

TASK 5.3

Part 1:

1. How many states could have a process in Linux?

Process state: created, ready, terminated, running, waiting, admitted, interrupt

2. Examine the pstree command. Make output (highlight) the chain (ancestors) of the current process.

```
* Documentation: https://help.ubuntu.com/
root@CsnKhai:~# echo $$
884
root@CsnKhai:~# pstree -s 884
init--login--bash--pstree
root@CsnKhai:~#
```

3. What is a proc file system?

-- is a virtual file system that provides an interface to kernel data structures and information about processes and system resources. It is not a traditional file system that stores data on disk but rather a way for the kernel to expose various system and process-related information to userspace applications and users.

4. Print information about the processor (its type, supported technologies, etc.).

```
root@CsnKhai:~# lscpu
Architecture:        i686
CPU op-mode(s):      32-bit
Byte Order:          Little Endian
CPU(s):              1
On-line CPU(s) list: 0
Thread(s) per core:  1
Core(s) per socket:  1
Socket(s):           1
Vendor ID:           GenuineIntel
CPU family:          6
Model:               140
Stepping:            1
CPU MHz:             2419.332
BogoMIPS:            4838.66
L1d cache:           48K
L1i cache:           32K
L2 cache:            1280K
L3 cache:            8192K
root@CsnKhai:~#
```

5. Use the ps command to get information about the process. The information should be as follows: the owner of the process, the arguments with which the process was launched for execution, the group owner of this process, etc.

```
root      [kworker/u3:1]          root
root      [jbd2/sda1-8]          root
root      [ext4-rsv-conver]      root
root      upstart-udev-bridge --daemo root
root      /lib/systemd/systemd-udev root
message+ dbus-daemon --system --fork messagebus
root      /lib/systemd/systemd-logind root
syslog    rsyslogd              syslog
root      upstart-file-bridge --daemo root
root      upstart-socket-bridge --dae root
root      dhclient -1 -v -pf /run/dhc root
root      /sbin/getty -8 38400 tty4    root
root      /sbin/getty -8 38400 tty5    root
root      /sbin/getty -8 38400 tty2    root
root      /sbin/getty -8 38400 tty3    root
root      /sbin/getty -8 38400 tty6    root
root      /usr/sbin/sshd -D           root
root      cron                     root
root      /bin/login --              root
root      [kauditd]                root
root      -bash                     root
root      [kworker/u2:1]          root
root      [kworker/0:0]           root
root      [kworker/u2:0]          root
root      ps -eo user,args,group    root
```

6. How to define kernel processes and user processes?

```
root@CsnKhai:~# ps ef
PID TTY STAT TIME COMMAND
854 tty1 Ss 0:00 /bin/login -- PATH=/usr/local/sbin:/usr/local/bi
884 tty1 S 0:00 \_ -bash TERM=linux HOME=/root SHELL=/bin/bash USER=
940 tty1 R+ 0:00 \_ ps ef XDG_VTNR=1 XDG_SESSION_ID=c1 SHELL=/bin
761 tty6 Ss+ 0:00 /sbin/getty -8 38400 tty6 PATH=/usr/local/sbin:/usr/l
759 tty3 Ss+ 0:00 /sbin/getty -8 38400 tty3 PATH=/usr/local/sbin:/usr/l
758 tty2 Ss+ 0:00 /sbin/getty -8 38400 tty2 PATH=/usr/local/sbin:/usr/l
755 tty5 Ss+ 0:00 /sbin/getty -8 38400 tty5 PATH=/usr/local/sbin:/usr/l
753 tty4 Ss+ 0:00 /sbin/getty -8 38400 tty4 PATH=/usr/local/sbin:/usr/l
root@CsnKhai:~#

root@CsnKhai:~# ps aux | grep dd
root      2  0.0  0.0   0   0 ?        S   12:26   0:00 [kthreadd]
root     954  0.0  0.3  4676  820 tty1    S+  12:47   0:00 grep --color=au
to dd
root@CsnKhai:~#
```

7. Print the list of processes to the terminal. Briefly describe the statuses of the processes.

What condition are they in, or can they be arriving in?

Файл	Машина	Перегляд	Введення	Пристрої	Довідка					
root	68	0.0	0.0	0	0 ?	S<	12:26	0:00	[charger_manage	
root	114	0.0	0.0	0	0 ?	S<	12:26	0:00	[kpsmoused]	
root	116	0.0	0.0	0	0 ?	S	12:26	0:00	[scsi_eh_2]	
root	117	0.0	0.0	0	0 ?	S<	12:26	0:00	[kworker/u3:1]	
root	127	0.0	0.0	0	0 ?	S	12:26	0:00	[jbd2/sda1-8]	
root	128	0.0	0.0	0	0 ?	S<	12:26	0:00	[ext4-rsv-conve	
root	259	0.0	0.2	3008	620 ?	S	12:26	0:00	upstart-udev-br	
root	264	0.0	0.6	12052	1500 ?	Ss	12:26	0:00	/lib/systemd/sy	
message+	331	0.0	0.3	4236	988 ?	Ss	12:26	0:00	dbus-daemon --s	
root	361	0.0	0.6	4212	1720 ?	Ss	12:26	0:00	/lib/systemd/sy	
syslog	366	0.0	0.4	30476	1104 ?	Ssl	12:26	0:00	rsyslogd	
root	373	0.0	0.2	2880	596 ?	S	12:26	0:00	upstart-file-br	
root	521	0.0	0.2	2868	600 ?	S	12:26	0:00	upstart-socket-	
root	633	0.0	0.9	5512	2336 ?	Ss	12:26	0:00	dhclient -1 -v	
root	753	0.0	0.3	4644	832 tty4	Ss+	12:26	0:00	/sbin/getty -8	
root	755	0.0	0.3	4644	824 tty5	Ss+	12:26	0:00	/sbin/getty -8	
root	758	0.0	0.3	4644	832 tty2	Ss+	12:26	0:00	/sbin/getty -8	
root	759	0.0	0.3	4644	836 tty3	Ss+	12:26	0:00	/sbin/getty -8	
root	761	0.0	0.3	4644	832 tty6	Ss+	12:26	0:00	/sbin/getty -8	
root	787	0.0	1.0	7796	2480 ?	Ss	12:26	0:00	/usr/sbin/sshd	
root	792	0.0	0.3	3052	796 ?	Ss	12:26	0:00	cron	
root	854	0.0	0.8	4400	2016 tty1	Ss	12:26	0:00	/bin/login --	
root	867	0.0	0.0	0	0 ?	S	12:26	0:00	[kauditd]	
root	884	0.0	1.1	6520	2868 tty1	S	12:26	0:00	-bash	
root	908	0.0	0.0	0	0 ?	S	12:31	0:00	[kworker/u2:1]	
root	941	0.0	0.0	0	0 ?	S	12:46	0:00	[kworker/0:0]	
root	952	0.0	0.0	0	0 ?	S	12:47	0:00	[kworker/u2:0]	
root	976	0.0	0.0	0	0 ?	S	12:52	0:00	[kworker/u2:2]	
root	977	0.0	0.4	5216	1156 tty1	R+	12:54	0:00	ps aux	

S< (High-Priority Scheduling Class): This status indicates a process that is in the "idle" state and is using the "high-priority scheduling class." It's a special state that some processes might enter when they are waiting for CPU time and have been placed in the high-priority queue.

R+ (Running or Runnable): The "+" sign after "R" indicates that the process is in the foreground process group. This status means the process is currently running or ready to run on a CPU core.

Ssl (Interruptible Sleep - Scheduling): This status indicates a process that is in an interruptible sleep state. The "l" means that the process is multi-threaded and is using a lock. The process is waiting for a condition to be met and can be awakened by signals or other events.

Ss (Session Leader - Interruptible Sleep): This status indicates a session leader process that is in an interruptible sleep state. Session leaders are typically the first process in a session, and they manage terminal sessions. Like "Ssl," this process can be awakened by signals or other events.

8. Display only the processes of a specific user.

ps -u root command

```
 45 ?      00:00:00 scsi_eh_1
 67 ?      00:00:00 deferwq
 68 ?      00:00:00 charger_manager
114 ?      00:00:00 kpsmouse
116 ?      00:00:00 scsi_eh_2
117 ?      00:00:00 kworker/u3:1
127 ?      00:00:00 jbd2/sda1-8
128 ?      00:00:00 ext4-rsv-conver
259 ?      00:00:00 upstart-udev-br
264 ?      00:00:00 systemd-udev
361 ?      00:00:00 systemd-logind
373 ?      00:00:00 upstart-file-br
521 ?      00:00:00 upstart-socket-
633 ?      00:00:00 dhclient
753 tty4    00:00:00 getty
755 tty5    00:00:00 getty
758 tty2    00:00:00 getty
759 tty3    00:00:00 getty
761 tty6    00:00:00 getty
787 ?      00:00:00 sshd
792 ?      00:00:00 cron
854 tty1    00:00:00 login
867 ?      00:00:00 kauditd
884 tty1    00:00:00 bash
941 ?      00:00:00 kworker/0:0
952 ?      00:00:00 kworker/u2:0
976 ?      00:00:00 kworker/u2:2
988 ?      00:00:00 kworker/u2:1
990 tty1    00:00:00 ps
root@CsnKhai:~#
```

9. What utilities can be used to analyze existing running tasks (by analyzing the help for the ps command)?

ps aux: This command provides a detailed list of all running processes along with various information, including process IDs, statuses, resource usage, and more.

ps -ef: Similar to ps aux, this command displays a detailed list of all processes in a different format.

ps -e: This command displays a list of all processes, showing only their process IDs and statuses. It's a more compact view compared to ps aux.

10. What information does top command display?

Файл

Машина

Перегляд

Введення

Пристрої

Довідка

top - 13:07:18 up 41 min, 1 user, load average: 0.00, 0.00, 0.00

Tasks: 60 total, 1 running, 59 sleeping, 0 stopped, 0 zombie

%Cpu(s): 0.0 us, 0.0 sy, 0.0 ni, 100.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st

Mem: 247792 total, 101640 used, 146152 free, 11732 buffers

Swap: 0 total, 0 used, 0 free, 67188 cached Mem

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
976	root	20	0	0	0	0	S	0.3	0.0	0:00.17	kworker/u2
1	root	20	0	4200	2184	1392	S	0.0	0.9	0:00.70	init
2	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kthreadd
3	root	20	0	0	0	0	S	0.0	0.0	0:00.00	ksoftirqd/0
5	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	kworker/0:0
7	root	20	0	0	0	0	S	0.0	0.0	0:00.12	rcu_sched
8	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcu_bh
9	root	rt	0	0	0	0	S	0.0	0.0	0:00.00	migration/0
10	root	rt	0	0	0	0	S	0.0	0.0	0:00.02	watchdog/0
11	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	khelper
12	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kdevtmpfs
13	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	netns
14	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	writeback
15	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	kintegrityd
16	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	bioaset
17	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	kworker/u3
18	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	kblockd
19	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	ata_sff
20	root	20	0	0	0	0	S	0.0	0.0	0:00.26	khubd
21	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	md
22	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	devfreq_wq
23	root	20	0	0	0	0	S	0.0	0.0	0:00.82	kworker/0:1
25	root	20	0	0	0	0	S	0.0	0.0	0:00.00	khungtaskd

A convenient tool to manage processes is top.

For common process management tasks, top is so great because it gives an overview of the most active processes currently running (hence the name top). This enables you to easily find processes that might need attention. From top, you can also perform common process management tasks, such as adjusting the current process priority and killing processes.

Among the information that you can conveniently obtain from the top utility is the process state.

Here is the information that the «top» command typically displays: Overall System Information, Global Resource Usage, Process List, Interactive Features.

12. Display the processes of the specific user using the top command.

```
top - 13:12:34 up 46 min, 1 user, load average: 0.00, 0.00, 0.00
Tasks: 60 total, 1 running, 59 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.3 us, 0.0 sy, 0.0 ni, 99.7 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem: 247792 total, 101640 used, 146152 free, 11748 buffers
KiB Swap: 0 total, 0 used, 0 free. 67192 cached Mem
Which user (blank for all) root
  PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM    TIME+  COMMAND
    1 root        20   0     4200    2184    1392 S   0.0   0.9   0:00.70  init
    2 root        20   0         0         0         0 S   0.0   0.0   0:00.00  kthreadd
    3 root        20   0         0         0         0 S   0.0   0.0   0:00.00  ksoftirqd/0
    5 root         0 -20         0         0         0 S   0.0   0.0   0:00.00  kworker/0:0
    7 root        20   0         0         0         0 S   0.0   0.0   0:00.14  rcu_sched
    8 root        20   0         0         0         0 S   0.0   0.0   0:00.00  rcu_bh
    9 root        rt    0         0         0         0 S   0.0   0.0   0:00.00  migration/0
   10 root        rt    0         0         0         0 S   0.0   0.0   0:00.02  watchdog/0
   11 root         0 -20         0         0         0 S   0.0   0.0   0:00.00  khelper
   12 root        20   0         0         0         0 S   0.0   0.0   0:00.00  kdevtmpfs
   13 root         0 -20         0         0         0 S   0.0   0.0   0:00.00  netns
   14 root         0 -20         0         0         0 S   0.0   0.0   0:00.00  writeback
   15 root         0 -20         0         0         0 S   0.0   0.0   0:00.00  kintegrityd
   16 root         0 -20         0         0         0 S   0.0   0.0   0:00.00  bioset
   17 root         0 -20         0         0         0 S   0.0   0.0   0:00.00  kworker/u3:0
   18 root         0 -20         0         0         0 S   0.0   0.0   0:00.00  kblockd
   19 root         0 -20         0         0         0 S   0.0   0.0   0:00.00  ata_sff
   20 root        20   0         0         0         0 S   0.0   0.0   0:00.26  khubd
   21 root         0 -20         0         0         0 S   0.0   0.0   0:00.00  md
   22 root         0 -20         0         0         0 S   0.0   0.0   0:00.00  devfreq_wq
   23 root        20   0         0         0         0 S   0.0   0.0   0:00.94  kworker/0:1
   25 root        20   0         0         0         0 S   0.0   0.0   0:00.00  khungtaskd
   26 root        20   0         0         0         0 S   0.0   0.0   0:00.00  kswapd0
```

12. What interactive commands can be used to control the top command? Give a couple of examples.

q: Quit top.

Example: Press q to exit top.

u: Enter "User filter" mode to display processes for a specific user.

Example: Press u, enter the username, and press Enter to filter processes for that user.

k: Kill a process. You'll be prompted to enter the PID of the process you want to kill.

Example: Press k, enter the PID of the process, and press Enter.

r: Renice a process to change its priority.

Example: Press r, enter the PID of the process, enter the new nice value, and press Enter.

i: Toggle idle-only tasks display. When enabled, only tasks that are currently using CPU time are displayed.

Example: Press i to toggle idle-only tasks display.

13. Sort the contents of the processes window using various parameters (for example, the amount of processor time taken up, etc.)

Sort by CPU Usage:

```
top - 13:16:13 up 50 min, 1 user, load average: 0.00, 0.00, 0.00
Tasks: 60 total, 1 running, 59 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.3 sy, 0.0 ni, 99.3 id, 0.3 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem: 247792 total, 101660 used, 146132 free, 11760 buffers
KiB Swap: 0 total, 0 used, 0 free. 67192 cached Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1038	root	20	0	5420	1304	988	R	0.3	0.5	0:00.05	top
1	root	20	0	4200	2184	1392	S	0.0	0.9	0:00.70	init
2	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kthreadd
3	root	20	0	0	0	0	S	0.0	0.0	0:00.00	ksoftirqd/0
5	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	kworker/0:0
7	root	20	0	0	0	0	S	0.0	0.0	0:00.14	rcu_sched
8	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcu_bh
9	root	rt	0	0	0	0	S	0.0	0.0	0:00.00	migration/0
10	root	rt	0	0	0	0	S	0.0	0.0	0:00.02	watchdog/0
11	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	khelper
12	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kdevtmpfs
13	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	netns
14	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	writeback
15	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	kintegrity
16	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	bioaset
17	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	kworker/u3
18	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	kblockd
19	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	ata_sff
20	root	20	0	0	0	0	S	0.0	0.0	0:00.26	khudb
21	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	md
22	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	devfreq_wq
23	root	20	0	0	0	0	S	0.0	0.0	0:01.00	kworker/0:1
25	root	20	0	0	0	0	S	0.0	0.0	0:00.00	khungtaskd

Sort by Process ID:

```
top - 13:17:47 up 51 min, 1 user, load average: 0.00, 0.00, 0.00
Tasks: 60 total, 1 running, 59 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.0 sy, 0.0 ni, 100.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem: 247792 total, 101900 used, 145892 free, 11776 buffers
KiB Swap: 0 total, 0 used, 0 free. 67200 cached Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1038	root	20	0	5420	1304	988	R	0.0	0.5	0:00.12	top
1037	root	20	0	0	0	0	S	0.0	0.0	0:00.03	kworker/u2:2
1025	root	20	0	0	0	0	S	0.0	0.0	0:00.03	kworker/u2:1
1001	root	20	0	0	0	0	S	0.0	0.0	0:00.12	kworker/u2:0
941	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kworker/0:0
884	root	20	0	6524	2872	1648	S	0.0	1.2	0:00.05	bash
867	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kauditd
854	root	20	0	4400	2016	1536	S	0.0	0.8	0:00.02	login
792	root	20	0	3052	796	624	S	0.0	0.3	0:00.00	cron
787	root	20	0	7796	2480	1988	S	0.0	1.0	0:00.00	sshd
761	root	20	0	4644	832	716	S	0.0	0.3	0:00.00	getty
759	root	20	0	4644	836	716	S	0.0	0.3	0:00.00	getty
758	root	20	0	4644	832	716	S	0.0	0.3	0:00.00	getty
755	root	20	0	4644	824	716	S	0.0	0.3	0:00.00	getty
753	root	20	0	4644	832	716	S	0.0	0.3	0:00.00	getty
633	root	20	0	5512	2336	616	S	0.0	0.9	0:00.00	dhclient
521	root	20	0	2868	600	432	S	0.0	0.2	0:00.01	upstart-soc+
373	root	20	0	2880	596	372	S	0.0	0.2	0:00.02	upstart-fil+
366	syslog	20	0	30476	1104	824	S	0.0	0.4	0:00.00	rsyslogd
361	root	20	0	4212	1720	1412	S	0.0	0.7	0:00.00	systemd-log+
331	message+	20	0	4236	988	708	S	0.0	0.4	0:00.02	dbus-daemon
264	root	20	0	12052	1500	972	S	0.0	0.6	0:00.01	systemd-ude+
259	root	20	0	3008	620	472	S	0.0	0.3	0:00.05	upstart-ude+

14. Concept of priority, what commands are used to set priority?

«nice» command, «renice» command

15. Can I change the priority of a process using the top command? If so, how?

We need to use press the «r» key. Enter the PID process whose priority we want to change and press Enter. Then we need to enter the new priority value.

16. Examine the kill command. How to send with the kill command

process control signal? Give an example of commonly used signals.

Sometimes the kill command does not work because the process you want to kill is busy. In that case, you can use kill -9 to send the SIGKILL signal to the process. Because the SIGKILL signal cannot be ignored, it forces the process to stop, but you also risk losing data while using this command.

SIGTERM (15): This is the default signal sent by the kill command. It requests that the process terminate gracefully, allowing it to clean up resources before exiting.

17. Commands jobs, fg, bg, nohup. What are they for? Use the sleep, yes command to demonstrate the process control mechanism with fg, bg.

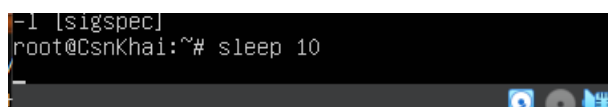
They allow you to manage the execution of processes in the foreground, background, and even after you log out of a terminal session.

jobs: This command is used to list the current jobs that are running or suspended in the background. Each job is assigned a job number.

fg: The fg command is used to bring a job to the foreground, allowing it to receive input and display output on the terminal. You can specify the job number to bring a specific job to the foreground.

bg: The bg command is used to resume a suspended job in the background. This allows a job to continue running even if you switch to another task or log out.

nohup: The nohup command is used to run a command immune to hangups (i.e., it will keep running even if the terminal session is disconnected or closed). It's often used in combination with the & symbol to run a command in the background.



```
-l [sigspec]
root@CsnKhai:~# sleep 10
```

A terminal window screenshot showing a shell prompt and the command 'sleep 10' being executed. The terminal title bar shows '-l [sigspec]' and the prompt is 'root@CsnKhai:~#'. The command 'sleep 10' is entered on the next line. The terminal window has a dark background and a light-colored text. There are some icons visible in the bottom right corner of the terminal window.

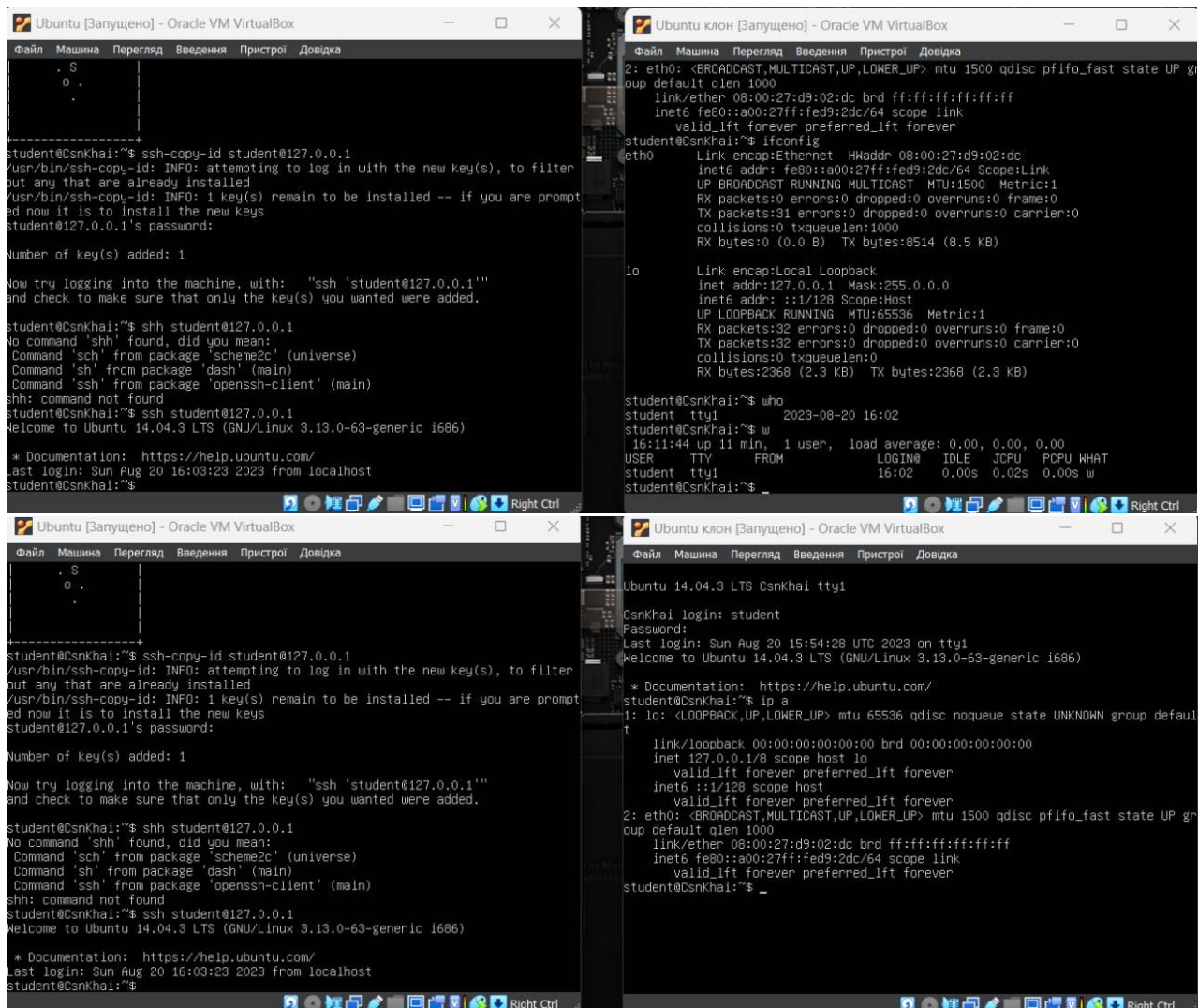

```
root@CsnKhai:~# sleep 10
^Z
[1]+  Stopped                  sleep 10
root@CsnKhai:~# bg
[1]+  sleep 10 &
root@CsnKhai:~# fg
-bash: fg: job has terminated
[1]+  Done                    sleep 10
root@CsnKhai:~#
```

Part 2:

1. Check the implementability of the most frequently used OPENSCH commands in the MS Windows operating system. (Description of the expected result of the commands + screenshots: command – result should be presented)

Implement basic SSH settings to increase the security of the client-server connection

Implement port forwarding for the SSH client from the host machine to the guest Linux virtual machine behind NAT.



```
Ubuntu [Запущено] - Oracle VM VirtualBox
Файл Машина Перегляд Введення Пристрої Довідка
Enter same passphrase again:
Your identification has been saved in /home/student/.ssh/id_rsa.
Your public key has been saved in /home/student/.ssh/id_rsa.pub.
The key fingerprint is:
52:be:44:81:65:4f:63:f8:e5:00:1d:dd:55:04:86:a3 student@CsnKhai
The key's randomart image is:
+--[ RSA 2048 ]-----+
  o=+o 0o+=
  ...=0...
  o..= .
  + E .
  . S
  o .
  .
+-----+
student@CsnKhai:~$ ssh-copy-id student@127.0.0.1
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter
out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are promp
ed now it is to install the new keys
student@127.0.0.1's password:

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'student@127.0.0.1'"
and check to make sure that only the key(s) you wanted were added.

student@CsnKhai:~$
```

```
Ubuntu клон [Запущено] - Oracle VM VirtualBox
Файл Машина Перегляд Введення Пристрої Довідка
Ubuntu 14.04.3 LTS CsnKhai tty1
CsnKhai login: student
Password:
Last login: Sun Aug 20 15:54:28 UTC 2023 on tty1
Welcome to Ubuntu 14.04.3 LTS (GNU/Linux 3.13.0-63-generic i686)

* Documentation: https://help.ubuntu.com/
student@CsnKhai:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group defaul
t
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP gr
oup default qlen 1000
    link/ether 08:00:27:d9:02:dc brd ff:ff:ff:ff:ff:ff
    inet6 fe80::a00:27ff:fed9:2dc/64 scope link
        valid_lft forever preferred_lft forever
student@CsnKhai:~$
```

```
Ubuntu [Запущено] - Oracle VM VirtualBox
Файл Машина Перегляд Введення Пристрої Довідка
Warning: Permanently added '127.0.0.1' (ECDSA) to the list of known hosts.
student@127.0.0.1's password:
Permission denied, please try again.
student@127.0.0.1's password:
Welcome to Ubuntu 14.04.3 LTS (GNU/Linux 3.13.0-63-generic i686)

* Documentation: https://help.ubuntu.com/
Last login: Sun Aug 20 15:57:27 2023 from localhost
student@CsnKhai:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/student/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/student/.ssh/id_rsa.
Your public key has been saved in /home/student/.ssh/id_rsa.pub.
The key fingerprint is:
52:be:44:81:65:4f:63:f8:e5:00:1d:dd:55:04:86:a3 student@CsnKhai
The key's randomart image is:
+--[ RSA 2048 ]-----+
  o=+o 0o+=
  ...=0...
  o..= .
  + E .
  . S
  o .
  .
+-----+
student@CsnKhai:~$
```

```
Ubuntu клон [Запущено] - Oracle VM VirtualBox
Файл Машина Перегляд Введення Пристрої Довідка
Ubuntu 14.04.3 LTS CsnKhai tty1
CsnKhai login: student
Password:
Last login: Sun Aug 20 15:54:28 UTC 2023 on tty1
Welcome to Ubuntu 14.04.3 LTS (GNU/Linux 3.13.0-63-generic i686)

* Documentation: https://help.ubuntu.com/
student@CsnKhai:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group defaul
t
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP gr
oup default qlen 1000
    link/ether 08:00:27:d9:02:dc brd ff:ff:ff:ff:ff:ff
    inet6 fe80::a00:27ff:fed9:2dc/64 scope link
        valid_lft forever preferred_lft forever
student@CsnKhai:~$
```

```
Ubuntu [Запущено] - Oracle VM VirtualBox
Файл Машина Перегляд Введення Пристрої Довідка
link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
inet 127.0.0.1/8 scope host lo
    valid_lft forever preferred_lft forever
inet6 ::1/128 scope host
    valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP gr
oup default qlen 1000
    link/ether 08:00:27:ed:14:1d brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.4/24 brd 10.0.2.255 scope global eth0
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:feed:141d/64 scope link
        valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen
1000
    link/ether 08:00:27:6d:16:d3 brd ff:ff:ff:ff:ff:ff
root@CsnKhai:~# ssh student@127.0.0.1/8
ssh: Could not resolve hostname 127.0.0.1/8: Name or service not known
root@CsnKhai:~# ssh student@127.0.0.1
The authenticity of host '127.0.0.1 (127.0.0.1)' can't be established.
ECDSA key fingerprint is d6:eb:2b:a9:bd:63:86:af:31:2f:bb:01:b5:14:63:ab.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '127.0.0.1' (ECDSA) to the list of known hosts.
student@127.0.0.1's password:
Permission denied, please try again.
student@127.0.0.1's password:
Welcome to Ubuntu 14.04.3 LTS (GNU/Linux 3.13.0-63-generic i686)

* Documentation: https://help.ubuntu.com/
Last login: Sun Aug 20 15:57:27 2023 from localhost
student@CsnKhai:~$
```

```
Ubuntu клон [Запущено] - Oracle VM VirtualBox
Файл Машина Перегляд Введення Пристрої Довідка
Ubuntu 14.04.3 LTS CsnKhai tty1
CsnKhai login: student
Password:
Last login: Sun Aug 20 15:54:28 UTC 2023 on tty1
Welcome to Ubuntu 14.04.3 LTS (GNU/Linux 3.13.0-63-generic i686)

* Documentation: https://help.ubuntu.com/
student@CsnKhai:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group defa
t
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
oup default qlen 1000
    link/ether 08:00:27:d9:02:dc brd ff:ff:ff:ff:ff:ff
    inet6 fe80::a00:27ff:fed9:2dc/64 scope link
        valid_lft forever preferred_lft forever
student@CsnKhai:~$
```

```
Командный рядок
SHA256:60asnTf8L8+wwCtocyXPNVDgkqfH2HYfLsJK84RvHvA user@DESKTOP
-B2MSQ0D
The key's randomart image is:
+---[RSA 3072]-----+
|      .o.      |
|      o.      |
|     o.o.     |
|    .B o      |
|   .S+=....   |
|  o0+. o.    |
| +=E + o      |
| ooB**.*      |
| =o+=+++++   |
+---[SHA256]-----+

C:\Users\User>ssh-copy-id student@127.0.0.1
'ssh-copy-id' is not recognized as an internal or external comm
and,
operable program or batch file.

C:\Users\User>ssh student@127.0.0.1
ssh: connect to host 127.0.0.1 port 22: Connection refused

C:\Users\User>ssh student@127.0.0.1
ssh: connect to host 127.0.0.1 port 22: Connection refused

C:\Users\User>

Ubuntu клон [Запущено] - Oracle VM VirtualBox
Файл  Машина  Перегляд  Введення  Пристрої  Довідка
UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:31 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:0 (0.0 B)  TX bytes:8514 (8.5 KB)

lo
Link encap:Local Loopback
inet addr:127.0.0.1  Mask:255.0.0.0
inet6 addr: ::1/128 Scope:Host
UP LOOPBACK RUNNING  MTU:65536  Metric:1
RX packets:32 errors:0 dropped:0 overruns:0 frame:0
TX packets:32 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:2368 (2.3 KB)  TX bytes:2368 (2.3 KB)

student@CsnKhai:~$ who
student  tty1          2023-08-20 16:02
student@CsnKhai:~$ w
 16:11:44 up 11 min,  1 user,  load average: 0.00, 0.00, 0.00
USER      TTY      FROM            LOGIN@   IDLE   JCPU   PCPU   WHAT
student  tty1          16:02            0.00s   0.02s   0.00s  w
student@CsnKhai:~$ w
 16:20:48 up 20 min,  1 user,  load average: 0.00, 0.00, 0.00
USER      TTY      FROM            LOGIN@   IDLE   JCPU   PCPU   WHAT
student  tty1          16:02            0.00s   0.02s   0.00s  w
student@CsnKhai:~$ w
 16:21:44 up 21 min,  1 user,  load average: 0.00, 0.00, 0.00
USER      TTY      FROM            LOGIN@   IDLE   JCPU   PCPU   WHAT
student  tty1          16:02            1.00s   0.02s   0.00s  w
student@CsnKhai:~$
```

3. List the options for choosing keys for encryption in SSH. Implement 3 of them.

1) RSA (Rivest-Shamir-Adleman):

RSA is one of the most widely used key types in SSH. It provides strong security and is well-supported by SSH clients and servers.

2) ECDSA (Elliptic Curve Digital Signature Algorithm):

ECDSA is a modern and efficient alternative to RSA. It uses elliptic curve cryptography to provide strong security with smaller key sizes.

3) Ed25519:

Ed25519 is another modern key type that offers high security and performance. It's designed to be resistant to various cryptographic attacks