Name: reut abergel

SID :s8

Unit: TMagen7736.38

Program code: xe103

Date:28/8/25

Linux Project - Guide Manual

Table of contents:

1) Introduction

Hey!
This is a friendly guide for my Linux script **System info for everyone**. The script's job is simple in a few seconds it prints a clean report about your machine It's perfect for class labs, quick troubleshooting, or just learning how Linux exposes system information

This script is called **System info for everyone** because everyone can activate it with this manual (:

it prints a clean summary of your machine:

Public IP

Private (local) IP

MAC address

Top 5 processes by CPU

Memory usage

Running services

Top 10 largest files in /home

Who this is for:

Beginners, students who want quick visibility into their system.

How the guide helps:

Explains the **purpose** of the tool and when to use it

Shows **exactly how to run** the script and read the output

Describing **what each command does** and shows you how it looks on the terminal , so you understand what it prints and why

What you'll need:

A Linux **Kali** machine

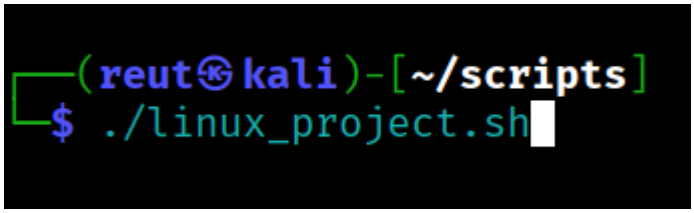
permission's to your kali machine

Internet connectivity for the public IP check

How to run

I already gave the script execute permissions all you need to do after you downloaded the file is to write

./linux.project.sh



2) Goal of the tool

The goal is to give you an **instant report** of your system.
Instead of typing many commands and scrolling long outputs, you get one compact report you can copy, screenshot, or attach to homework.

3) Summary

You now have a simple one-file tool that pulls together the system facts that matter **public/private IPs, MAC address ,Top 5 processes for CPU, memory usage, running services, and the largest files** into one clean, repeatable report you can run in seconds. its beginner-friendly, and helps you avoid juggling many separate commands

What I learned building this script:

how to see and read **processes** and **running services**

How to get a clean output with command like **awk**, **sort** ,**uniq** inside the script

And How to combine small Linux commands into a practical script to make things easier

What each command does

The exact wording in your echo "here" messages

```
echo '[=>]this is the memory usege statistics:'
```

can be whatever you like^

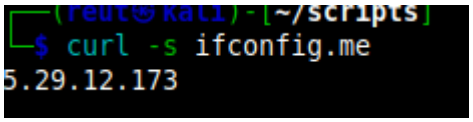
those are the commands in the script and their explanation:

Public IP:

```
#Identify the system's public IP
echo "[=>]system public ip is:$(curl -s ifconfig.me)"
```

---

curl -s ifconfig.me



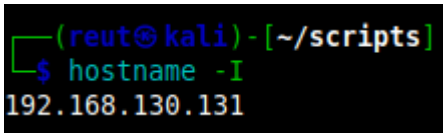
What it shows: the public IP of your computer (the wan ip)

Private IP:

```
#Identify the private IP address assigned to the system's network interface
echo "[=>]this is the user privte ip:$(hostname -I)"
```

---

hostname -I



What it shows: your privet IP address on the Lan.

MAC address:

```
#display the mac address
echo "[=>]this is the system mac address:$(ifconfig eth0 |grep ether |awk '{print $2}')"
```

ifconfig eth0 | grep ether | awk '{print $2}'

If your interface is not eth0, change it

```
┌──(reut㉿kali)-[~/scripts]
└─$ ifconfig eth0 |grep ether |awk '{print $2}'
00:0c:29:14:24:90
```

Top 5 processes for CPU:

```
#Display the percentage of CPU usage for the top 5 processes.
echo '[=>]the percentage of CPU usage for the top 5 processes is:'
top -b -n 1 -o %CPU | sed -n '8,12p'
```

top -b -n 1 -o %CPU | sed -n '8,12p'

What it shows: five lines listing the busiest processes by CPU at that moment. (Batch mode -b, one snapshot -n 1, sort by CPU -o %CPU, then print lines 8–12.)

```
top -b -n 1 -o %CPU | sed -n '8,12p'
    1 root      20   0   24284  14740  10600 S   0.0   0.2   0:07.86 systemd
    2 root      20   0       0      0      0 S   0.0   0.0   0:00.19 kthreadd
    3 root      20   0       0      0      0 S   0.0   0.0   0:00.00 pool_wo+
    4 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kworker+
    5 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kworker+
```

Memory usage:

```
#Display memory usage statistics: total and available memory
echo '[=>]this is the memory usege statistics:'
 free -h | awk 'NR==1{print "Total","Available"} NR==2{print $2,$7}'
```

free -h | awk 'NR==1{print "Total","Available"} NR==2{print $2,$7}'

What it shows: total and **available** RAM (**h**uman-readable).

```
┌──(reut㉿kali)-[~/scripts]
└─$  free -h | awk 'NR==1{print "Total","Available"} NR==2{print $2,$7}'
Total Available
7.7Gi 6.1Gi
```

Running services:

```
#List active system services with their status
echo '[=>]**the runing services and there names are:'
systemctl list-units --type=service --state=running |awk '{print $1,$4}' |tail -27|head -21
```

systemctl list-units --type=service --state=running | awk '{print $1, $4}'|

tail -27|head -21

What it shows: shows a report of the currently running services and their state.

```
┌──(reut㉿kali)-[~/scripts]
└─$ systemctl list-units --type=service --state=running |awk '{print $1,$4}' |tail -27|head -21
colord.service running
cron.service running
dbus.service running
getty@tty1.service running
haveged.service running
lightdm.service running
ModemManager.service running
NetworkManager.service running
open-vm-tools.service running
pcscd.service running
polkit.service running
rsyslog.service running
rtkit-daemon.service running
systemd-journald.service running
systemd-logind.service running
systemd-timesyncd.service running
systemd-udevd.service running
udisks2.service running
upower.service running
user@1000.service running
vsftpd.service running
```

Top 10 largest files in /home:

```
#Locate the Top 10 Largest Files in /home
echo '[=>]**this are the top 10 largest files in the /home directory'
find /home -type f -exec du -h {} + 2>/dev/null | sort -hr | head -n 10
```

find /home -type f -exec du -h {} + 2>/dev/null | sort -hr | head -n 10

What it shows: the biggest space users in /home directory.

```
┌──(reut㉿kali)-[~/scripts]
└─$ find /home -type f -exec du -h {} + 2>/dev/null | sort -hr | head -n 10
2.1G    /home/reut/shadow9/mylist.txt
133M    /home/reut/Downloads/tor-browser/Browser/libxul.so
131M    /home/reut/tools/john-latest/.git/objects/pack/pack-06b1fb7aa8fda9daa1ad43bb2a2441269a0deaa6.pack
118M    /home/reut/Downloads/tor-browser-linux-x86_64-14.5.6.tar.xz
118M    /home/reut/Desktop/tor-browser-linux-x86_64-14.5.6.tar.xz
118M    /home/reut/.cache/vmware/drag_and_drop/WGA6Tp/tor-browser-linux-x86_64-14.5.6.tar.xz
33M     /home/reut/Downloads/tor-browser/Browser/TorBrowser/Data/Tor/cached-microdescs.new
27M     /home/reut/Downloads/tor-browser/Browser/browser/omni.ja
23M     /home/reut/tools/john-latest/run/john
20M     /home/reut/scripts/auth.log.5
```

How to read the output You'll see a size on the left and on the right the directory and its path (largest first)