



WILSON LAU WEI QIANG



INTRODUCTION

In digital age, understanding the configuration, performance and managing information about your device is essential for maintenance and optimization. The info extractor will serve as a handy tool for system administrators, developers, power users to extract, collect, utilize and enhance their research, decision-making, operational processes, and competitive advantage in today's data-rich environment. The purpose creating a script is to provide a user-friendly interface for extracting information from a system, compile detailed reports on system hardware, software, and performance metrics and to facilitate the troubleshooting process by offering comprehensive system insights.

Methodology

Public IP

“Curl” -- command line tool is widely used to interact with web services and API.

‘-s’ -- The ‘-s’ flag in ‘curl’ stands for "silent mode"

Lan IP & MAC Address

“ifconfig” -- It is used to display information about LAN IP and MAC address.

CPU USAGE

“ps” -- stands for "process status", used to display information about processes running on the system.

‘-e’ -- used to select all processes in the system.

‘o’ -- specify the output format.

‘comm%cpu’-- is a command name to display with %cpu for each process.

'--sort=-%cpu' -- Sorts the output based on the CPU usage.

Memory Ram

"free" -- display information about the system's memory usage, including both physical RAM and swap space.

'-h' -- option is used to display the output in a human-readable format.

Active System Service Status

"systemctl" -- used to manage units of the systemd system and service manager.

'list-units': This subcommand is used to list loaded units.

'--type=service': This option specifies that only services should be listed.

'--state=active': This option filters the output to show only units that are currently active.

Show DIR files by size

"du" -- used to estimate file and directory space usage.

'-a' -- Displays disk usage for all files and directories.

'-h' -- Displays disk usage in a human-readable format.

Public IP

```

29
30 #Fetch public IP using ifconfig.
31 PUBLIC_IP=$(curl -s ifconfig.io)
32
33 #Display public IP address.
34 echo -e "\nYour Public IP Address:"
35
36 #=====
37 "

```

```

(kali@kali)-[~]
$ curl -s ifconfig.io
.203.37

```

“Curl” command line tool is widely used to interact with web services and APIs. Will be using ifconfig.io website to obtain our public IP. We still can obtain from ifconfig.me /ifconfig.co /ip.guide.

Lan IP & MAC Address

```

37
38 #Fetch LAN IP using ifconfig.
39 LAN_IP=$(ifconfig | grep inet | awk '{print $2}' | head -n1)
40
41 #Display public IP address.
42 echo -e "\nYour Internal IP Address:\n${YELLOW}$LAN_IP${WHITE}"
43
44 #=====
45 #=====
46
47 #Fetch MAC Address.
48 MAC_ADD=$(ifconfig | grep ether | awk '{print $2}' | sed -r 's/([0-9a-fA-F]{2}){3}/xx:xx:xx/')
49
50
51 #Display MAC Address.
52 echo -e "\nYour Mac Address:\n${YELLOW}$MAC_ADD${WHITE}"
53
54 #=====
55 #=====
56

```

```

(kali@kali)-[~]
$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      inet 192.168.1.37 netmask 255.255.255.0 broadcast 192.168.1.255
      ether 08:00:27:1d:18:1e txqueuelen 1000 (Ethernet)

```

By flags such as ‘grep inet / grep ether’, awk and head to grab just the IP and MAC address output.

CPU USAGE

```
58 #Fetch Top5 CPU usage in %.
59 CPU5=$(ps -eo comm,%cpu --sort=-%cpu | head -n 6 | tail -n +2)
60
61 #Display Top5 CPU usage in %.4d res
62 echo -e "\nYour Top 5 CPU Usage In Percentage:\n${YELLOW}$CPU5${WHITE}"
63
64 #=====
65 #=====

(kali㉿kali)-[~]
$ ps -eo comm,%cpu --sort=-%cpu | head -n 6 | tail -n +2
Xorg          0.9
geany         0.3
xfwm4         0.1
qterminal     0.1
panel-15-genmon 0.1
```

By adding additional flags “-eo comm,%cpu –sort-%cpu” the output will display a list of processes along with their command names and the percentage of CPU usage, sorted in descending order based on CPU usage.

Memory Ram Used & Free

```
66 #Fetch Top5 Memory usage in %.
67 MEM5=$(free -h | awk 'NR==1 {print $1,$2,$3} NR==2 {print $1,$2,$3,$4}')
68
69 #Display Top5 Memory usage in %.
70 echo -e "\nYour Top 5 Memory Usage In Percentage:\n${YELLOW}$MEM5${WHITE}"
71
72 #=====

(kali㉿kali)-[~]
$ free -h

```

	total	used	free	shared	buff/cache	available
Mem:	1.9Gi	853Mi	784Mi	9.1Mi	490Mi	1.1Gi
Swap:	1.0Gi	0B	1.0Gi			

Adding flag ‘-h’ is to display the memory sizes in a human-readable format. Using ‘awk NR==’ I can grab which lines and column to show as output.

Active System Service Status

```

75 #Fetch Active System Service and Status .
76 System_Service=$(systemctl list-units --type=service --state=active )
77
78 #Display Active System Service and Status.
79 echo -e "\nYour Active Service and Status:\n$(YELLOW)$System_Service$(WHITE)"
80
81
82
83

```

UNIT	LOAD	ACTIVE	SUB	DESCRIPTION
accounts-daemon.service	loaded	active	running	Accounts Service
colord.service	loaded	active	running	Manage, Install and Generate Color Pro
console-setup.service	loaded	active	exited	Set console font and keymap
cron.service	loaded	active	running	Regular background program processing
dbus.service	loaded	active	running	D-Bus System Message Bus
getty@tty1.service	loaded	active	running	Getty on tty1
haveged.service	loaded	active	running	Entropy Daemon based on the HAVEGE alg
ifupdown-pre.service	loaded	active	exited	Helper to synchronize boot up for ifup

lines 1-9

By adding flags such as “list-units’ used to list unit in the system, “-state=active`: This option filters the list to show only units that are currently in the active state.

Display /home top 10 files by size.

```

84 #Fetch Top 10 Files by size from /home dir .
85 TOP10_SIZE=$(du -ah /home | sort -n -r | head -n 10)
86
87 #Display Top10 Files by size from /home dir.
88 echo -e "\nTop 10 Files by size from /HOME/:\n$(YELLOW)$TOP10_SIZE$(WHITE)"
89
90
91

```

```

$ sudo du -ah /home | sort -n -r | head -n 10
1008K /home/kali/geany-2.0/scintilla/lexilla/lexers/.libs/liblexilla_la-LexGDScript.o
992K /home/kali/geany-2.0/ctags/dsl
956K /home/kali/.mozilla/firefox/e2m73iyf.default-esr/storage/permanent
952K /home/kali/.mozilla/firefox/e2m73iyf.default-esr/storage/permanent/chrome
944K /home/kali/.mozilla/firefox/e2m73iyf.default-esr/storage/permanent/chrome/idb
932K /home/kali/geany-2.0/scintilla/src/.libs/libscintilla_la-Decoration.o
932K /home/kali/geany-2.0/scintilla/lexilla/lexers/.libs/liblexilla_la-LexRust.o
912K /home/kali/geany-2.0/icons
896K /home/kali/geany-2.0/scintilla/lexilla/lexers/.libs/liblexilla_la-LexD.o

```

‘du’ stands for "disk usage". It is used to estimate file space usage. It shows the disk space used by the files and subdirectories in that directory. However, we need ‘sudo’ command to run as it may have restricted permission that prevent regular user from accessing their contents thus allowing it to access all directories and provide a more comprehensive overview of disk usage.

Recommendations

Servers can run on both Linux or Windows operating system thus scripts may have to execute either OS, in order for scripts to able to run on Window OS, we may install 'Ubuntu' on Window OS to execute scripts.

```
root@nasukiZ:/mnt/c# bash script.sh

HELLO WORLD

Your Public IP Address:
[REDACTED]

Your Internal IP Address:
172.21.42.211

Your Mac Address:
[REDACTED] 51:ba:09

Your Top 5 CPU Usage In Percentage:
python3.10 0.2
snapfuse 0.1
python3 0.1
systemd 0.0
init-systemd(Ub 0.0

Your Top 5 Memory Usage In Percentage:
total free shared
Mem: 524Mi 14Gi

Your Active Service and Status:
UNIT                                LOAD    ACTIVE SUB    DESCRIPTION
apport.service                     loaded active exited LSB: automatic crash report generation
console-getty.service              loaded active running Console Getty
console-setup.service             loaded active exited Set console font and keymap
cron.service                       loaded active running Regular background program processing daemon
dbus.service                      loaded active running D-Bus System Message Bus
getty@tty1.service                loaded active running Getty on tty1
keyboard-setup.service            loaded active exited Set the console keyboard layout
networkd-dispatcher.service       loaded active running Dispatcher daemon for systemd-networkd
packagekit.service               loaded active running PackageKit Daemon
plymouth-quit-wait.service         loaded active exited Hold until boot process finishes up
plymouth-quit.service             loaded active exited Terminate Plymouth Boot Screen
plymouth-read-write.service        loaded active exited Tell Plymouth To Write Out Runtime Data
polkit.service                    loaded active running Authorization Manager
rsyslog.service                   loaded active running System Logging Service
setvtrgb.service                  loaded active exited Set console scheme
snap.ubuntu-desktop-installer.subi loaded active running Service for snap application ubuntu-desktop-installer.subi
quity-server.service              loaded active running
snapd.seeded.service              loaded active exited Wait until snapd is fully seeded
snapd.service                     loaded active running Snap Daemon
systemd-journal-flush.service      loaded active exited Flush Journal to Persistent Storage
```

Links:

Ubuntu : <https://ubuntu.com/>