User Credentials

Name: Sean Kells

Email: chipmunkd3@gmail.com

Password: Spaghettios4!

Reflected XSS

Unlike the Persistent XSS flaw that I found on the website, this flaw can only be found on the updated version of armbook and not the with the source code provided to us. Although, judging from the HTML of search.php I can make a safe assumption where and how this vulnerability came from. The reflected XSS stems from search.php lines 129 – 131 in our provided source code where instead of merely saying "Sorry, there were no results" the updated armbook states "Sorry there were no results for <query parameter>". This query parameter being unsanitized results in the vulnerability. Since it isn't stored in the database nor are we modifying the DOM we know for certain that this is Reflected XSS. The javascript I used to determine the reflected XSS vulnerability is:

<script>alert("REFLECTED XSS")</script>

Proof: Reflected.png

Persistent XSS

Inside of home.php, specifically the posts.text of database displayed by timeline.php which is then displayed by home.php. This XSS vulnerability is due to the add_comment.php which adds unsanitized comments with a maximum size of 300. The adversary can add a malicious comment to a homepage of another user with javascript that will then be displayed on the page. We can see the comment being uploaded to the database without any effective sanitation in add_comment.php lines 31 to 39. The javascript I used to determine the persistence XSS vulnerability is:

<script>alert("PERSISTENT XSS");</script>

Proof: Persistent.png