



Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

Experiment No.2

Linux shell script

2.1 Write shell scripts to do the following:

- a. Display OS version, release number, kernel version
- b. Display top 10 processes in descending order
- c. Display processes with highest memory usage.
- d. Display current logged in user and log name. Display current shell, home directory, operating system type, current path setting, current working directory

Date of Performance:

Date of Submission:

Marks:

Sign:



Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

Aim: Write shell scripts to: a. Display OS version, release number, and kernel version b. Display top 10 processes in descending order c. Display processes with highest memory usage d. Display current logged in user and log name e. Display current shell, home directory, operating system type, current path setting, current working directory

Objective: Perform some operation on os.

Theory:

Shell is a user program, or its environment is provided for user interaction. It is a command prompt within Linux where you can type commands. It is a program that takes your commands from the keyboard and gives them to the OS to perform. Shell is not part of system KERNAL but it uses system KERNAL to execute programs, create files, etc. A Shell Script is a text file that contains a sequence of commands for a UNIX based OS. It is called a Shell Script because it combines into a "Script" in a single file a sequence of commands, that would otherwise have to be presented to the system from a keyboard one at a time. A Shell Script is usually created for command sequences for which a user has a repeated need. You initiate the sequence of commands in Shell Script by simply entering the name of the Shell Script on a command line.

Types of Shell

Script :-1. sh -

Simple Shell

2. bash - Bourne Again

Shell 3. ksh - Korne

Shell



Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

4. csh - C Shell

5. ssh - Secure Shell

To use a particular Shell type the Shell name at the command prompt. Eg:-

\$csh - It will switch the current Shell to C Shell. To view the current Shell that is being used, type echo \$ SHELL at the command prompt.

Code :

a. Display OS version, release number, and kernel version:

bash

Copy code

```
#!/bin/bash
```

```
# Display OS version
```

```
echo "OS Version:"
```

```
cat /etc/os-release | grep PRETTY_NAME
```

```
# Display release number
```

```
echo "Release Number:"
```

```
cat /etc/os-release | grep VERSION_ID
```

```
# Display kernel version
```

```
echo "Kernel Version:"
```

```
uname -r
```

b. Display top 10 processes in descending order:

bash

Copy code

```
#!/bin/bash
```

```
# Display top 10 processes
```

```
echo "Top 10 Processes:"
```

```
ps aux --sort=-%cpu | head -n 11
```

c. Display processes with highest memory usage:

bash

Copy code

```
#!/bin/bash
```

```
# Display processes with highest memory usage
```



Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

echo "Processes with Highest Memory Usage:"

```
ps aux --sort=-%mem | head -n 11
```

d. Display current logged in user and log name:

bash

Copy code

```
#!/bin/bash
```

```
# Display current logged in user
```

```
echo "Current Logged in User:"
```

```
whoami
```

```
# Display log name
```

```
echo "Log Name:"
```

```
who
```

e. Display current shell, home directory, operating system type, current path setting, current working directory:

bash

Copy code

```
#!/bin/bash
```

```
# Display current shell
```

```
echo "Current Shell:"
```

```
echo $SHELL
```

```
# Display home directory
```

```
echo "Home Directory:"
```

```
echo $HOME
```

```
# Display operating system type
```

```
echo "Operating System Type:"
```

```
uname -o
```

```
# Display current path setting
```

```
echo "Current Path Setting:"
```

```
echo $PATH
```

```
# Display current working directory
```

```
echo "Current Working Directory:"
```

```
pwd
```



Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

You can save each script in a separate file with a .sh extension, make it executable using `chmod +x script_name.sh`, and then execute it using `./script_name.sh`.

These scripts will provide the requested information on a Linux system.

Output :

```
sudo apt install re
b17gb17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ rm bar1
b17gb17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ cp bar1
cp: missing destination file operand after 'bar1'
Try 'cp --help' for more information.
b17gb17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ cd bar1
b17gb17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/bar1$ cp
cp: missing file operand
Try 'cp --help' for more information.
b17gb17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/bar1$ mv
mv: missing file operand
Try 'mv --help' for more information.
b17gb17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/bar1$ head bar1
head: cannot open 'bar1' for reading: No such file or directory
b17gb17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/bar1$ head bar1
head: cannot open 'bar1' for reading: No such file or directory
b17gb17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/bar1$ cd..
cd..: command not found
b17gb17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~/bar1$ cd
b17gb17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ head bar1
head: error reading 'bar1': Is a directory
b17gb17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ uname
Linux
b17gb17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$

b17gb17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ uname -r
5.15.0-43-generic
b17gb17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$

For example, try 'man man'.
b17gb17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ man man
b17gb17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ ps
  PID TTY          TIME CMD
 9655 pts/0    00:00:00 bash
10185 pts/0    00:00:00 ps
b17gb17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ hostname
b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC
b17gb17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ w
 3210317 up 53 min, 1 user,  load average: 0.40, 0.34, 0.29
USER          TTY        FROM          LOGIN@   IDLE   JCPU   PCPU WHAT
b17          tty2      tty2          11:10    1      0.00s  0.00s /usr/libexec/gn
```



Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

```
Activities Terminal Feb 23 12:12
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~$ uname -r
5.15.0-43-generic
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~$ ps -sort
error: process ID list syntax error

Usage:
ps [options]

Try 'ps --help <simple|list|output|threads|misc|all>'
or 'ps --help <sl|o|t|m|a>'
for additional help text.

For more details see ps(1).
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~$ ps 10
PID TTY STAT TIME COMMAND
10 ? S 0:00 [mm_percpu_wq]
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~$
```

```
Activities Terminal Feb 23 12:16
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~$ ps
12446 ? I 0:00 [kworker/2:0-cgroup_destroy]
12448 ? S1 0:00 /usr/lib/libreoffice/program/soffice --writer file:///home/b17/Documents/os_exp_1.odt
12464 ? S1 0:07 /usr/lib/libreoffice/program/soffice.bin --writer file:///home/b17/Documents/os_exp_1.odt
12839 ? S1 0:00 /snap/firefox/1635/usr/lib/firefox/firefox -content
proc -chldID 37 -lsForBrowser -prefsLen 38543 -pref
12842 ? S1 0:00 /snap/firefox/1635/usr/lib/firefox/firefox -content
proc -chldID 38 -lsForBrowser -prefsLen 38543 -pref
12978 ? S1 0:00 /snap/firefox/1635/usr/lib/firefox/firefox -content
proc -chldID 39 -lsForBrowser -prefsLen 38543 -pref
13001 ? I 0:00 [kworker/10:2-events]
13002 ? I 0:00 [kworker/0:2-mm_percpu_wq]
13009 ? I= 0:00 [kworker/u25:1-L915_Flip]
13066 ? I 0:00 [kworker/u24:4-ext4-rsv-conversion]
13092 ? I 0:00 [kworker/2:2-events]
13112 ? I 0:00 [kworker/8:1-events]
13113 ? I 0:00 [kworker/4:2-events]
13117 ? S1 0:00 /snap/firefox/1635/usr/lib/firefox/firefox -content
proc -chldID 40 -lsForBrowser -prefsLen 38543 -pref
13145 pts/0 R+ 0:00 ps aux
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~$ whoami
b17
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~$
```

```
Activities Terminal Feb 23 12:20
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~$ ps
12978 ? S1 0:00 /snap/firefox/1635/usr/lib/firefox/firefox -content
proc -chldID 39 -lsForBrowser -prefsLen 38543 -pref
13001 ? I 0:00 [kworker/10:2-events]
13002 ? I 0:00 [kworker/0:2-mm_percpu_wq]
13009 ? I= 0:00 [kworker/u25:1-L915_Flip]
13066 ? I 0:00 [kworker/u24:4-ext4-rsv-conversion]
13092 ? I 0:00 [kworker/2:2-events]
13112 ? I 0:00 [kworker/8:1-events]
13113 ? I 0:00 [kworker/4:2-events]
13117 ? S1 0:00 /snap/firefox/1635/usr/lib/firefox/firefox -content
proc -chldID 40 -lsForBrowser -prefsLen 38543 -pref
13145 pts/0 R+ 0:00 ps aux
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~$ whoami
b17
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~$ echo $0
bash
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~$ lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description: Ubuntu 22.04.1 LTS
Release: 22.04
Codename: jammy
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~$
```




Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

```
Activities Terminal Feb 23 12:21
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~
proc -chldID 39 -lsForBrowser -prefLen 30543 -pref
13001 ? I 0:00 [kworker/18:2-events]
13002 ? I 0:00 [kworker/51:2-mm_percpu_wq]
13009 ? I 0:00 [kworker/0:3:1-1915-flip]
13000 ? I 0:00 [kworker/u24:4-ext4-rsv-conversion]
13002 ? I 0:00 [kworker/2:2-events]
13112 ? I 0:00 [kworker/0:3-events]
13115 ? I 0:00 [kworker/4:12-events]
13117 ? Sl 0:00 /snap/firefox/1635/usr/lib/firefox/firefox -content
13145 pts/0 B+ 0:00 ps aux
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~$ whoami
b17
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~$ echo $0
bash
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~$ lsb_release -a
No LSB modules are available.
Distribution: Ubuntu 22.04.1 LTS
Description: Ubuntu 22.04.1 LTS
Release: 22.04
Codename: jammy
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~$ pwd
/home/b17
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~$

Activities Terminal Feb 23 12:19
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~
12338 ? I 0:00 [kworker/4:1-events]
12446 ? I 0:00 [kworker/0:2-events]
12446 ? I 0:00 [kworker/21:0-cgroup_destroy]
12446 ? Sl 0:00 /usr/lib/libreoffice/program/sooplash --writer File
:///home/b17/Documents/ms_exp_1.odt
12446 ? Sl 0:00 /usr/lib/libreoffice/program/sooplash --writer f
ile:///home/b17/Documents/ms_exp_1.odt
12039 ? Sl 0:00 /snap/firefox/1635/usr/lib/firefox/firefox -content
proc -chldID 37 -lsForBrowser -prefLen 30543 -pref
12042 ? Sl 0:00 /snap/firefox/1635/usr/lib/firefox/firefox -content
proc -chldID 38 -lsForBrowser -prefLen 30543 -pref
12078 ? Sl 0:00 /snap/firefox/1635/usr/lib/firefox/firefox -content
proc -chldID 39 -lsForBrowser -prefLen 30543 -pref
13001 ? I 0:00 [kworker/10:2-events]
13002 ? I 0:00 [kworker/51:2-mm_percpu_wq]
13009 ? I 0:00 [kworker/0:3:1-1915-flip]
13000 ? I 0:00 [kworker/u24:4-ext4-rsv-conversion]
13002 ? I 0:00 [kworker/2:2-events]
13112 ? I 0:00 [kworker/0:1-events]
13115 ? I 0:00 [kworker/4:12-events]
13117 ? Sl 0:00 /snap/firefox/1635/usr/lib/firefox/firefox -content
proc -chldID 40 -lsForBrowser -prefLen 30543 -pref
13145 pts/0 B+ 0:00 ps aux
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~$

Activities Terminal Feb 23 12:02
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~
192 re bart
193 rm bart
194 cp bart
195 cd bart
196 cp
197 mv
198 head bart
199 head bart
200 cd..
201 cd
202 head bart
203 uname
204 history
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~$ man
What manual page do you want?
For example, try 'man man'.
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~$ man man
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~$ ps
PID TTY TIME CMD
9653 pts/0 00:00:00 bash
10183 pts/0 00:00:00 ps
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~$ hostname
b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~$
```



Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

Conclusion:

In conclusion, the shell scripts provided offer a comprehensive toolkit for system administrators and users alike to efficiently manage and monitor their operating environment.

- a. By displaying crucial system information such as OS version, release number, and kernel version, users can quickly assess the configuration of their system.
- b. The script showcasing the top 10 processes in descending order aids in identifying resource-intensive tasks, facilitating smoother system operation.
- c. Identifying processes with the highest memory usage enables users to optimize memory utilization and address potential bottlenecks effectively.
- d. Displaying the current logged-in user and log name provides essential user context, enhancing security and accountability measures.
- e. Lastly, presenting details such as the current shell, home directory, operating system type, path settings, and working directory offers users a comprehensive snapshot of their shell environment, aiding in navigation and customization efforts.

What is a shell?

In computing, a shell is a user interface that allows you to interact with an operating system's services. It's like the intermediary between you and the operating system, providing you with a way to issue commands and receive responses.

There are two primary types of shells:

Command-line shell: This type of shell provides a text-based interface where you type commands and receive text-based output. Examples include the Unix shell (such as Bash, Zsh, and tcsh) and Windows Command Prompt (cmd.exe).

Graphical shell: Also known as a desktop environment, this type of shell provides a graphical user interface (GUI) where you interact with the operating system using windows, icons, buttons, and menus. Examples include the Windows graphical shell (explorer.exe) and desktop environments in Linux distributions like GNOME and KDE.

Shells interpret your commands and translate them into instructions that the operating system can understand and execute. They can also provide features like scripting capabilities, command history, tab completion, and customizable prompts to enhance productivity and ease of use.



Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

Name Different types of shells.

Bash (Bourne Again Shell): Bash is one of the most widely used command-line shells, especially in Unix-like operating systems such as Linux and macOS. It's a successor to the original Bourne Shell (sh) and includes many enhancements, such as command-line editing, history, and scripting capabilities.

Zsh (Z Shell): Zsh is another popular command-line shell that offers additional features beyond Bash, including advanced tab completion, spelling correction, and powerful scripting capabilities. It's highly customizable and is often preferred by power users.

Ksh (Korn Shell): The Korn Shell is another Unix shell that combines features from the Bourne Shell (sh) and C Shell (csh). It offers a rich command-line editing experience, along with powerful scripting capabilities. It's commonly used in commercial Unix environments.

Csh (C Shell): The C Shell is a Unix shell with a syntax inspired by the C programming language. It includes features like command history, aliases, and job control. While less popular than Bash or Zsh, it's still used in some Unix environments.

Tcsh (Tenex C Shell): Tcsh is an enhanced version of the C Shell, with additional features such as command-line editing, filename completion, and job control. It's often used as an interactive shell in Unix-like systems.

Fish (Friendly Interactive Shell): Fish is a modern command-line shell designed for interactive use. It focuses on user-friendly features such as syntax highlighting, autosuggestions, and a powerful scripting language. Fish aims to be easy to use for both beginners and advanced users.

Dash: Dash is a minimalistic Unix shell designed for efficiency and speed. It's often used as the default system shell in some Unix-like operating systems due to its small size and fast execution.

PowerShell: PowerShell is a command-line shell and scripting language developed by Microsoft for Windows operating systems. It's designed to automate administrative tasks and manage system configurations using a command-line interface. PowerShell includes features like object-oriented scripting and integration with the .NET Framework.