

OPERATING SYSTEM LAB – PRACTICAL 1

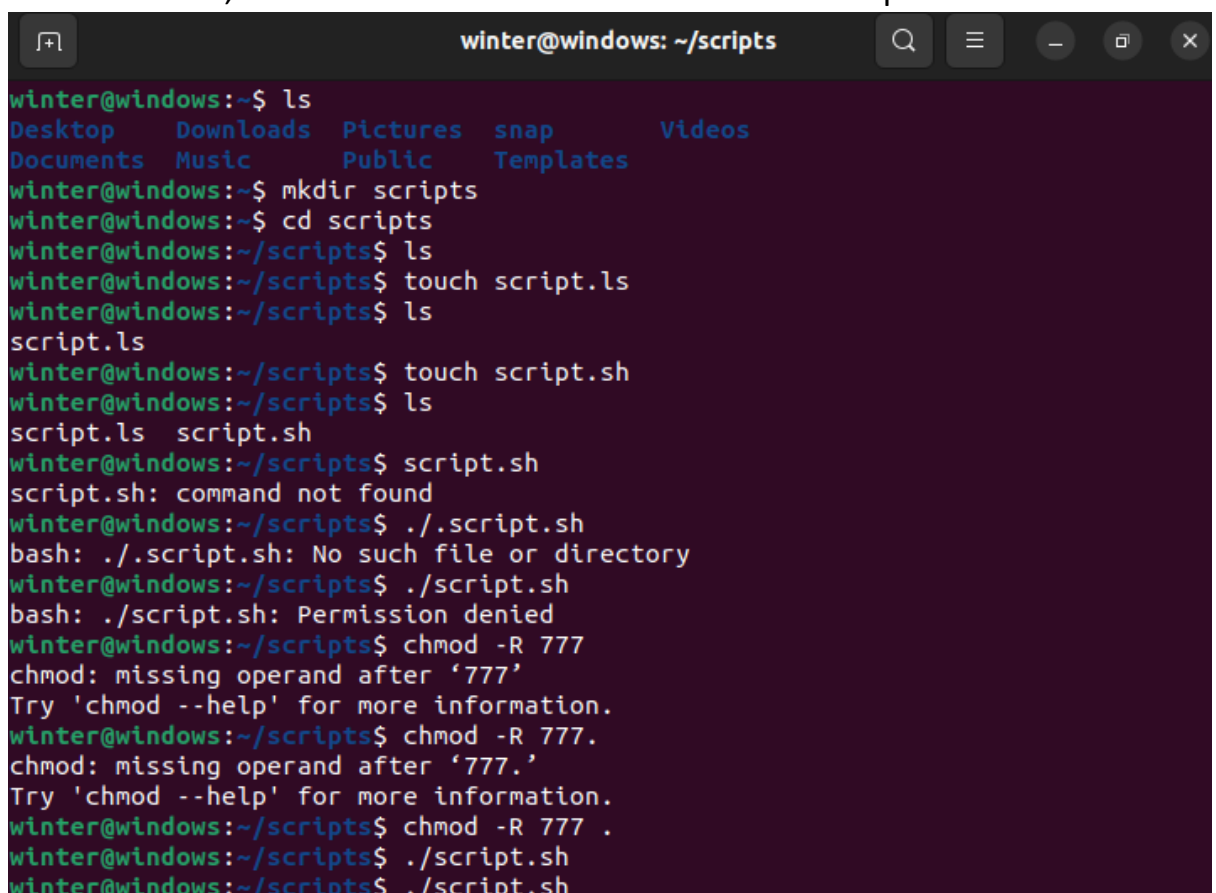
NAME – Sakshi Soni

ROLL NO -13

AIM - To study and execute basic Linux System Commands and write shell scripts to display the system particulars (processor, processes and memory).

QUESTIONS –

- 1) Execution of all the steps given in below snippet. (touch, ls, cd, mkdir , chmod –R 777, how to create and handle and execute script.

A terminal window titled 'winter@windows: ~/scripts' with standard window controls. The terminal shows a series of commands and their outputs. The user starts in the home directory, lists files, creates a 'scripts' directory, moves into it, and creates two files: 'script.ls' and 'script.sh'. When they try to run 'script.sh', they get an error 'command not found'. Then they try './script.sh' and get 'No such file or directory'. Next, they try './script.sh' and get 'Permission denied'. They then attempt 'chmod -R 777' but get an error 'missing operand after '777''. After adding a space, they get another error 'missing operand after '777.''. Finally, they add a dot and successfully run 'chmod -R 777 .' and then execute './script.sh' twice without errors.

```
winter@windows:~$ ls
Desktop  Downloads  Pictures  snap      Videos
Documents Music      Public   Templates

winter@windows:~$ mkdir scripts
winter@windows:~$ cd scripts
winter@windows:~/scripts$ ls
winter@windows:~/scripts$ touch script.ls
winter@windows:~/scripts$ ls
script.ls
winter@windows:~/scripts$ touch script.sh
winter@windows:~/scripts$ ls
script.ls  script.sh
winter@windows:~/scripts$ script.sh
script.sh: command not found
winter@windows:~/scripts$ ./script.sh
bash: ./script.sh: No such file or directory
winter@windows:~/scripts$ ./script.sh
bash: ./script.sh: Permission denied
winter@windows:~/scripts$ chmod -R 777
chmod: missing operand after '777'
Try 'chmod --help' for more information.
winter@windows:~/scripts$ chmod -R 777.
chmod: missing operand after '777.'
Try 'chmod --help' for more information.
winter@windows:~/scripts$ chmod -R 777 .
winter@windows:~/scripts$ ./script.sh
winter@windows:~/scripts$ ./script.sh
```

```
winter@windows: ~/scripts
winter@windows:~$ ls
Desktop  Downloads  Pictures  snap      Videos
Documents Music      Public   Templates
winter@windows:~$ mkdir scripts
winter@windows:~$ cd scripts
winter@windows:~/scripts$ ls
winter@windows:~/scripts$ touch script.ls
winter@windows:~/scripts$ ls
script.ls
winter@windows:~/scripts$ touch script.sh
winter@windows:~/scripts$ ls
script.ls  script.sh
winter@windows:~/scripts$ script.sh
script.sh: command not found
winter@windows:~/scripts$ ./script.sh
bash: ./script.sh: No such file or directory
winter@windows:~/scripts$ ./script.sh
bash: ./script.sh: Permission denied
winter@windows:~/scripts$ chmod -R 777
chmod: missing operand after '777'
Try 'chmod --help' for more information.
winter@windows:~/scripts$ chmod -R 777.
chmod: missing operand after '777.'
Try 'chmod --help' for more information.
winter@windows:~/scripts$ chmod -R 777 .
winter@windows:~/scripts$ ./script.sh
winter@windows:~/scripts$ ./script.sh
./script.sh: line 1: Hello: command not found
winter@windows:~/scripts$ S
```

2) Create a shell script to perform the following tasks.

- (a) Find the number of processors your machine has.
- (b) How many cores does your machine have?
- (c) What is the frequency of each processor?
- (d) How much physical memory does your system have?
- (e) How much of this memory is free?
- (f) What is total number of number of forks since the boot in the system?
- (g) How many context switches has the system performed since bootup?

```
winter@windows: ~  
winter@windows:~/scripts$ cd  
winter@windows:~$ cat proc/cpuinfo  
cat: proc/cpuinfo: No such file or directory  
winter@windows:~$ cat /proc/cpuinfo  
processor       : 0  
vendor_id      : GenuineIntel  
cpu family     : 6  
model          : 140  
model name     : 11th Gen Intel(R) Core(TM) i5-1135G7 @ 2.40GHz  
stepping       : 1  
cpu MHz        : 2419.202  
cache size     : 8192 KB  
physical id    : 0  
siblings       : 3  
core id        : 0  
cpu cores      : 3  
apicid         : 0  
initial apicid : 0  
fpu            : yes  
fpu_exception  : yes  
cpuid level    : 22  
wp             : yes  
flags          : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov  
pat pse36 clflush mmx fxsr sse sse2 ht syscall nx rdtscp lm constant_tsc rep_g  
ood nopl xtopology nonstop_tsc cpuid tsc_known_freq pni pclmulqdq ssse3 cx16 pc  
id sse4_1 sse4_2 x2apic movbe popcnt aes xsave avx rdrand hypervisor lahf_lm ab  
m 3dnowprefetch invpcid_single fsgsbase bmi1 avx2 bmi2 invpcid rdseed clflushop  
t md_clear flush_l1d arch_capabilities  
bugs           : spectre_v1 spectre_v2 spec_store_bypass swapgs
```

```
winter@windows: ~  
winter@windows:~$ cat /proc/meminfo  
MemTotal:      10506480 kB  
MemFree:       8361008 kB  
MemAvailable:  9353556 kB  
Buffers:       39712 kB  
Cached:        1178936 kB  
SwapCached:    0 kB  
Active:        599008 kB  
Inactive:      1285564 kB  
Active(anon):   1860 kB  
Inactive(anon): 694448 kB  
Active(file):   597148 kB  
Inactive(file): 591116 kB  
Unevictable:    0 kB  
Mlocked:        0 kB  
SwapTotal:     2097148 kB  
SwapFree:      2097148 kB  
Zswap:         0 kB  
Zswapped:       0 kB  
Dirty:         844 kB  
Writeback:      0 kB  
AnonPages:     665956 kB  
Mapped:        336036 kB  
Shmem:         35860 kB  
KReclaimable:  64516 kB  
Slab:          159092 kB  
SReclaimable:  64516 kB  
SUnreclaim:    94576 kB  
KernelStack:   7344 kB
```

3) Run strace along with the binary program of empty.c given in subdirectory strace . What do you think the output of strace indicates in this case? How many different system call functions do you see?

Next, use strace along with another binary program of hello.c (which is in the same directory).

Compare that is observe the two strace outputs

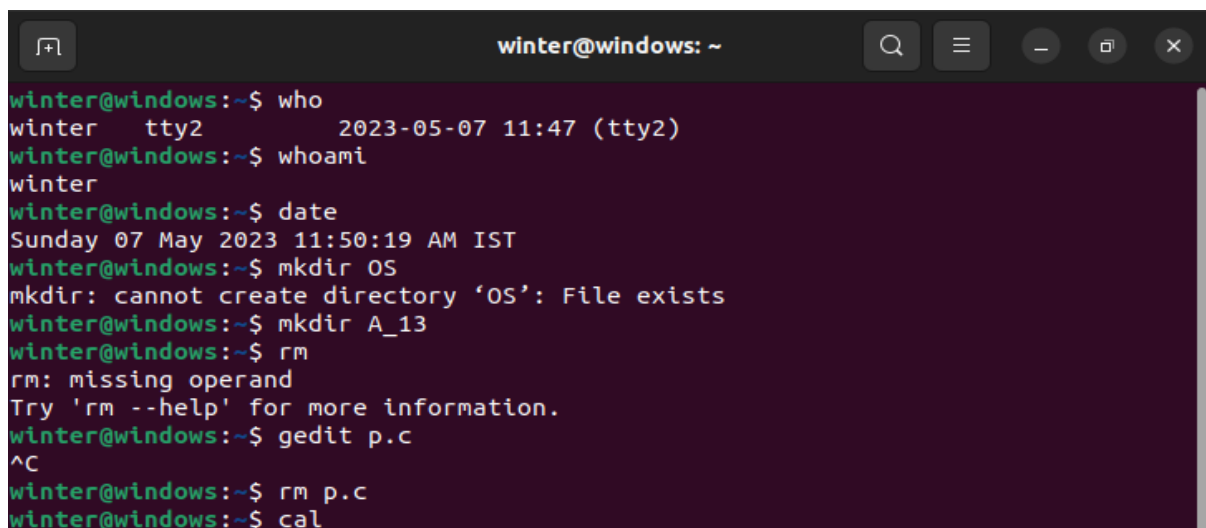
- Which part of the strace output is common, and which part has to do with the specific program?
- Observe the list of all unique system calls along with input and output parameters and study those system call ?

```
winter@windows:~/scripts$ strace -o hello-trace1 ./hello
Hello Worldwinter@windows:~/scripts$ strace -o hello-trace2 ./hello
Hello Worldwinter@windows:~/scripts$ diff hello-trace1 hello-trace2
1,4c1,4
< execve("./hello", [ "./hello" ], 0x7ffd8e25f590 /* 46 vars */) = 0
< brk(NULL)                                = 0x55725551f000
< arch_prctl(0x3001 /* ARCH_??? */, 0x7ffd11b93d20) = -1 EINVAL (Invalid argume
nt)
< mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x
7fe83935d000
---
> execve("./hello", [ "./hello" ], 0x7ffe81d28270 /* 46 vars */) = 0
> brk(NULL)                                = 0x55efc1d37000
> arch_prctl(0x3001 /* ARCH_??? */, 0x7ffe7e36f200) = -1 EINVAL (Invalid argume
nt)
> mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x
7f45aad8b000
8c8
< mmap(NULL, 64247, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7fe83934d000
---
> mmap(NULL, 64247, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f45aad7b000
17,21c17,21
< mmap(NULL, 2260560, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7fe839000
000
< mmap(0x7fe839028000, 1658880, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_
```

4) Execute following commands:

1. clear
2. cal
3. who&whoami
4. date
5. mkdir
6. rm
7. cat
8. cd
9. cp
10. grep
11. ls (execute all options)
12. mv
13. rm
14. rmdir

15. echo
16. uptime
17. uname, hostname
18. touch
19. cut
20. Head, tail
21. ps
22. Chmod
23. wc
24. chown
25. Man



```
winter@windows: ~  
winter@windows:~$ who  
winter  tty2      2023-05-07 11:47 (tty2)  
winter@windows:~$ whoami  
winter  
winter@windows:~$ date  
Sunday 07 May 2023 11:50:19 AM IST  
winter@windows:~$ mkdir OS  
mkdir: cannot create directory 'OS': File exists  
winter@windows:~$ mkdir A_13  
winter@windows:~$ rm  
rm: missing operand  
Try 'rm --help' for more information.  
winter@windows:~$ gedit p.c  
^C  
winter@windows:~$ rm p.c  
winter@windows:~$ cal
```

```
winter@windows: ~/OS
Unpacking ncal (12.1.7+nmu3ubuntu2) ...
Setting up ncal (12.1.7+nmu3ubuntu2) ...
Processing triggers for man-db (2.10.2-1) ...
winter@windows:~$ cal
      May 2023
Su Mo Tu We Th Fr Sa
      1  2  3  4  5  6
 7  8  9 10 11 12 13
14 15 16 17 18 19 20
21 22 23 24 25 26 27
28 29 30 31

winter@windows:~$ cd OS
winter@windows:~/OS$ gedit p2.c
^C
winter@windows:~/OS$ gcc p2.c
Command 'gcc' not found, but can be installed with:
sudo apt install gcc
winter@windows:~/OS$ sudo apt install gcc
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  binutils binutils-common binutils-x86-64-linux-gnu gcc-11 libasan6
  libbinutils libc-dev-bin libc-devtools libc6-dev libcc1-0 libcrypt-dev
  libctf-nobfd0 libctf0 libgcc-11-dev libitm1 liblsan0 libnsl-dev
  libtirpc-dev libubsan0 libubsan1 linux-libc-dev manpages-dev rpcsvc-proto
Suggested packages:
  binutils-doc gcc-multilib make autoconf automake libtool flex bison gcc-doc
winter@windows:~/OS$ cp p2.c practical.c
winter@windows:~/OS$ cd
winter@windows:~$ grep OS
^C
winter@windows:~$ ls
A_13  Documents  Music  Pictures  scripts  Templates
Desktop  Downloads  OS      Public    snap     Videos
winter@windows:~$ mv
mv: missing file operand
Try 'mv --help' for more information.
winter@windows:~$ cd OS
winter@windows:~/OS$ mv p2.c practical.c
winter@windows:~/OS$ cd
winter@windows:~$ rmdir A_13
winter@windows:~$ cd OS
winter@windows:~/OS$ cd p2.c
bash: cd: p2.c: No such file or directory
winter@windows:~/OS$ cd practical.c
bash: cd: practical.c: Not a directory
winter@windows:~/OS$ gcc practical.c
Command 'gcc' not found, but can be installed with:
sudo apt install gcc
winter@windows:~/OS$
```

```
winter@windows: ~/OS
winter@windows:~/OS$ cd
winter@windows:~$ grep OS
^C
winter@windows:~$ ls
A_13      Documents Music  Pictures scripts Templates
Desktop  Downloads OS     Public  snap    Videos
winter@windows:~$ mv
mv: missing file operand
Try 'mv --help' for more information.
winter@windows:~$ cd OS
winter@windows:~/OS$ mv p2.c practical.c
winter@windows:~/OS$ cd
winter@windows:~$ rmdir A_13
winter@windows:~$ cd OS
winter@windows:~/OS$ cd p2.c
bash: cd: p2.c: No such file or directory
winter@windows:~/OS$ cd practical.c
bash: cd: practical.c: Not a directory
winter@windows:~/OS$ gcc practical.c
Command 'gcc' not found, but can be installed with:
sudo apt install gcc
winter@windows:~/OS$ ps
  PID TTY          TIME CMD
  4018 pts/0        00:00:00 bash
  4641 pts/0        00:00:00 ps
winter@windows:~/OS$ wc
^C
winter@windows:~/OS$ man ls
winter@windows:~/OS$
```



```
winter@windows: ~/OS
LS(1) User Commands LS(1)

NAME
    ls - list directory contents

SYNOPSIS
    ls [OPTION]... [FILE]...

DESCRIPTION
    List information about the FILES (the current directory by default).
    Sort entries alphabetically if none of -cftuvSUX nor --sort is speci-
    fied.

    Mandatory arguments to long options are mandatory for short options
    too.

    -a, --all
        do not ignore entries starting with .

    -A, --almost-all
        do not list implied . and ..

    --author
        with -l, print the author of each file

    -b, --escape
        print C-style escapes for nongraphic characters

Manual page ls(1) line 1 (press h for help or q to quit)
```

CONCLUSION - Basic Linux System Commands and shell scripts to display the system particulars (processor, processes and memory) has been implemented