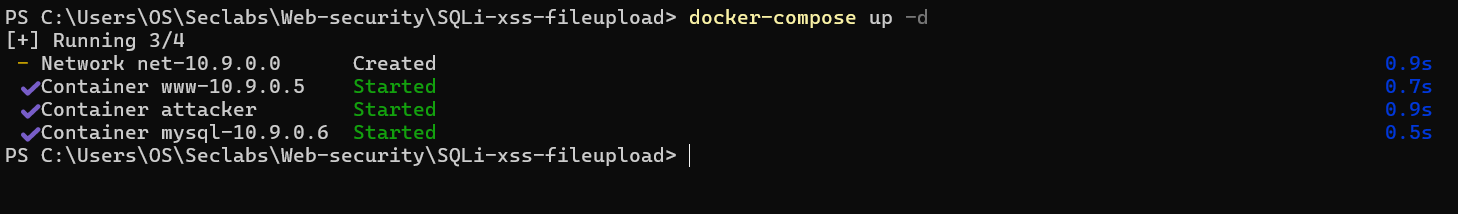
**Web security – XSS – File Upload Vulnerabilities**

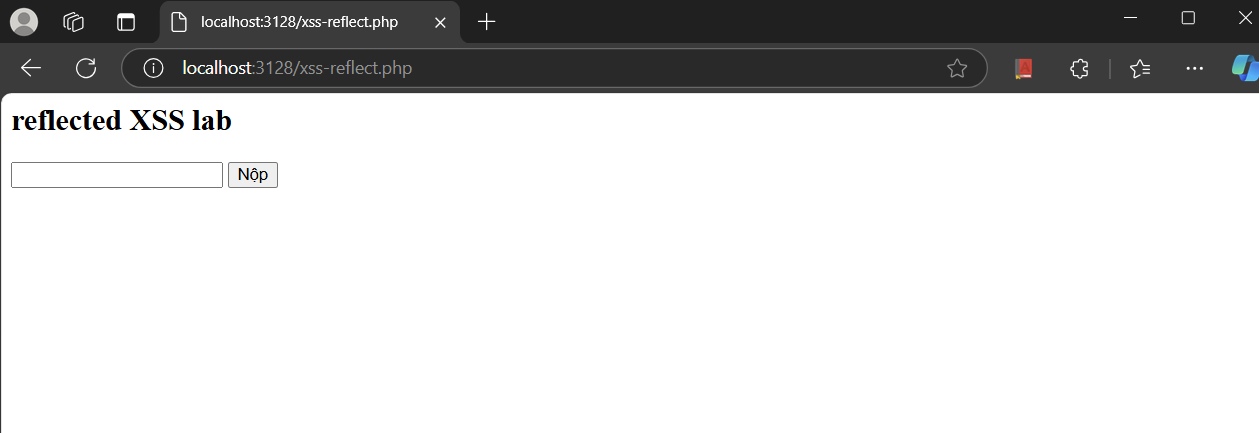
**Tasks**

**1. Setup**



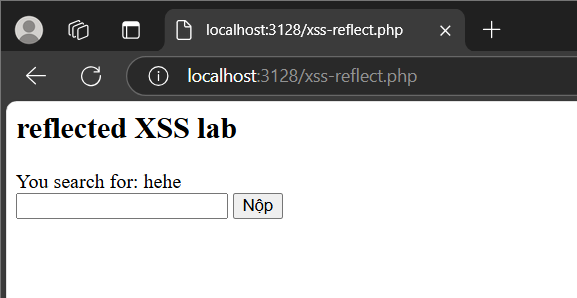
**2. Reflected-xss Attack**

Access page at <http://localhost:3128/xss-reflect.php>.

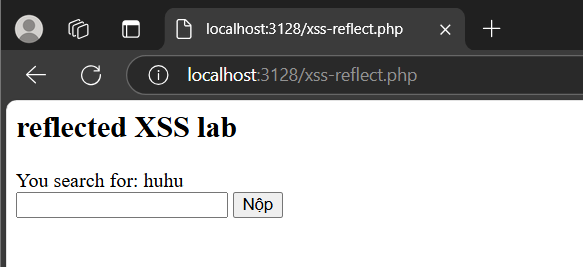


**a) Try various plain text input then click Submit to see the result**

When I search for “hehe”



When I search for huhu

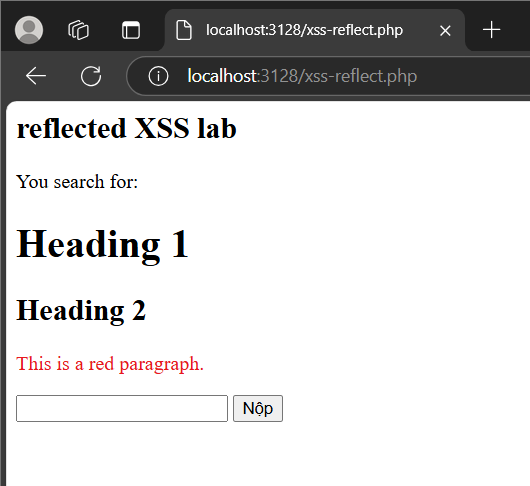


**b) Try various text with html tag input then click Submit to see the result.**

I try with this input text:

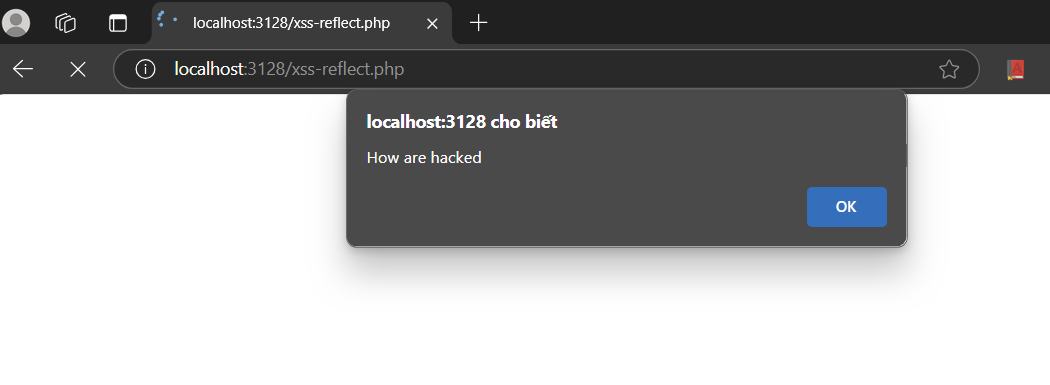
**<h1>Heading 1</h1><h2>Heading 2</h2><p style="color:red;">This is a red paragraph.</p>**

The result



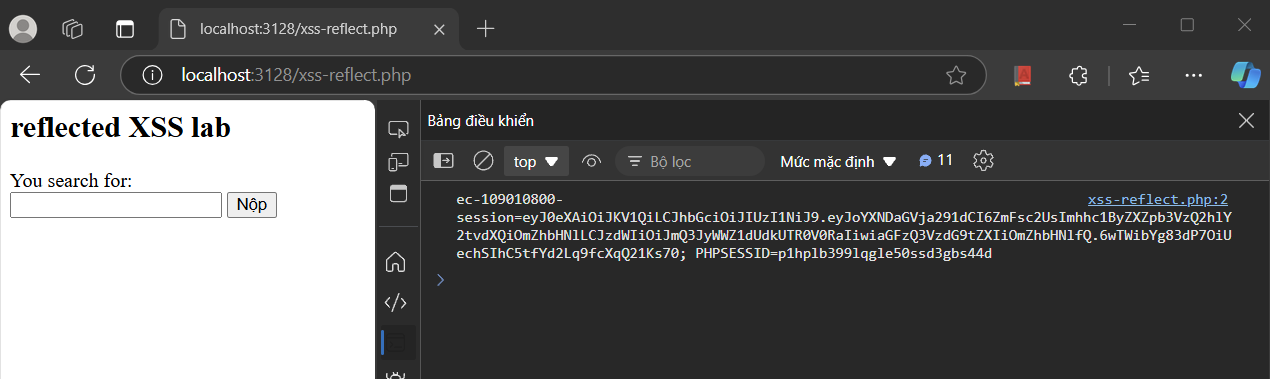
**2.1 Attack**

**c) Try to input this to see the result: <script>alert("How are hacked")</script>**

****

**d) Enter JavaScript code to print out cookies from the web page.**

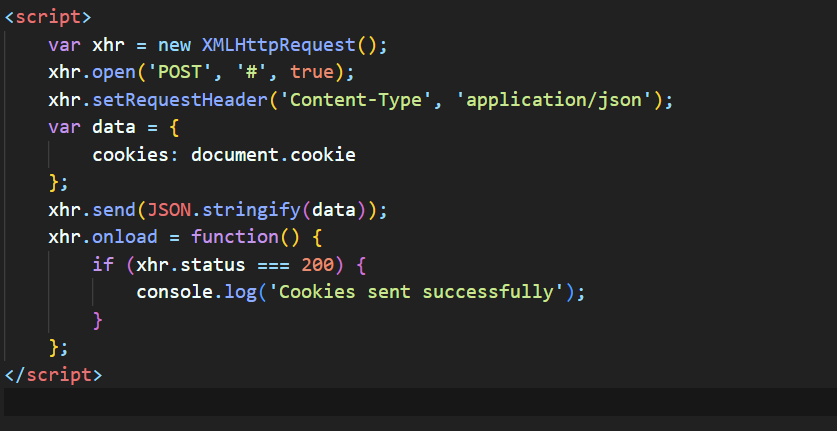
I can use <script>console.log(document.cookie)</script> to print out cookies from the web page



**e) Suppose you want to steal cookies from this page and send it to a remote computer. Find the**

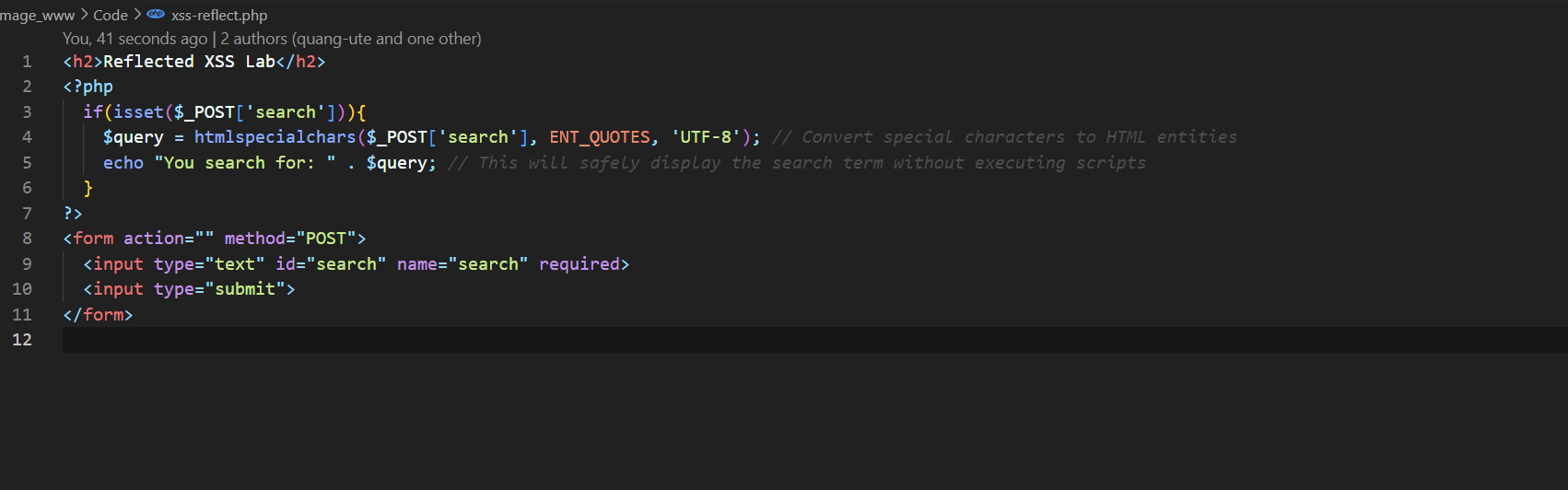
**way to implement this.**

So if I want to steel cookies from this pages and send it to a remote computer. I can construct a form and then send the data to that remote computer.

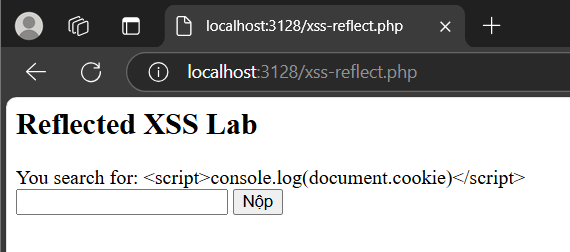


**2.2 Countermeasures**

So I can use the medthod in php, that is htmlspecialchars => will convert html special characters into HTML entities prevents website from being attacked



And the result

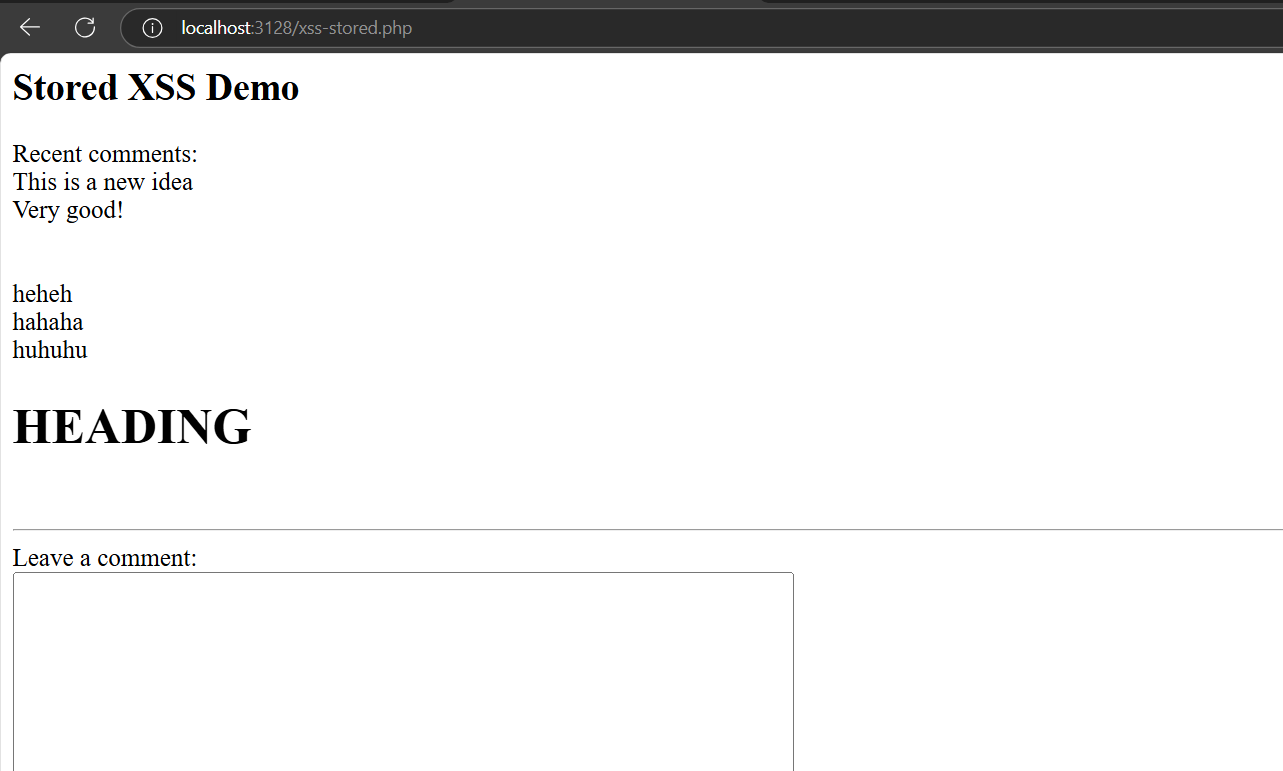


**3. Stored-XSS Attack**

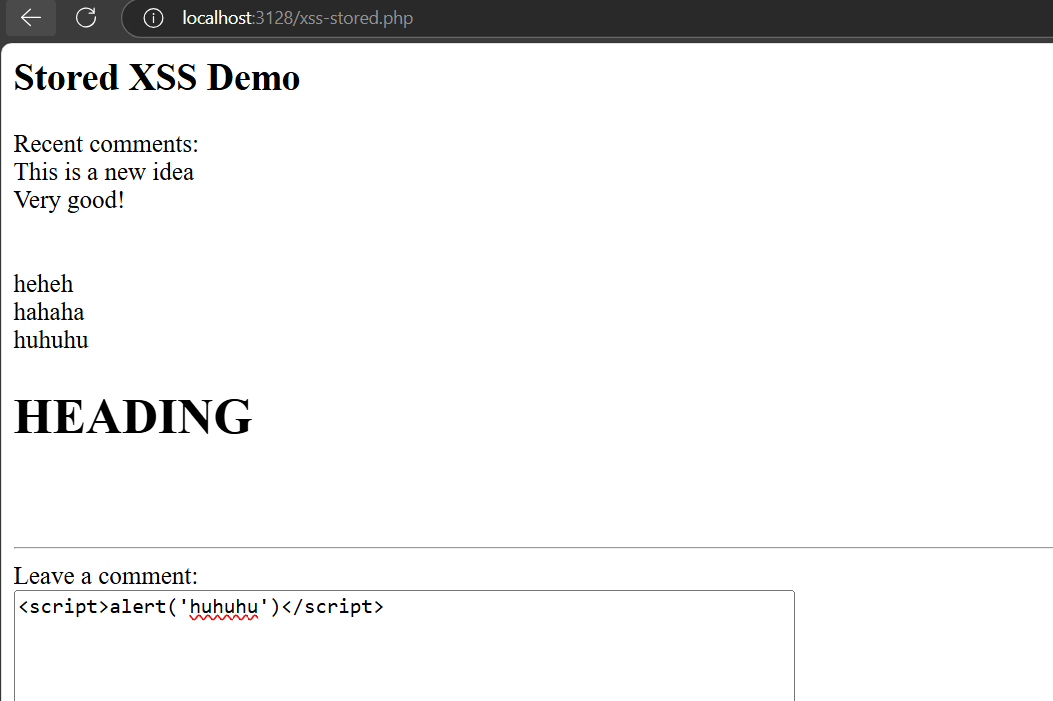
**Access page at <http://localhost:3128/xss-stored.php>.**

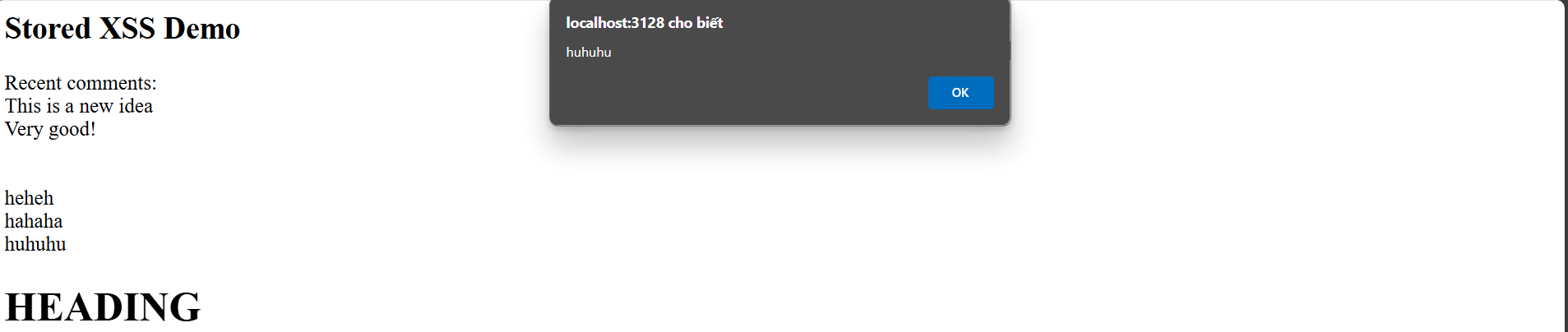
**3.1 Attack**

**a) Try various plain text or text inside html tag comments then click Submit to see the result.**



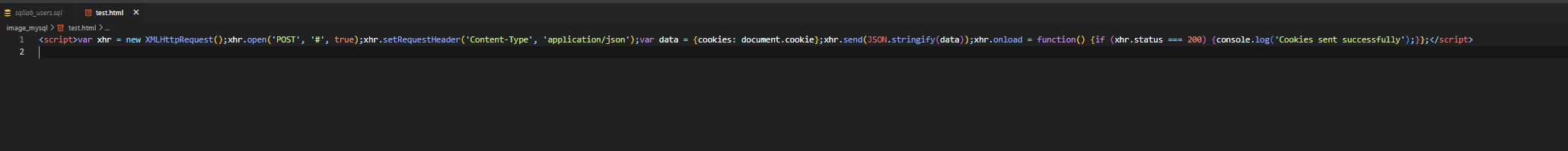
**b) Try to input a JavaScript code in the comment box, the code will be stored and executed each time submit button is hit.**

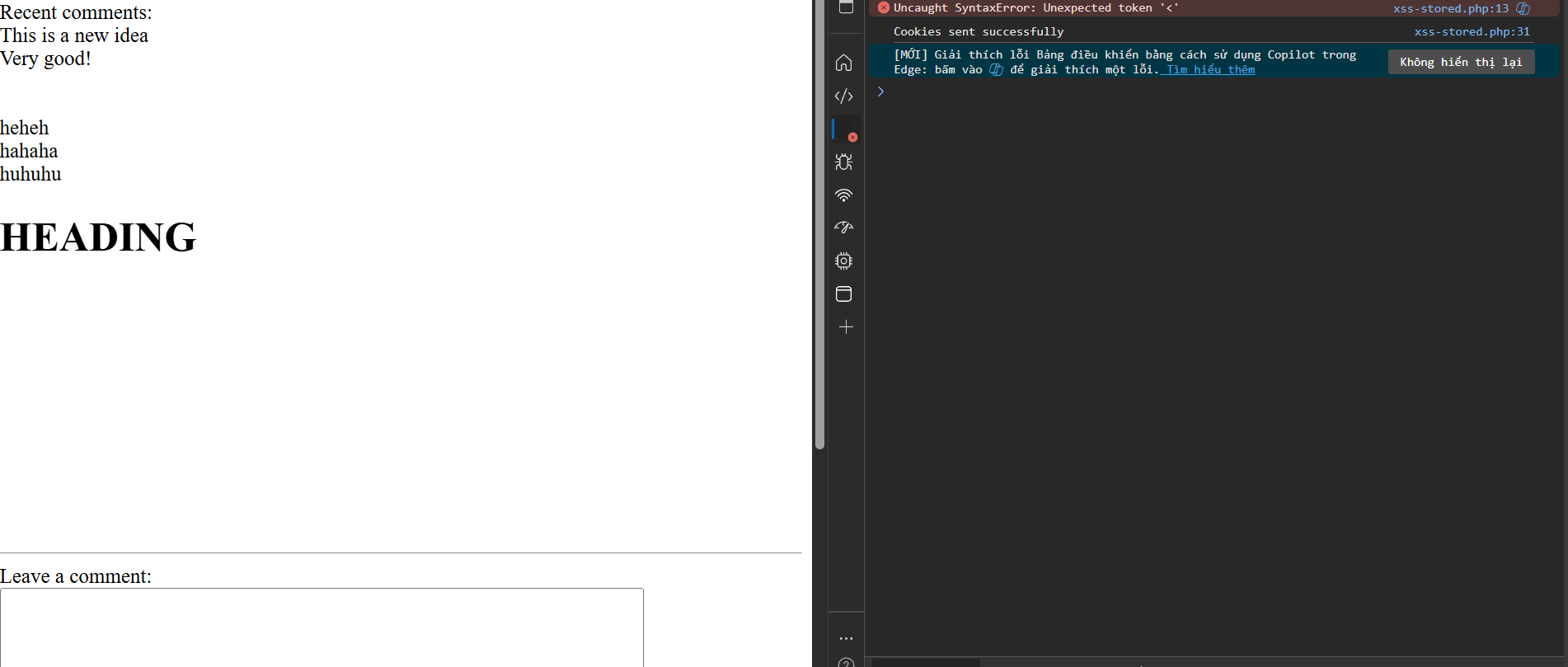




**c) Implement the way you steal cookie in 4.2.1 (e) for this task**

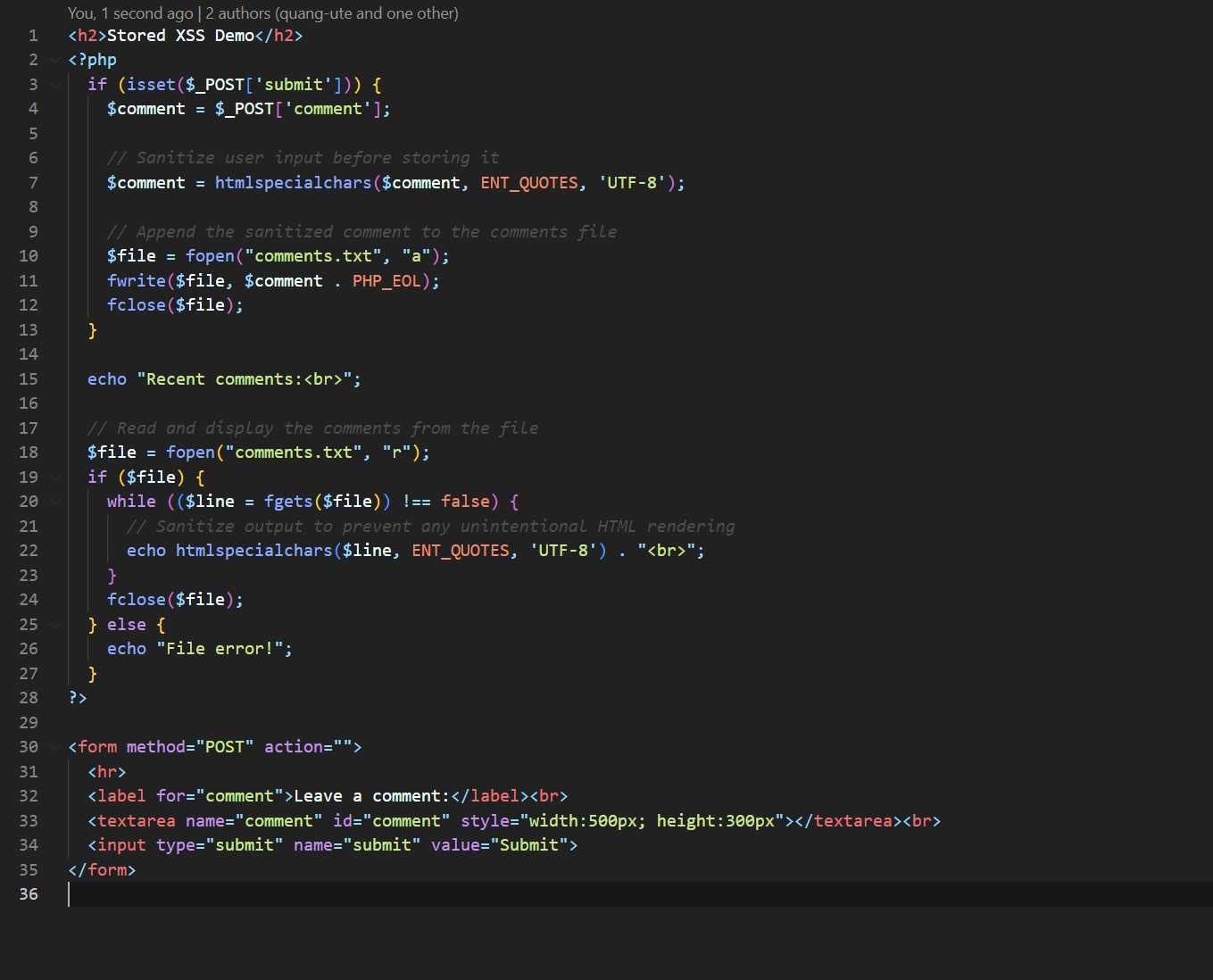
Because the comment display section has a br element every line so that it will reduce it to 1 line

Then as you can see, we have sent coookie succesfully

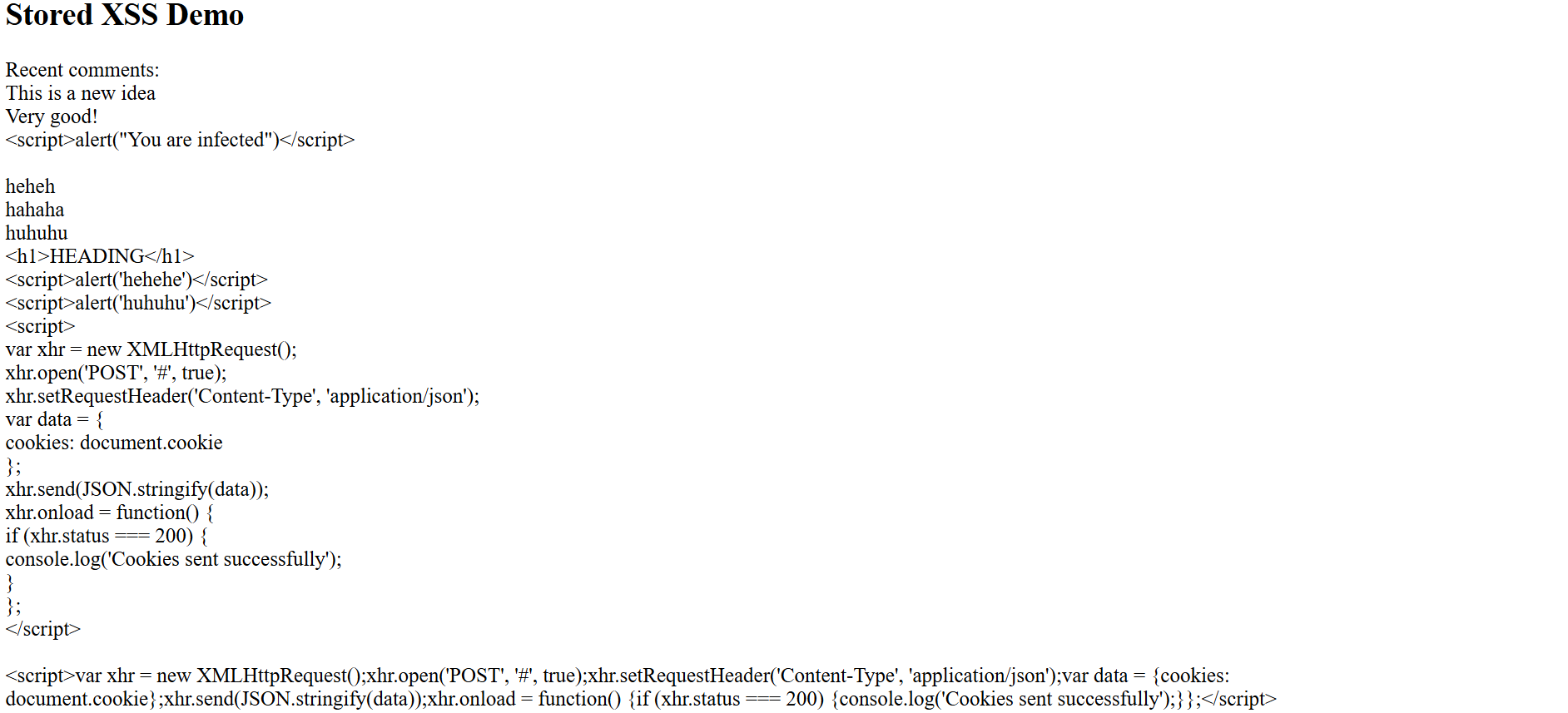


**3.2 Countermeasures**

We can use the htmlSpecialChar to preven all these guys!

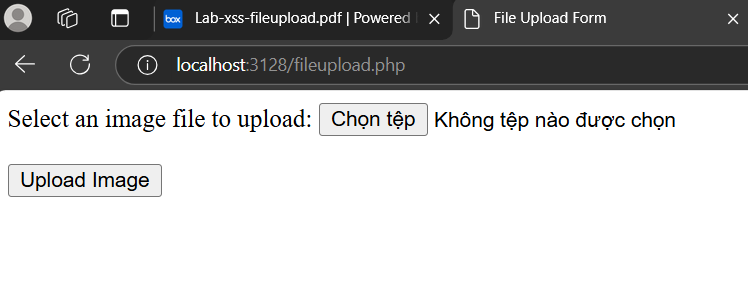


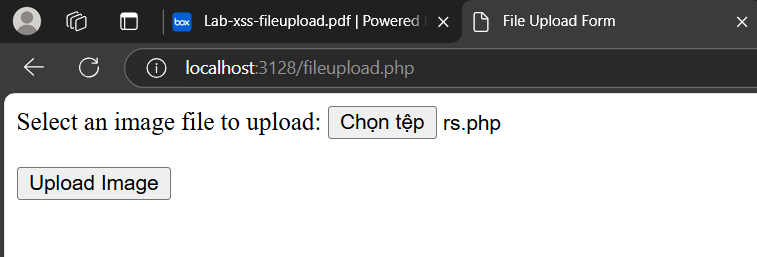
When we edit again!

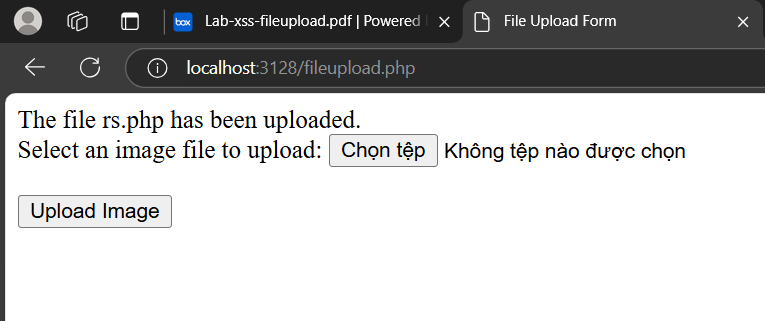


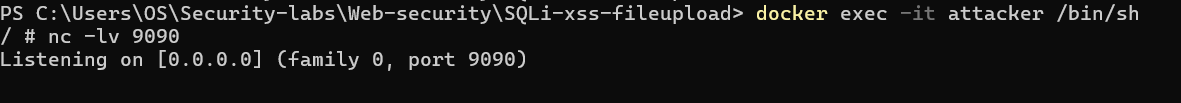
**4 Fileupload Attack**

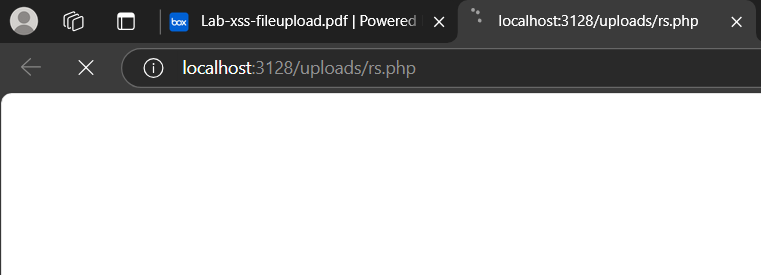
Access the file upload page at <http://localhost:3128/fileupload.php>

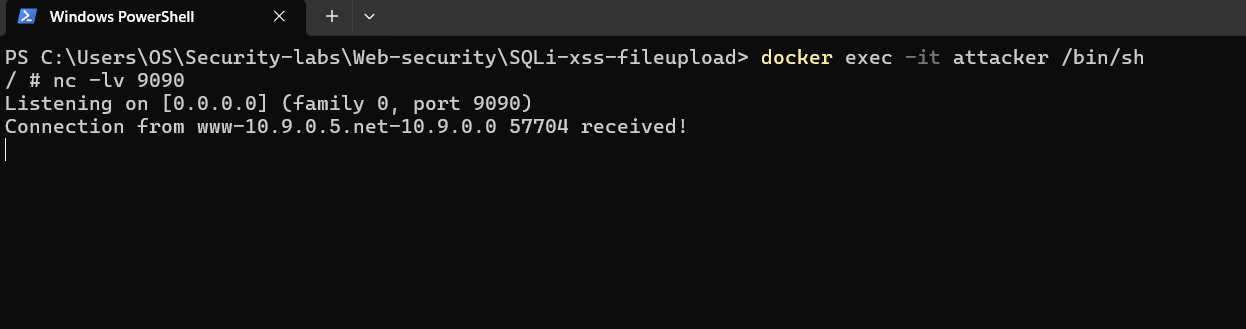












1. **Countermeasures**

