PROJECT

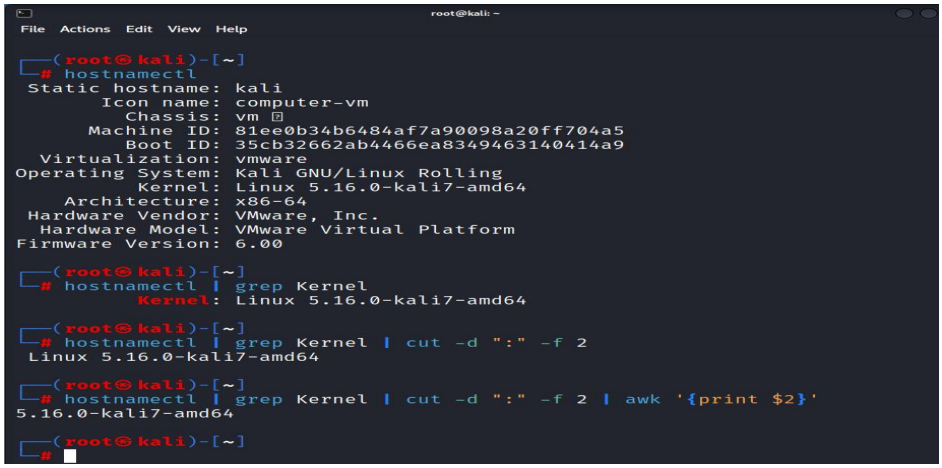
LINUX FUNDAMENTALS

Objective

Create automation to display the Linux operating system information.

1. Display the Linux version.

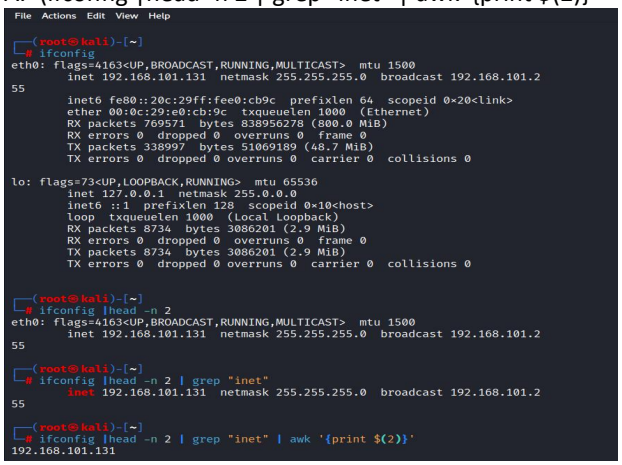
below command was and the following text manipulation
hostnamectl | grep Kernel | cut -d ":" -f 2 | awk '{print $2}'
Below is the break down of the build-up to getting the linux version.



2. Display the private IP address, public IP address, and the default gateway.

Command used:
"Private IP: $(ifconfig |head -n 2 | grep "inet" | awk '{print $(2)}')"
"Public IP: $(curl -s ifconfig.co)"
"Default gateway: $(ip r | grep "via" | awk '{print $(3)}')"

A. (ifconfig |head -n 2 | grep "inet" | awk '{print $(2)}'



B. curl -s ifconfig.co)

```
┌──(root㉿kali)-[~]
└─# curl ifconfig.co
105.113.18.118

┌──(root㉿kali)-[~]
└─# curl -s ifconfig.co
105.113.18.201

┌──(root㉿kali)-[~]
└─#
```

C. ip r | grep "via" | awk '{print $(3)}')

```
┌──(root㉿kali)-[~]
└─# ip r
default via 192.168.101.2 dev eth0 proto dhcp src 192.168.101.131 metric 100
192.168.101.0/24 dev eth0 proto kernel scope link src 192.168.101.131 metric 100

┌──(root㉿kali)-[~]
└─# ip route show default
default via 192.168.101.2 dev eth0 proto dhcp src 192.168.101.131 metric 100

┌──(root㉿kali)-[~]
└─# ip r
default via 192.168.101.2 dev eth0 proto dhcp src 192.168.101.131 metric 100
192.168.101.0/24 dev eth0 proto kernel scope link src 192.168.101.131 metric 100

┌──(root㉿kali)-[~]
└─# ip r | grep "via"
default via 192.168.101.2 dev eth0 proto dhcp src 192.168.101.131 metric 100

┌──(root㉿kali)-[~]
└─# ip r | grep "via" | awk '{print $(3)}'
192.168.101.2

┌──(root㉿kali)-[~]
└─#
```

3. Display the hard disk size; free and used space.

Command used:

echo -e "Size: $(df -H | grep -e "sd" -e "Filesystem" | awk '{print $2}' | tail -n1) \t Used: $(df -H | grep -e "sd" -e "Filesystem" | awk '{print $3}' | tail -n1) \t Free: $(df -H | grep -e "sd" -e "Filesystem" | awk '{print $4}' | tail -n1)"

Part 1 - Disk size

df -H | grep -e "sd" -e "Filesystem" | awk '{print $2}' | tail -n1

Part 2 - Used

df -H | grep -e "sd" -e "Filesystem" | awk '{print $3}' | tail -n1

Part 3 - Free

(df -H | grep -e "sd" -e "Filesystem" | awk '{print $4}' | tail -n1

Note: All put together with produce this out put

```
┌──(root㉿kali)-[~]
└─# echo -e "Size: $(df -H | grep -e "sd" -e "Filesystem" | awk '{print $2}' | tail -n1) \t
Used: $(df -H | grep -e "sd" -e "Filesystem" | awk '{print $3}' | tail -n1) \t Free: $(df -H
 | grep -e "sd" -e "Filesystem" | awk '{print $4}' | tail -n1)"
Size: 83G        Used: 18G       Free: 61G

┌──(root㉿kali)-[~]
└─#
```

4. Display the top five (5) directories and their size.

Command used:

du -sh /* 2>/dev/null | sort -rh | head -n5

Output:

```
File  Actions  Edit  View  Help

┌──(root㉿kali)-[~]
└─# du -sh /* 2>/dev/null | sort -rh | head -n5
9.4G    /usr
3.5G    /root
2.3G    /var
1.2G    /home
185M    /opt

┌──(root㉿kali)-[~]
└─#
```

du - Directory usage, -sh --- summarise $ present human readable, 2>/dev/null - don't output error
And , | sort -rh | head -n5 - to sort in reverse order of hierarchy and highlight top five

5. Display the CPU usage; refresh every 10 seconds.

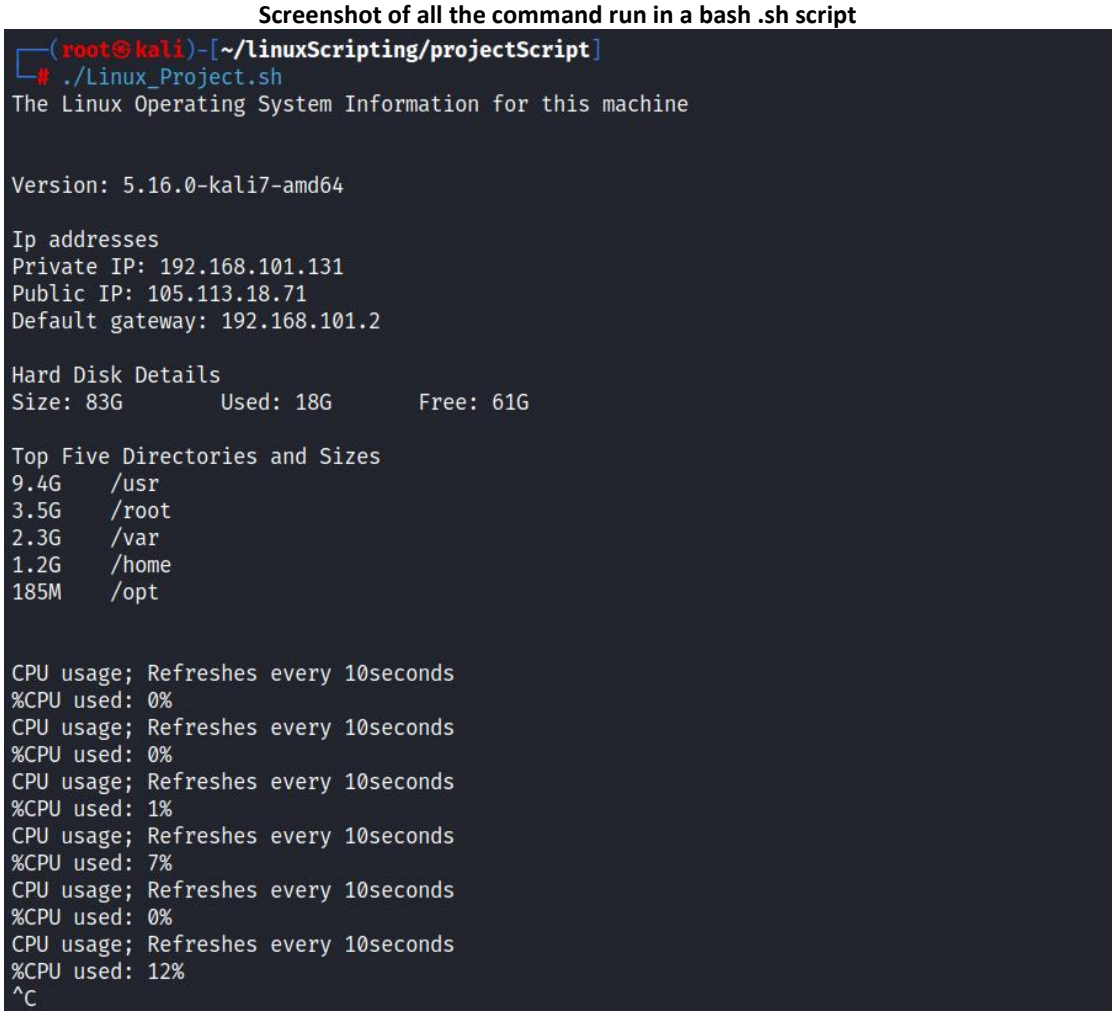
while true
do
echo "CPU usage; Refreshes every 10seconds"
echo "%CPU used: $(top -b -n 5 -d1 | grep "%Cpu(s)" | tail -n 1 | awk '{print $2}' | awk -F. '{print $1}')%"
sleep 10
Done

Note: I did this by wrapping it in a function in my script check my .sh script for this project.

```
#5.
function CPUusage()
{
    while true
    do
        echo "CPU usage; Refreshes every 10seconds"
        echo "%CPU used: $(top -b -n 5 -d1 | grep "%Cpu(s)" | tail -n 1 | awk '{print $2}'
        sleep 10

    done
}
CPUusage
```

```
┌──(root㉿kali)-[~]
└─# top -b -n 5 -d1 | grep "%Cpu(s)" | tail -n 1 | awk '{print $2}' | awk -F. '{print $1}'
0

┌──(root㉿kali)-[~]
└─# echo "%CPU used: $(top -b -n 5 -d1 | grep "%Cpu(s)" | tail -n 1 | awk '{print $2}' | awk
 -F. '{print $1}')%"
%CPU used: 0%

┌──(root㉿kali)-[~]
└─#
```

Screenshot of all the command run in a bash .sh script



End