14.08.2023

Курс:

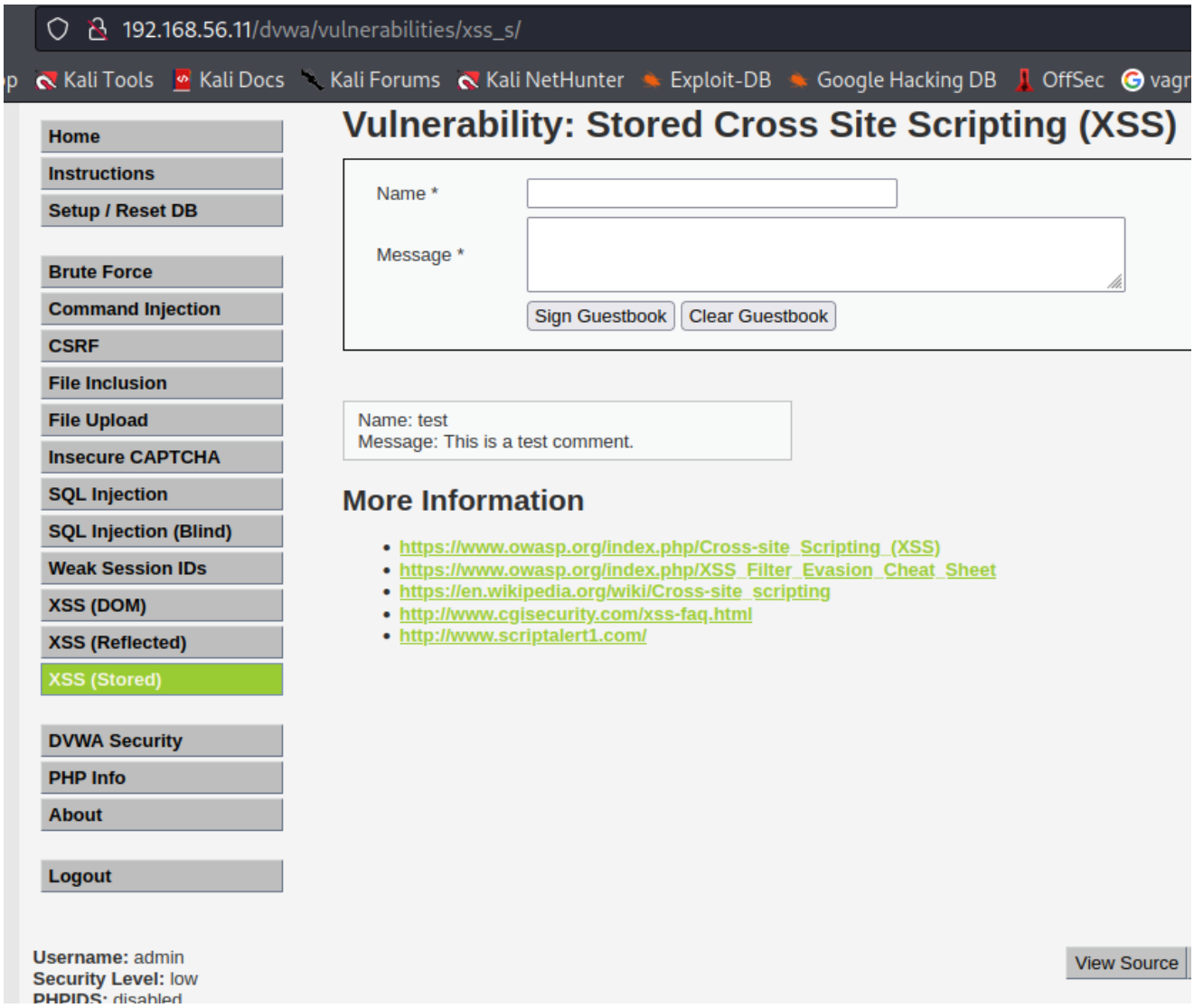
Практическая работа к уроку № Lesson_2

--

Задание_1:

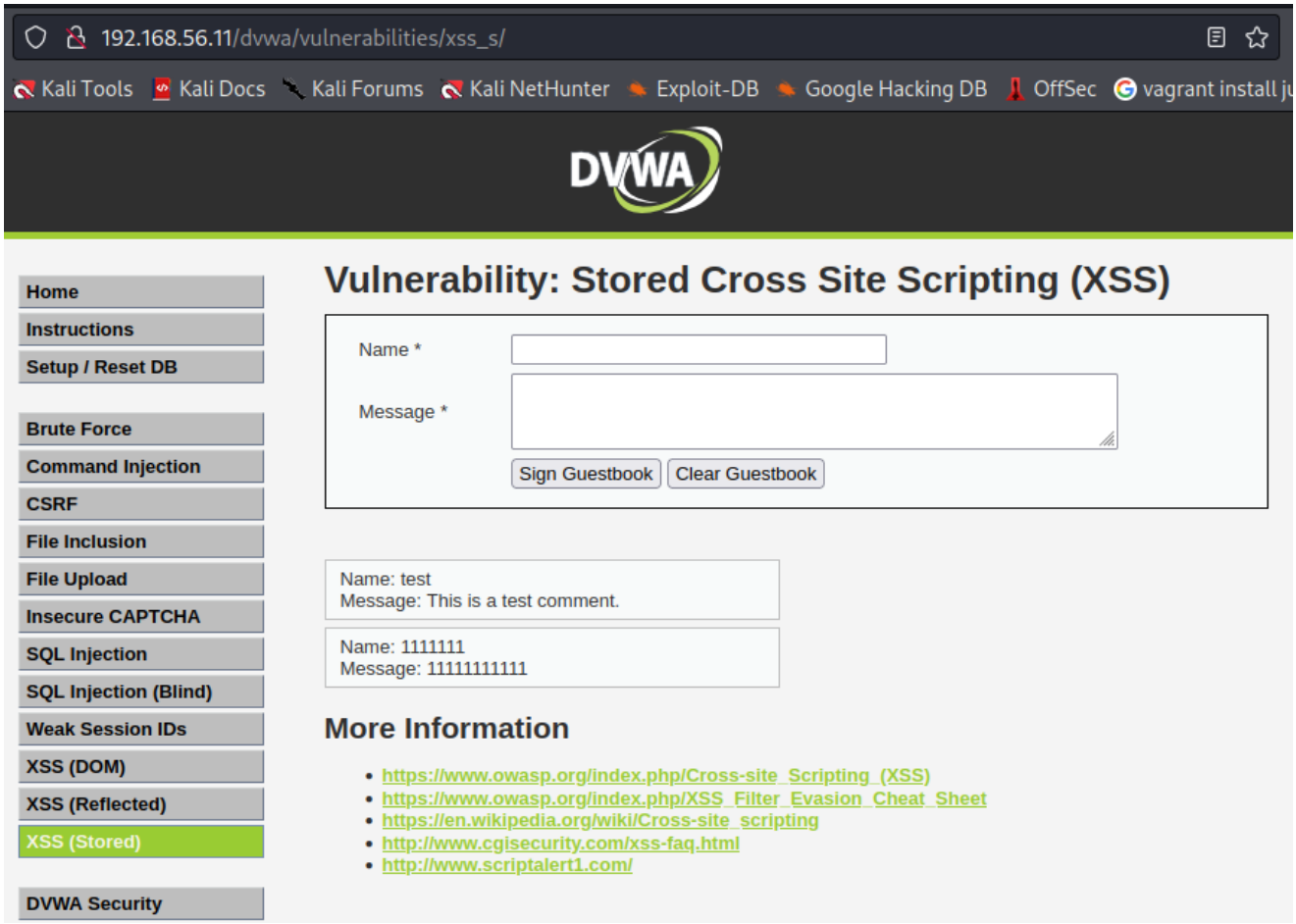
Найдите XSS на странице XSS(Stored) проекта DVWA на простом (Low) уровне сложности и определите ее тип. Составьте отчет о найденной уязвимости. В отчете укажите, в каком контексте присутствует XSS.

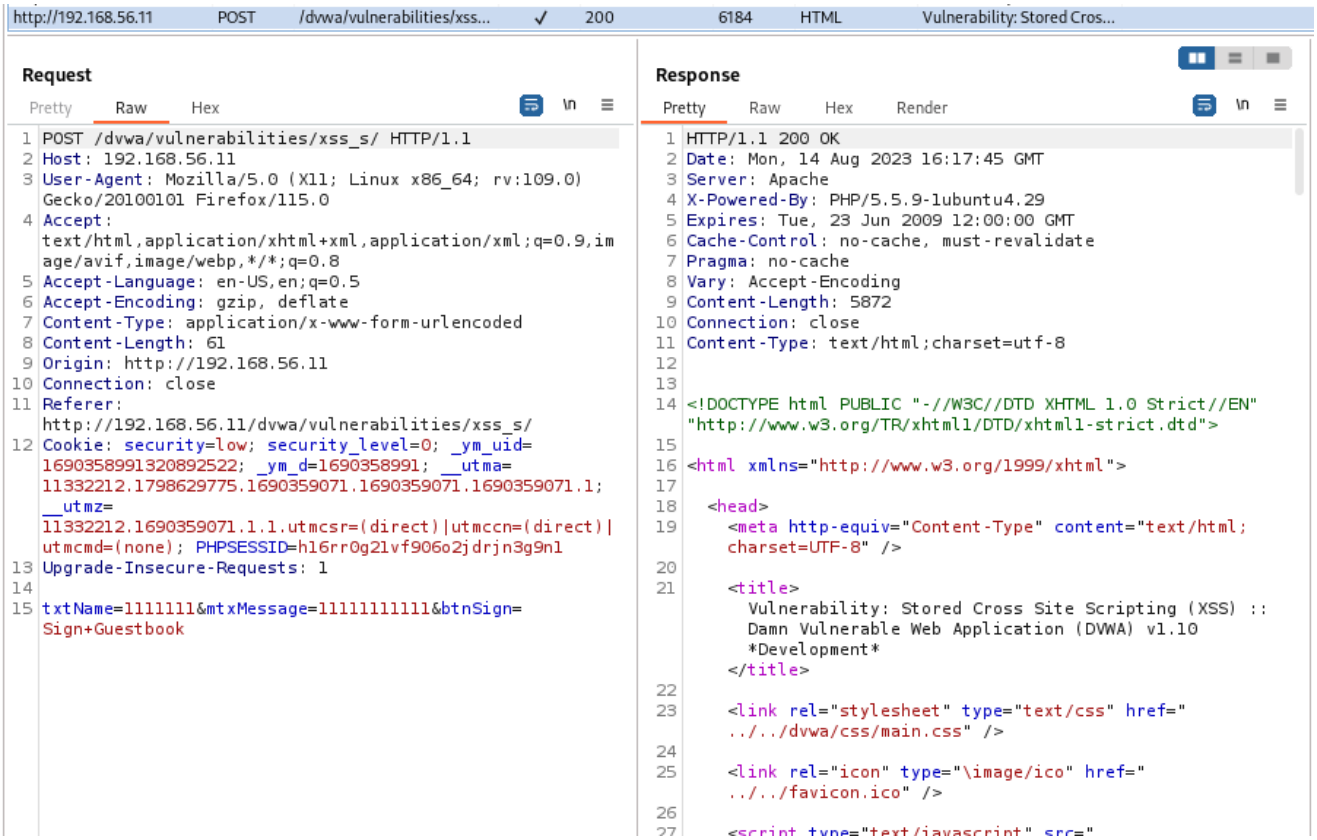
- http://192.168.56.11/dvwa/vulnerabilities/xss_s/




```
# Stored XSS Source
<?php      if( isset( $_POST[ 'btnSign' ] ) ) {
// Get input
```

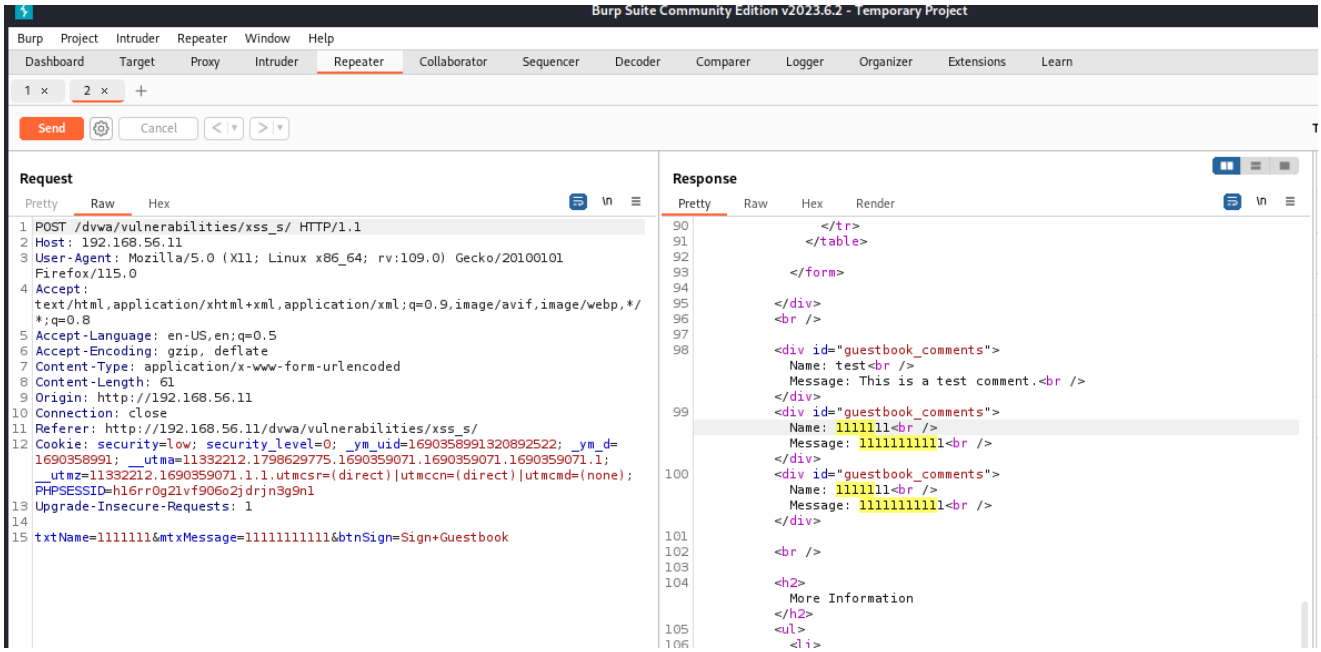
```php
$message = trim( $_POST[ 'mtxMessage' ] );
$name    = trim( $_POST[ 'txtName' ] );
// Sanitize message input
$message = stripslashes( $message );
$message = ((isset($GLOBALS["___mysqli_ston"]) &&
is_object($GLOBALS["___mysqli_ston"])) ?
mysqli_real_escape_string($GLOBALS["___mysqli_ston"],  $message ) :
((trigger_error("[MySQLConverterToo] Fix the mysql_escape_string() call! This code
does not work.", E_USER_ERROR)) ? "" : ""));
// Sanitize name input
$name = ((isset($GLOBALS["___mysqli_ston"]) &&
is_object($GLOBALS["___mysqli_ston"])) ?
mysqli_real_escape_string($GLOBALS["___mysqli_ston"],  $name ) : ((trigger_error("
[MySQLConverterToo] Fix the mysql_escape_string() call! This code does not work.",
E_USER_ERROR)) ? "" : ""));
// Update database
$query  = "INSERT INTO guestbook ( comment, name ) VALUES ( '$message', '$name' );";
$result = mysqli_query($GLOBALS["___mysqli_ston"],  $query ) or die( '<pre>' .
((is_object($GLOBALS["___mysqli_ston"])) ? mysqli_error($GLOBALS["___mysqli_ston"])
: (($___mysqli_res = mysqli_connect_error()) ? $___mysqli_res : false)) . '</pre>'
);
//mysql_close();   }
?>
```



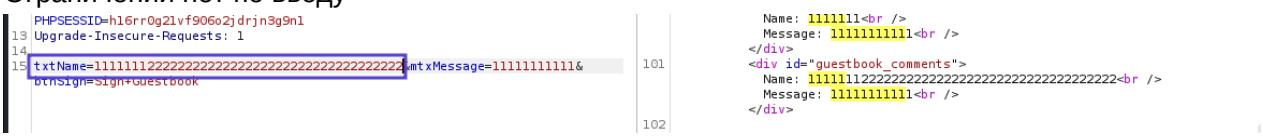


- Send to Repeater

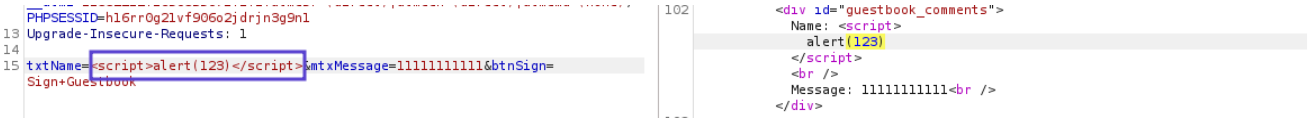
Видим, что УЯ XSS присутствует в обоих передаваемых параметрах



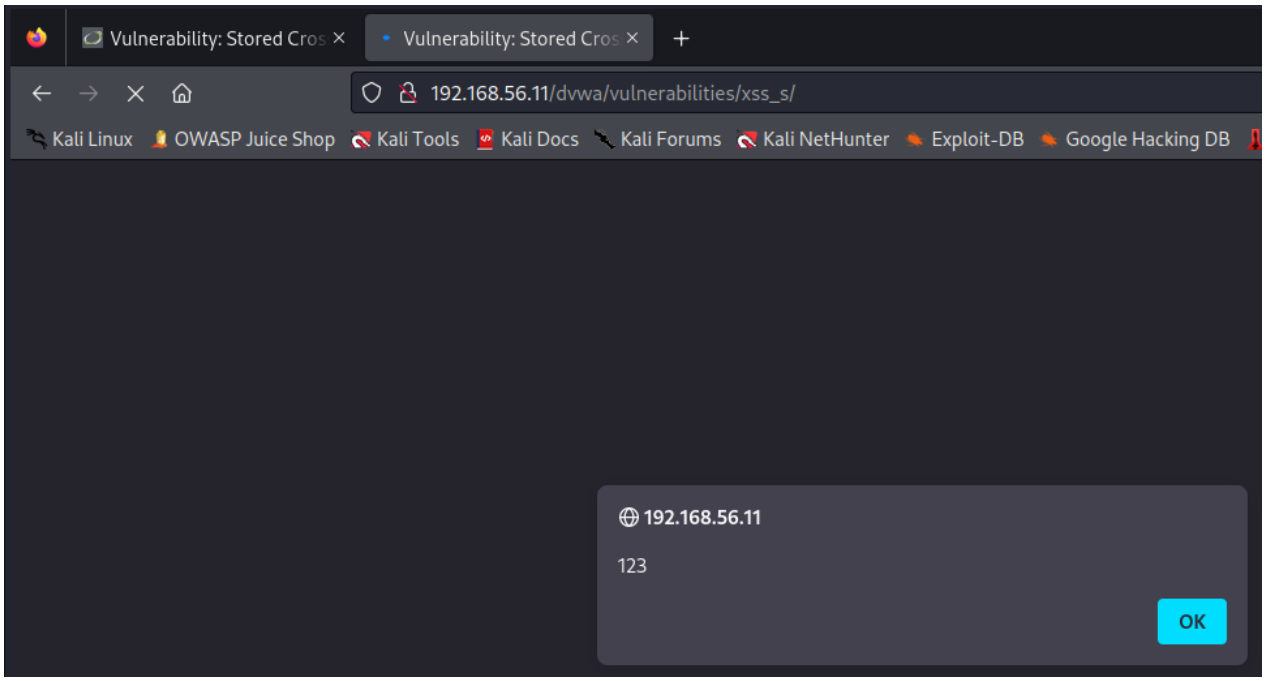
- Ограничений нет по вводу

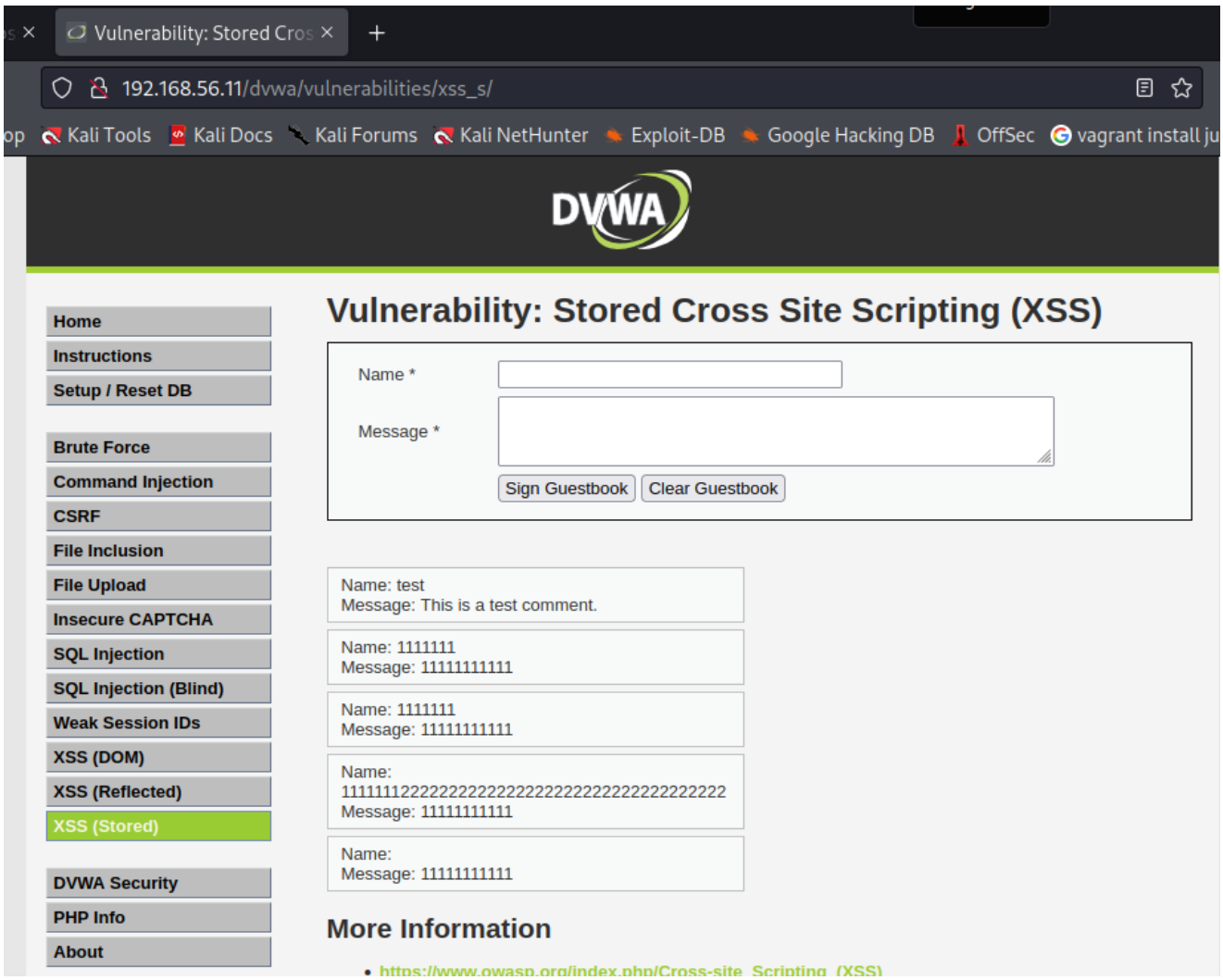


- Вектор атаки:
  - `<script>alert(123)</script>`



- Отправляем ссылку в браузер из репитера





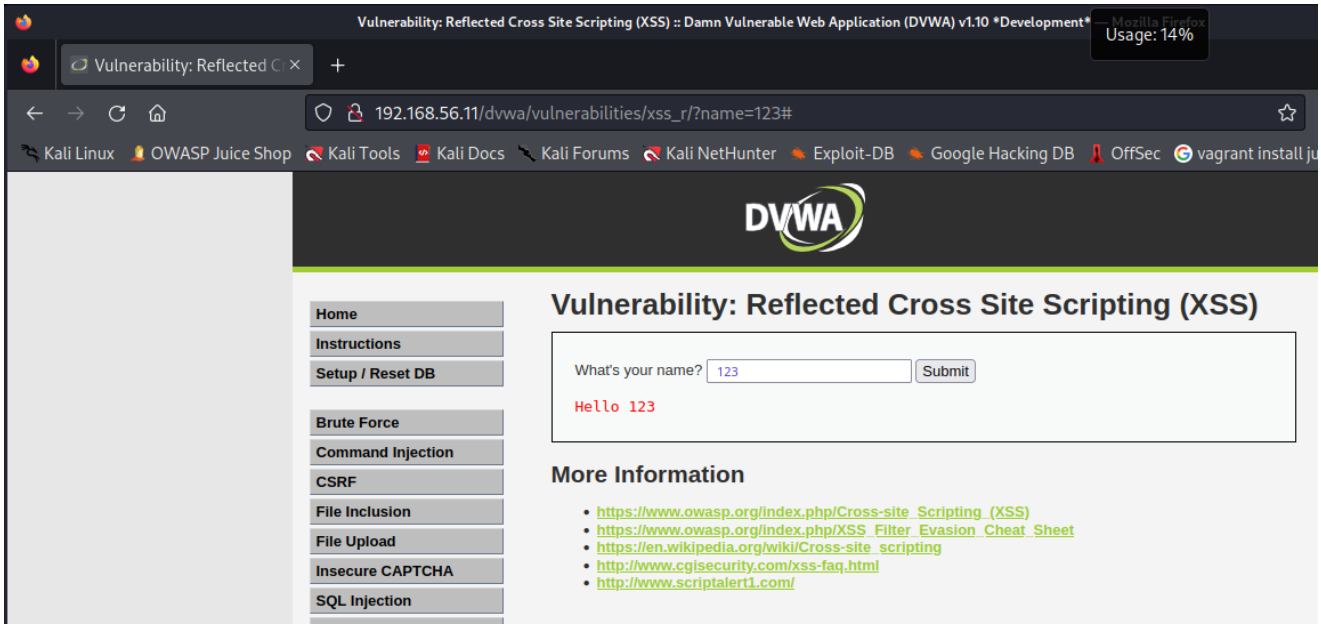
Срабатывает. Вектор на странице не остался, из этого следует, что это отраженный XSS.

Задание_2:

Найдите XSS на странице XSS(Reflected) проекта DVWA на простом (Low) уровне сложности и определите ее тип. Составьте отчет о найденной уязвимости. В отчете укажите, в каком контексте присутствует XSS.

- http://192.168.56.11/dvwa/vulnerabilities/xss_r/
- Burp Suite

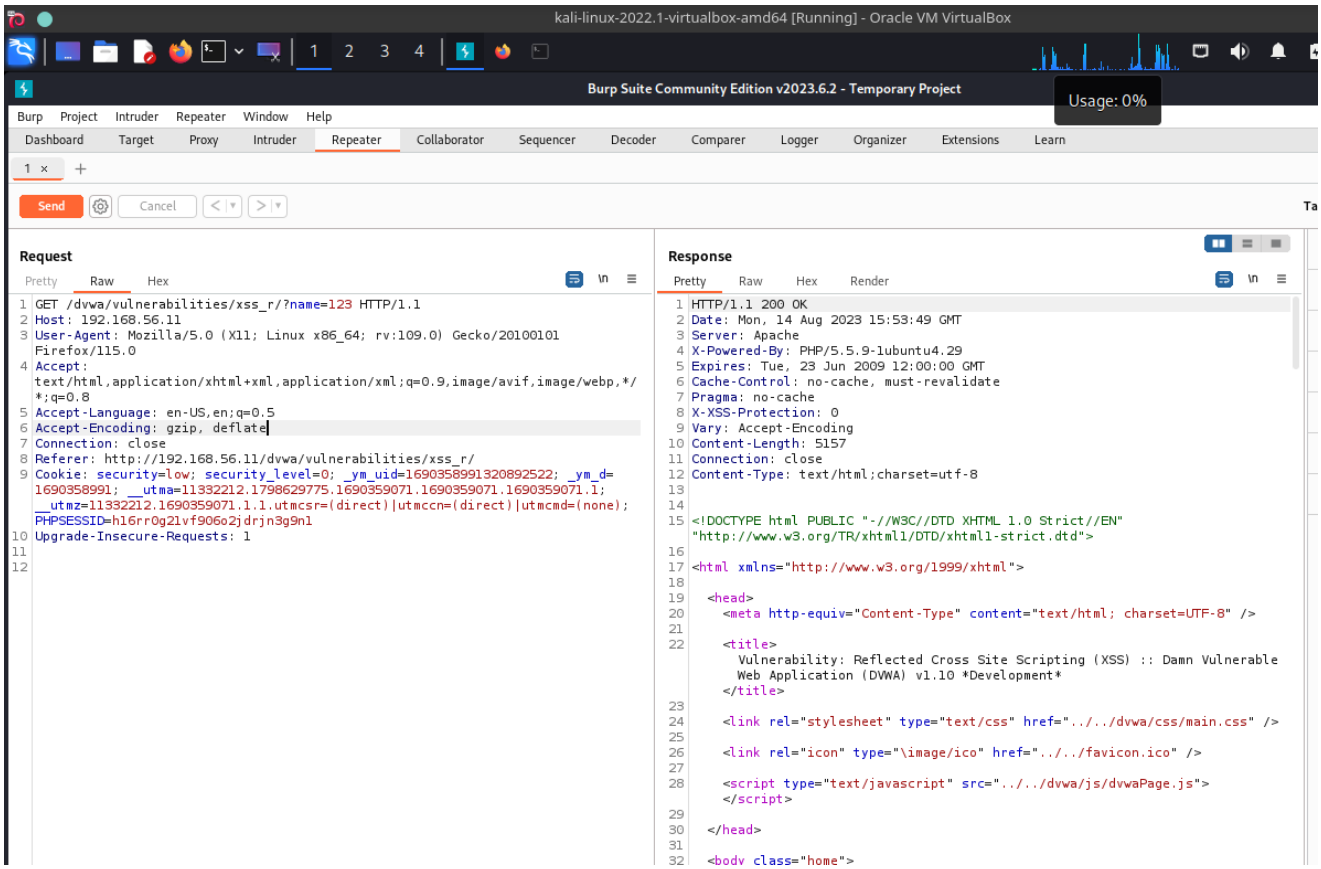
```php
 <?php
header ("X-XSS-Protection: 0");
// Is there any input?
if( array_key_exists( "name", $_GET ) && $_GET[ 'name' ] != NULL ) {
    // Feedback for end user
    echo '<pre>Hello ' . $_GET[ 'name' ] . '</pre>';
}
?>
```



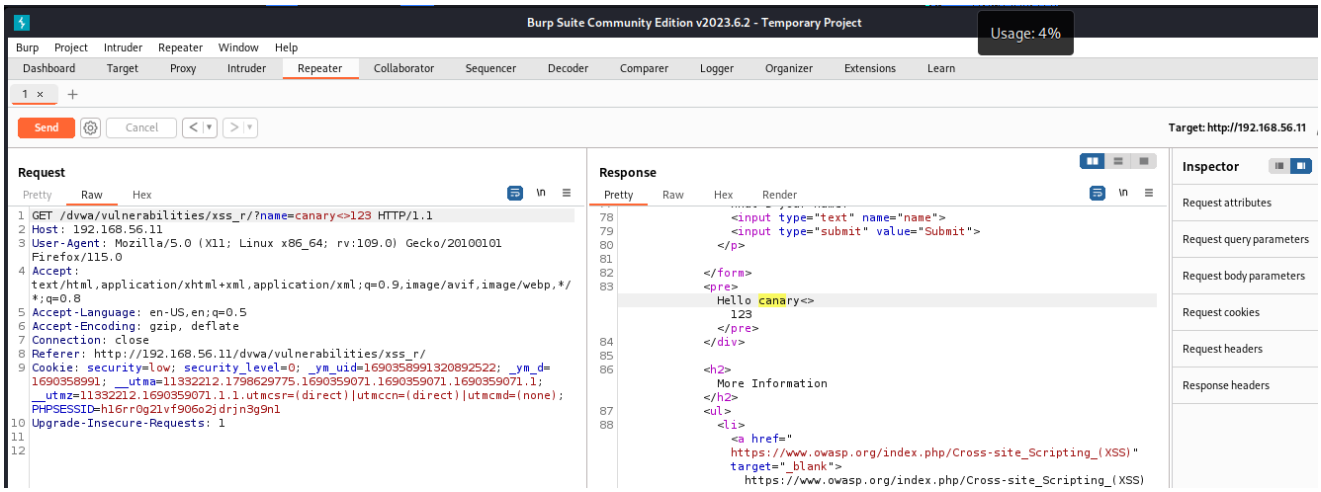
- Burp Suite (FoxitProxy: ON ; Intruder: OFF)

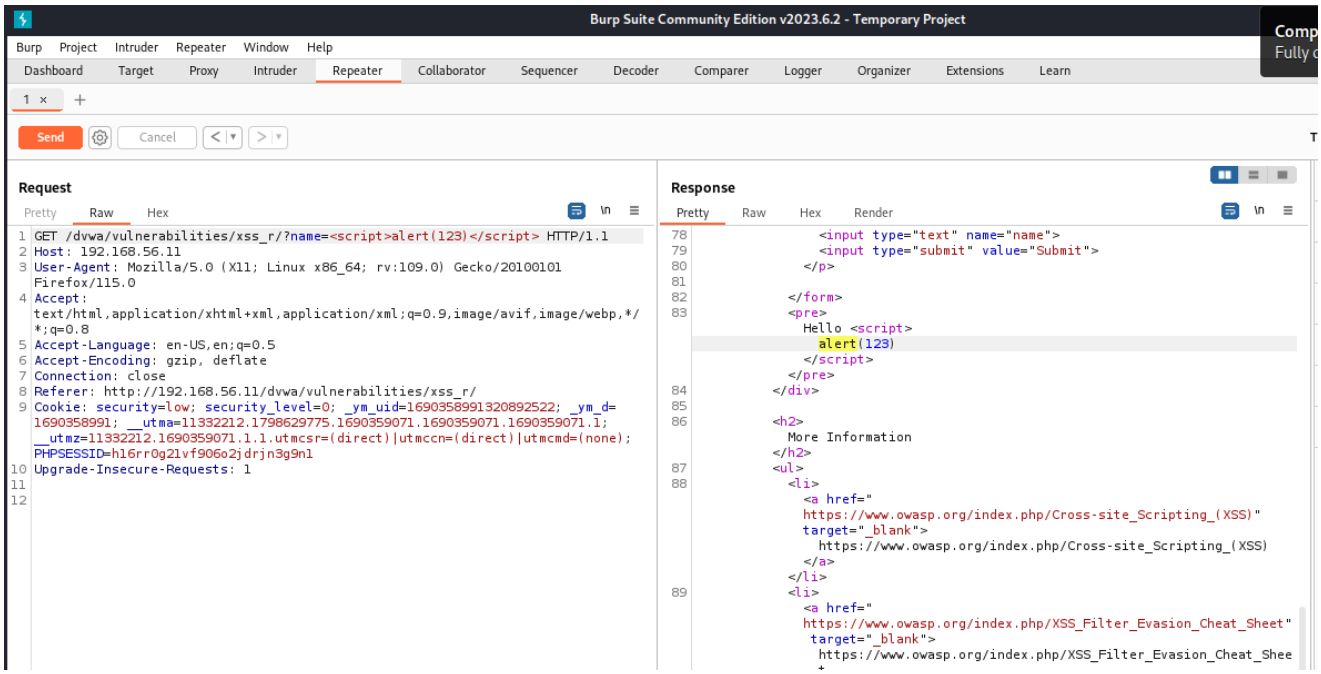
Данные не фильтруются, отображаются 1:1 в коде.

- Запускаем вектор атаки в Repeater:

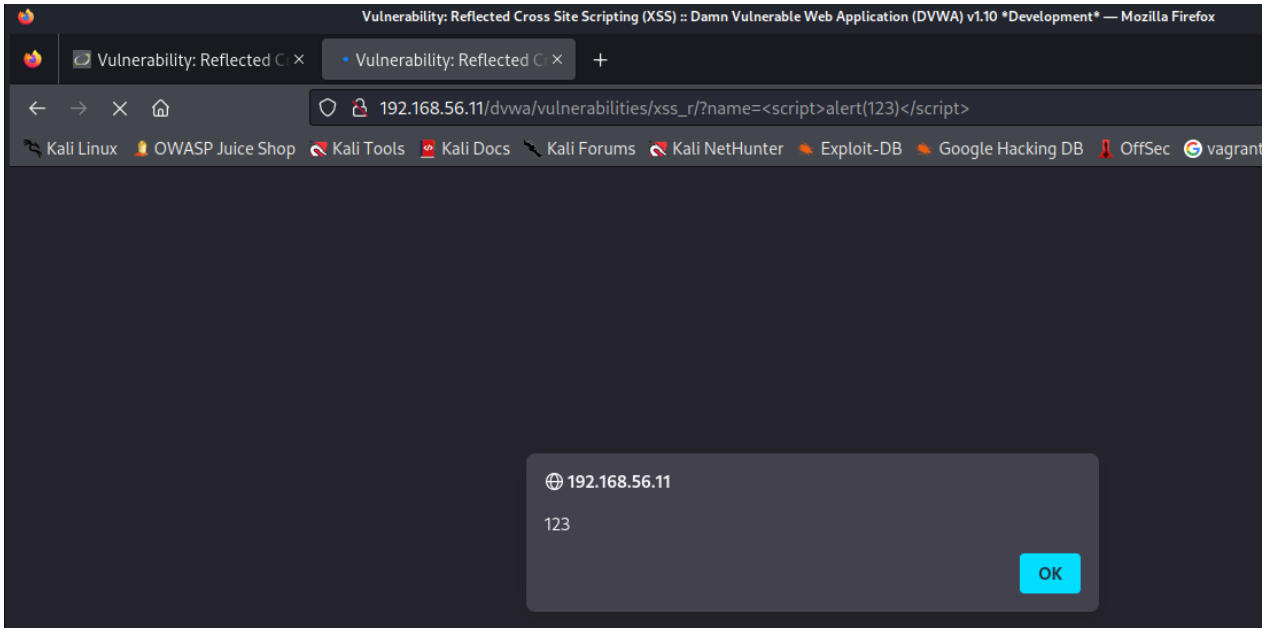


- Проверяем фильтрацию данных
- Вектор атаки:
  - 1. `canary<>123`
  - 2. `<script>alert(123)</script>`





- Отправляем ссылку в браузер из репитера

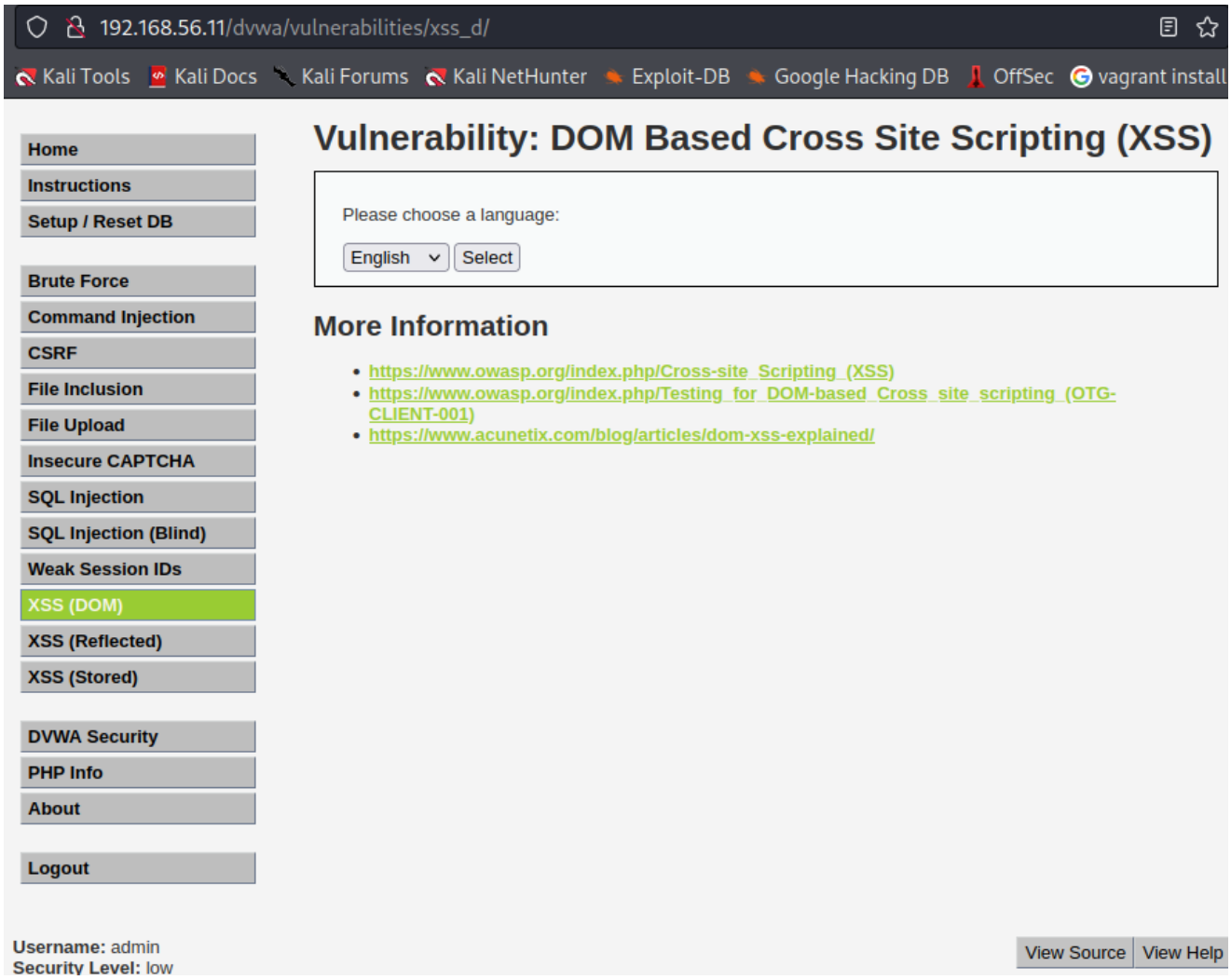


Срабатывает. Вектор на странице не остался, из этого следует, что это отраженный XSS.

Задание_3:

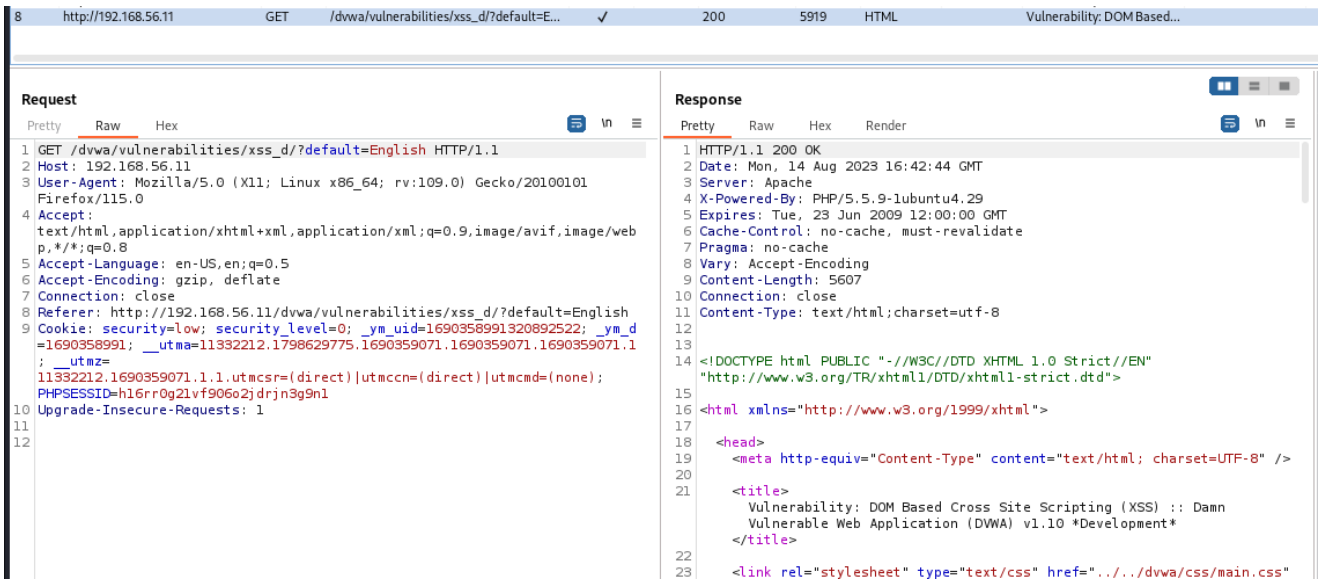
Найдите XSS на странице XSS(DOM) проекта DVWA на простом (Low) уровне сложности и определите ее тип. Составьте отчет о найденной уязвимости. В отчете укажите, в каком контексте присутствует XSS.

- http://192.168.56.11/dvwa/vulnerabilities/xss_d/

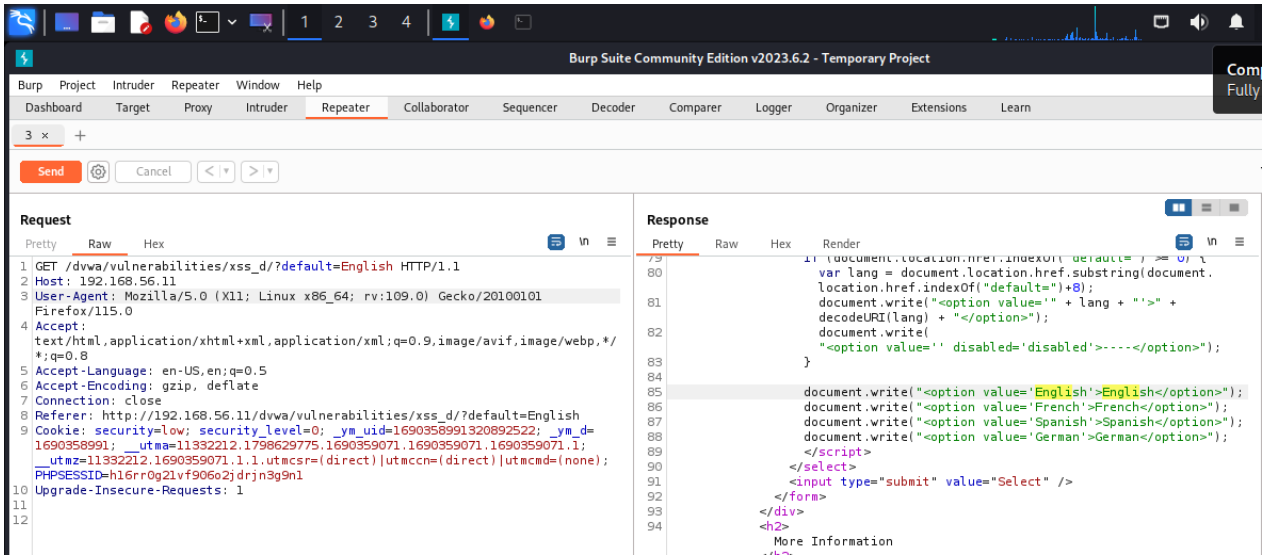



```php
<?php
# No protections, anything goes
?>
```

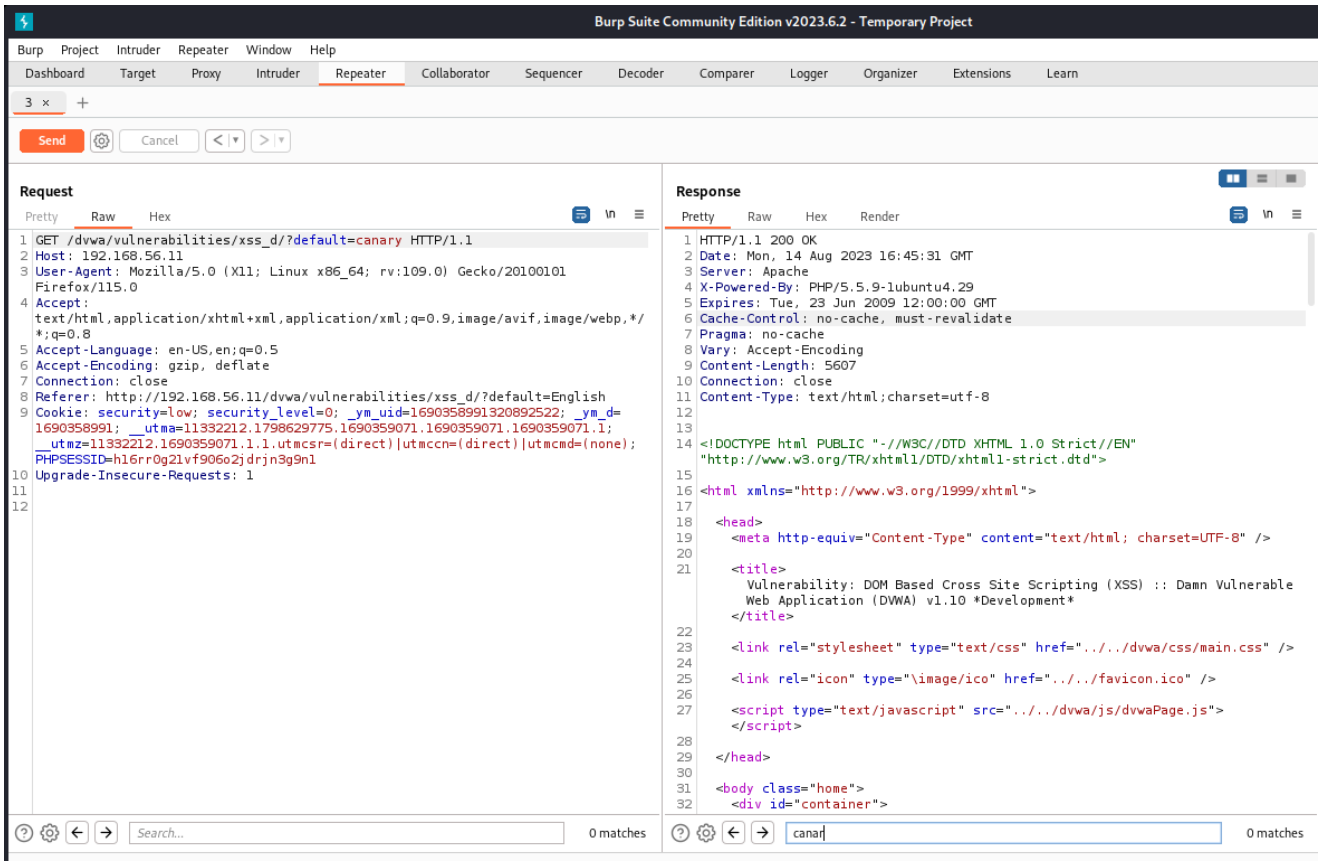
- Отправляем запрос (Select English), отлавливаем его в Burp Suite



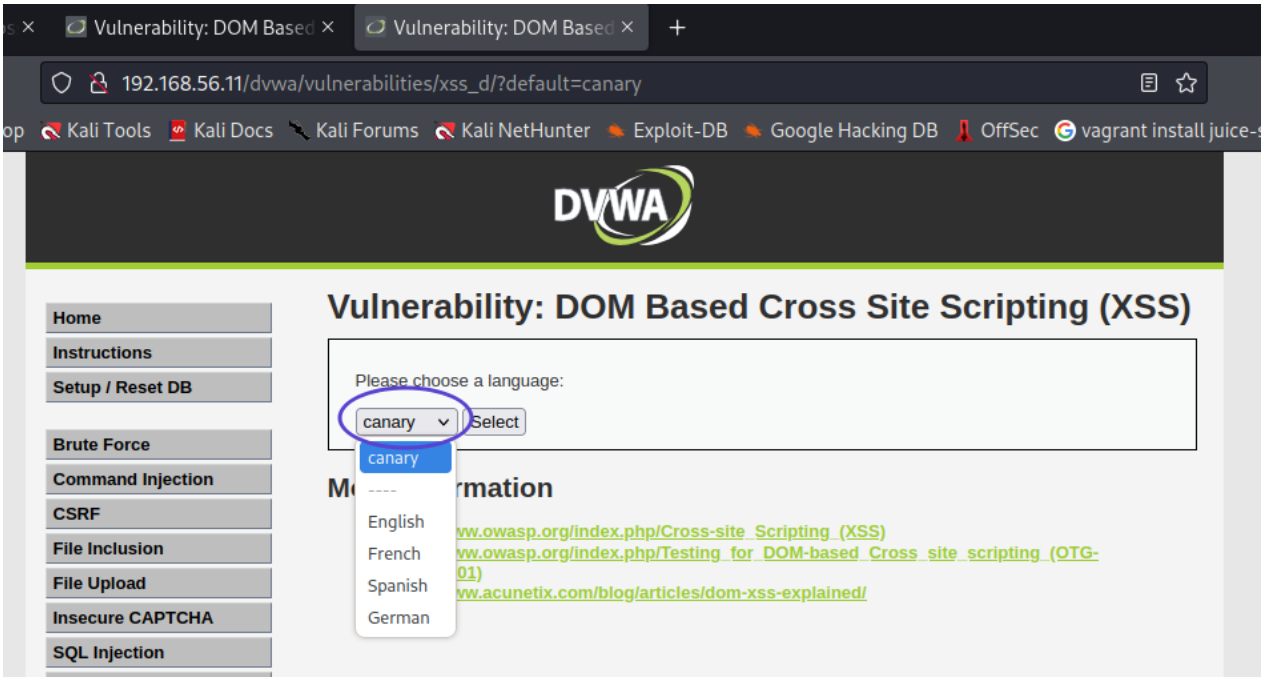
- Отправляем в репитер пробуем применить вектор атаки



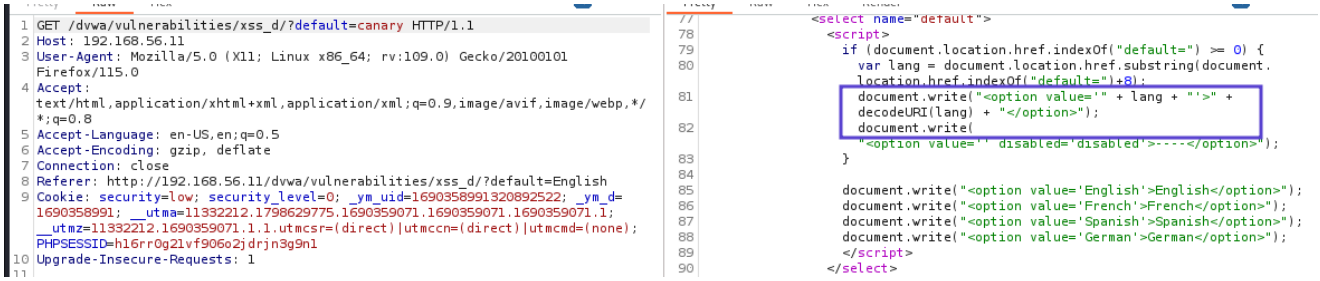
- Задаем `canary`
  Видим, что в тексте не присутствует



- Отправляем ссылку в браузер из репитера, видим, что уже можно выбрать Canary



из чего следует, что прописывается между тегами Option

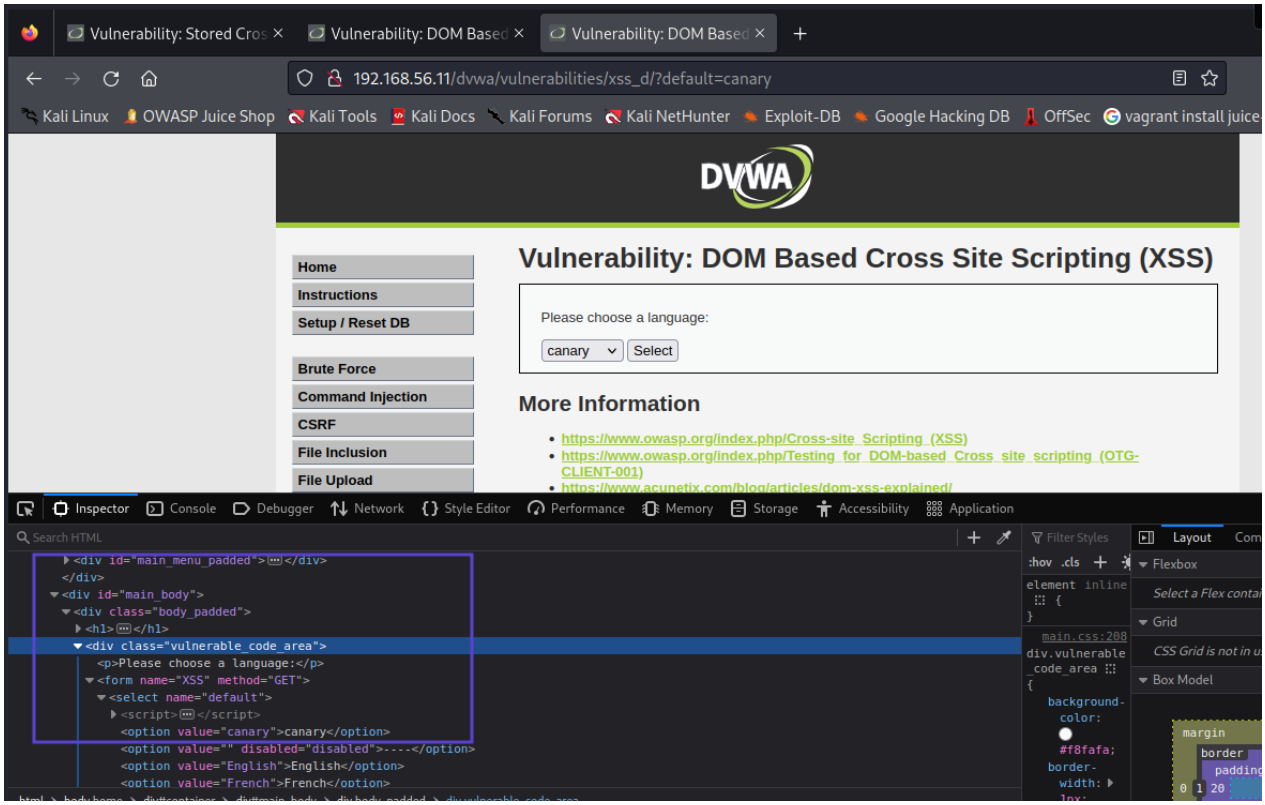


```
GET /dvwa/vulnerabilities/xss_d/?default=canary HTTP/1.1
Host: 192.168.56.11
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: close
Referer: http://192.168.56.11/dvwa/vulnerabilities/xss_d/?default=English
Cookie: security=low; security_level=0; _ym_uid=1690358991320892522; _ym_d=1690358991; __utma=11332212.1798629775.1690359071.1690359071.1690359071.1; __utmz=11332212.1690359071.1.1.utmcsr=(direct)|utmccn=(direct)|utmcmd=(none); PHPSESSID=h16rr0g21vf906o2jdrjn3g9n1
Upgrade-Insecure-Requests: 1
```

```
<select name="default">
  <script>
    if (document.location.href.indexOf("default=") >= 0) {
      var lang = document.location.href.substring(document.location.href.indexOf("default=")+8);
      document.write("<option value='" + lang + "'>" + decodeURI(lang) + "</option>");
      document.write("<option value='' disabled='disabled'>----</option>");
    }

    document.write("<option value='English'>English</option>");
    document.write("<option value='French'>French</option>");
    document.write("<option value='Spanish'>Spanish</option>");
    document.write("<option value='German'>German</option>");
  </script>
</select>
```

- См в Inspector браузера



Из чего следует, что это УЯ DOM XSS (base)

Задание_4:

(*) Выполните установку компонентов, необходимых для выполнения практических занятий, согласно инструкции из методички к уроку 2.

Задание_5:

(*) Найдите максимальное число XSS на странице Pen Test Tool Lookup (AJAX Version) в OWASP Mutillidae, определите их тип и составьте отчет о найденных уязвимостях.

Выводы:

- ...

> Вся информация в данной работе представлена исключительно в ознакомительных целях! Любое использование на практике без согласования тестирования подпадает под действие УК РФ

- https://gb.ru

Выполнил: AndreiM