exploit, and send it to the victim.  
Please note that the victim uses Google Chrome. When you test your exploit against yourself, we recommend using Burp's Browser or Chrome.

Craft a response

URL: `https://exploit-0a3c00c10483e2c2ac9a60a601e900ec.exploit-server.net/exploit`

HTTPS



File:

`/exploit`

Head:

HTTP/1.1 200 OK  
Content-Type: text/html; charset=utf-8