

Міністерство освіти і науки України
Одеський національний політехнічний університет
Інститут комп'ютерних систем
Кафедра інформаційних систем

Лабораторна робота №7
З дисципліни "Операційні системи"
Тема : «Команди управління процесами в ОС Unix»

Виконав
Ст.гр. АІ-204
Костецький Б.В.
Перевірили:
Блажко О.А.

Одеса 2021

Мета роботи: отримання навичок в управлінні процесами в ОС Unix засобами командної оболонки.

Завдання до виконання

1. Моделювання багатозадачності

Нехай оперативна пам'ять на комп'ютері-сервері становить N Гб. Системні процеси ОС займають до M Гбайт пам'яті, а кожна програма користувача може використовувати до K Гбайт пам'яті. Нехай в середньому процеси програм користувачів витрачають $F\%$ свого часу на очікування завершення вводу/виводу. Визначте середню завантаженість процесора, використовуючи значення з таблиці 3 варіантів.

№ команди	№ учасника команди	Пам'ять, N Гб	ОС, M Гб	Програма, K Гб	Очікування завершення вводу/виводу, F
7	1	9	1	0.5	50

2. Перегляд таблиці процесів

1. Отримайте ієрархію всіх процесів із зазначенням імен користувачів, їх запустили.
2. Отримайте ієрархію процесів, запущених від імені вашого профілю і з зазначенням PID цих процесів.
3. Отримайте список процесів, запущених в поточному терміналі, зі спрощеним набором колонок виведення таблиці процесів.
4. Отримайте список процесів, запущених від імені вашого користувача, з розширеним набором колонок таблиці процесів.
5. Отримайте список процесів, запущених від імені вашого користувача із зазначенням наступного набору колонок: PID, TTY, PPID, STAT, NI, CMD
6. Отримайте список всіх сплячих процесів зі спрощеним набором колонок виведення таблиці процесів.
7. Отримайте список процесів, відсортованих по PID, і визначте:
 - а. загальна кількість запущених процесів;
 - б. кількість процесів, які виконуються;
 - с. кількість сплячих процесів.
8. Отримайте список процесів, відсортованих за % використання процесора.

3. Керування станами процесів

1. У поточному терміналі виконайте команду `ping localhost`, але не завершуйте її роботу.
2. Запустіть другий термінал доступу до Linux-сервера.
3. У другому терміналі для команди `ping` отримаєте таблицю її процесу (колоники PID, STAT, CMD).
4. У другому терміналі призупиніть виконання процесу команди `ping`
5. У першому терміналі отримайте список фонових процесів
6. У другому терміналі відновіть виконання припиненого процесу
7. У другому терміналі зупиніть виконання процесу команди `ping`
8. У першому терміналі запустіть команду `ping` в фоновому режимі так, щоб він не був автоматично зупинений навіть після закриття терміналу, з якого був запущений.
9. Закрийте перший термінал.
10. У другому терміналі для команди `ping` отримаєте таблицю її процесу (колоники PID, STAT, CMD). Зробіть висновок про стан процесу.
11. Завершіть роботу процесу.

4. Управління пріоритетами процесів

1. Створіть bash-програму, що виконує операцію циклічного складання за формулою: $x = x + n$, де початкове значення x = кількість букв вашого прізвища, n - кількість букв у вашому імені. Ім'я програми збігається з транслітерацією вашого прізвища з розширенням `.sh`, наприклад, `ivanov.sh`
2. Запустіть bash-програму у фоновому режимі.
3. Перегляньте таблицю процесів для запущеного процесу, пов'язаного з bash-програмою, з урахуванням набором колонок: PID, PPID, STAT, NI, %CPU, CMD.
4. Виконайте команду призупинення запущеного процесу.
5. Ще раз перегляньте таблицю процесів для призупиненого процесу з урахуванням набором колонок: PID, PPID, STAT, NI, %CPU, CMD. Зробіть висновки про його стан.
6. Виконайте команду продовження виконання припиненого процесу.
7. Ще раз перегляньте таблицю процесів для процесу, який продовжив виконуватися, з урахуванням набором колонок: PID, PPID, STAT, NI, %CPU, CMD. Зробіть висновки про його стан.
8. Створіть два файли як символічні посилання на створену bash-програму з іменами як ім'я поточного файлу з додаванням цифр 2 і 3, відповідно, наприклад: `ivanov2.sh`, `ivanov3.sh`
9. Запустіть два файли у фоновому режимі.
10. Перегляньте таблицю процесів для трьох запущених процесів з урахуванням набором колонок: PID, PPID, STAT, NI, %CPU, CMD. Зробіть висновки за поточними значеннями NI та %CPU.

11. Зменшити пріоритет виконання одного з трьох процесів.
12. Перегляньте таблицю процесів для трьох запущених файлів з урахуванням набором колонок: PID, PPID, STAT, NI, %CPU, CMD. Зробіть висновки щодо змін значень % CPU для кожного процесу: як вони змінилися?

Хід роботи

1. Моделювання багатозадачності

Нехай оперативна пам'ять на комп'ютері-сервері становить N Гб. Системні процеси ОС займають до M Гбайт пам'яті, а кожна програма користувача може використовувати до K Гбайт пам'яті. Нехай в середньому процеси програм користувачів витрачають $F\%$ свого часу на очікування завершення вводу/виводу. Визначте середню завантаженість процесора, використовуючи значення з таблиці 3 варіантів.

№ команди	№ учасника команди	Пам'ять, N Гб	ОС, M Гб	Програма, K Гб	Очікування завершення вводу/виводу, F
7	1	9	1	0.5	50

Час задіяння процесора обчислюється за формулою: $1 - p^n$.

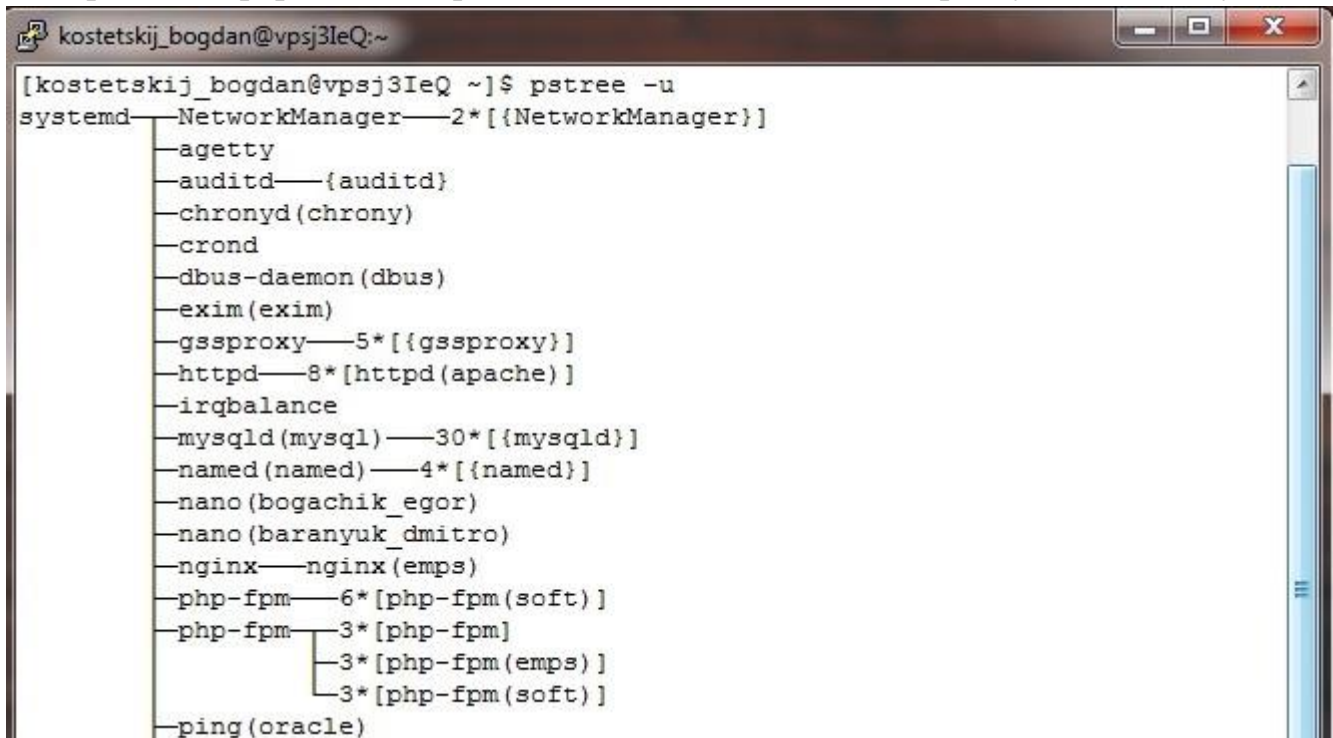
p – очікування завершення операцій вводу/виводу, n – кількість процесів.

$$N : (9-1) : 0,5 = 16$$

$$\text{Середня завантаженість процесора: } 1 - 0,5^{16} = 0,99$$

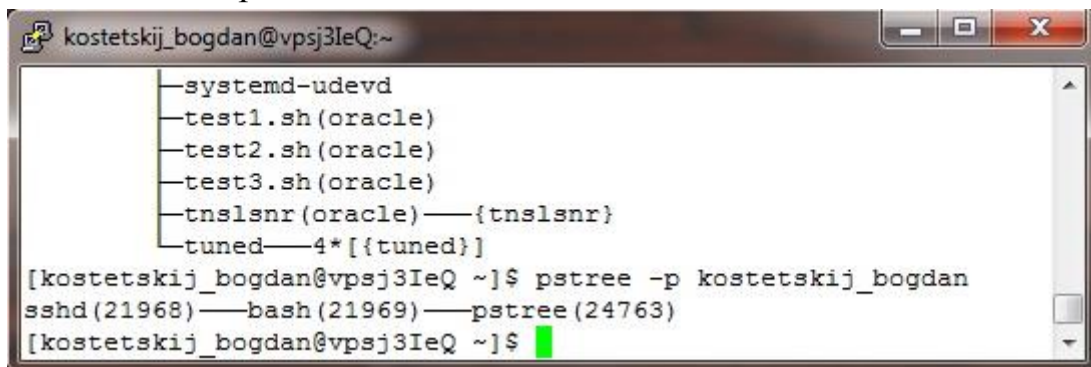
2. Перегляд таблиці процесів

1. Отримали ієрархію всіх процесів із зазначенням імен користувачів, їх запустили



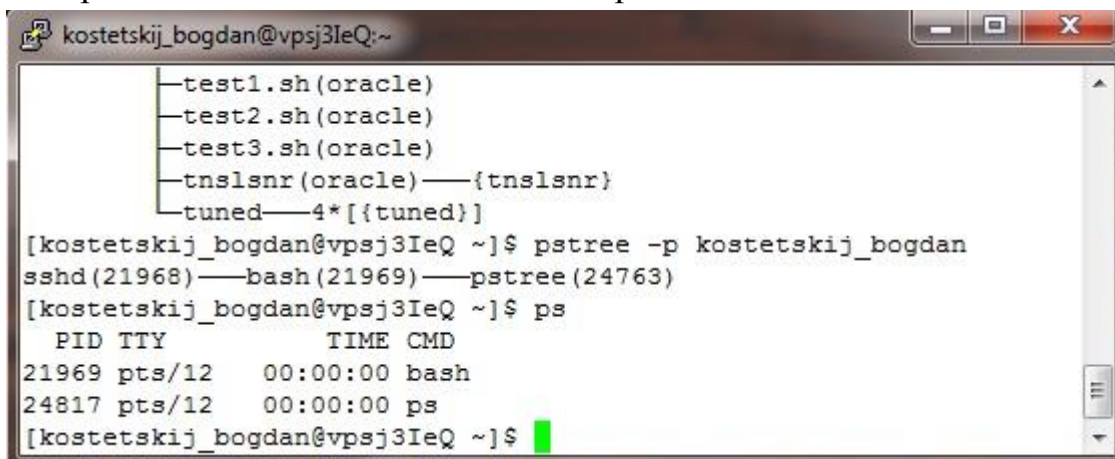
```
kostetskij_bogdan@vpsj3IeQ:~  
[kostetskij_bogdan@vpsj3IeQ ~]$ pstree -u  
systemd--NetworkManager--2*[{NetworkManager}]  
--agetty  
--auditd--{auditd}  
--chronyd(chrony)  
--crond  
--dbus-daemon(dbus)  
--exim(exim)  
--gssproxy--5*[{gssproxy}]  
--httpd--8*[{httpd(apache)}]  
--irqbalance  
--mysqld(mysql) --30*[{mysqld}]  
--named(named) --4*[{named}]  
--nano(bogachik_egor)  
--nano(baranyuk_dmitro)  
--nginx--nginx(emps)  
--php-fpm--6*[{php-fpm(soft)}]  
--php-fpm--3*[{php-fpm}]  
--php-fpm--3*[{php-fpm(emps)}]  
--php-fpm--3*[{php-fpm(soft)}]  
--ping(oracle)
```

2. Отримайли ієрархію процесів, запущених від імені вашого профілю і з зазначенням PID цих процесів.



```
kostetskij_bogdan@vpsj3IeQ:~  
--systemd-udevd  
--test1.sh(oracle)  
--test2.sh(oracle)  
--test3.sh(oracle)  
--tnslsnr(oracle)---{tnslsnr}  
--tuned--4*[{tuned}]  
[kostetskij_bogdan@vpsj3IeQ ~]$ pstree -p kostetskij_bogdan  
sshd(21968)---bash(21969)---pstree(24763)  
[kostetskij_bogdan@vpsj3IeQ ~]$
```

3. Отримали список процесів, запущених в поточному терміналі, зі спрощеним набором колонок виведення таблиці процесів.



```
kostetskij_bogdan@vpsj3IeQ:~  
--test1.sh(oracle)  
--test2.sh(oracle)  
--test3.sh(oracle)  
--tnslsnr(oracle)---{tnslsnr}  
--tuned--4*[{tuned}]  
[kostetskij_bogdan@vpsj3IeQ ~]$ pstree -p kostetskij_bogdan  
sshd(21968)---bash(21969)---pstree(24763)  
[kostetskij_bogdan@vpsj3IeQ ~]$ ps  
  PID TTY          TIME CMD  
 21969 pts/12    00:00:00 bash  
 24817 pts/12    00:00:00 ps  
[kostetskij_bogdan@vpsj3IeQ ~]$
```

4. Отримали список процесів, запущених від імені вашого користувача, з розширеним набором колонок таблиці процесів.

```
kostetskij_bogdan@vpsj3IeQ:~  
sshd(21968)---bash(21969)---pstree(24763)  
[kostetskij_bogdan@vpsj3IeQ ~]$ ps  
  PID TTY          TIME CMD  
21969 pts/12    00:00:00 bash  
24817 pts/12    00:00:00 ps  
[kostetskij_bogdan@vpsj3IeQ ~]$ ps -F -u kostetskij_bogdan  
UID          PID  PPID  C   SZ   RSS  PSR  STIME  TTY          TIME CMD  
kostets+ 21968 21926  0 40939 2548   1 23:02 ?            00:00:00 sshd: kostetskij_bogdan@pts/12  
kostets+ 21969 21968  0 28887 2112   1 23:02 pts/12    00:00:00 -bash  
kostets+ 24929 21969  0 38869 1876   0 23:31 pts/12    00:00:00 ps -F -u kostetskij_bogdan  
[kostetskij_bogdan@vpsj3IeQ ~]$
```

5. Отримали список процесів, запущених від імені вашого користувача із зазначенням наступного набору колонок: PID, TTY, PPID, STAT, NI, CMD

```
kostetskij_bogdan@vpsj3IeQ:~  
kostets+ 21968 21926  0 40939 2548   1 23:02 ?            00:00:00 sshd: kostetskij_bogdan@pt  
kostets+ 21969 21968  0 28887 2112   1 23:02 pts/12    00:00:00 -bash  
kostets+ 24929 21969  0 38869 1876   0 23:31 pts/12    00:00:00 ps -F -u kostetskij_bogdan  
[kostetskij_bogdan@vpsj3IeQ ~]$ ps -u kostetskij_bogdan -o pid, tty, ppid, stat, ni, cmd  
  PID TT          PPID STAT  NI  CMD  
21968 ?            21926 S      0    sshd: kostetskij_bogdan@pts/12  
21969 pts/12        21968 Ss     0    -bash  
25080 pts/12        21969 R+     0    ps -u kostetskij_bogdan -o pid, tty, ppid, stat, ni, cmd  
[kostetskij_bogdan@vpsj3IeQ ~]$
```

6. Отримайте список всіх сплячих процесів зі спрощеним набором колонок виведення таблиці процесів.

```
kostetskij_bogdan@vpsj3IeQ:~  
[kostetskij_bogdan@vpsj3IeQ ~]$ ps -ax | awk '$3 == "S"'  
  2 ?          S      0:01 [kthreadd]  
  6 ?          S      0:24 [ksoftirqd/0]  
  7 ?          S      0:09 [migration/0]  
  8 ?          S      0:00 [rcu_bh]  
 11 ?          S      0:17 [watchdog/0]  
 12 ?          S      0:19 [watchdog/1]  
 13 ?          S      0:05 [migration/1]  
 14 ?          S     12:11 [ksoftirqd/1]  
 18 ?          S      0:00 [kdevtmpfs]  
 20 ?          S      0:01 [khungtaskd]  
 35 ?          S     18:50 [kswapd0]  
114 ?          S      1:21 [kauditd]  
256 ?          S      0:00 [scsi_eh_0]  
258 ?          S      0:00 [scsi_eh_1]  
273 ?          S      2:15 [jbd2/vda1-8]  
520 ?          S      0:05 /usr/sbin/chronyd
```


7. Отримайте список процесів, відсортованих по PID, і визначте:

а. загальна кількість запущених процесів; -168

б. кількість процесів, які виконуються; -11

с. кількість сплячих процесів. - 157

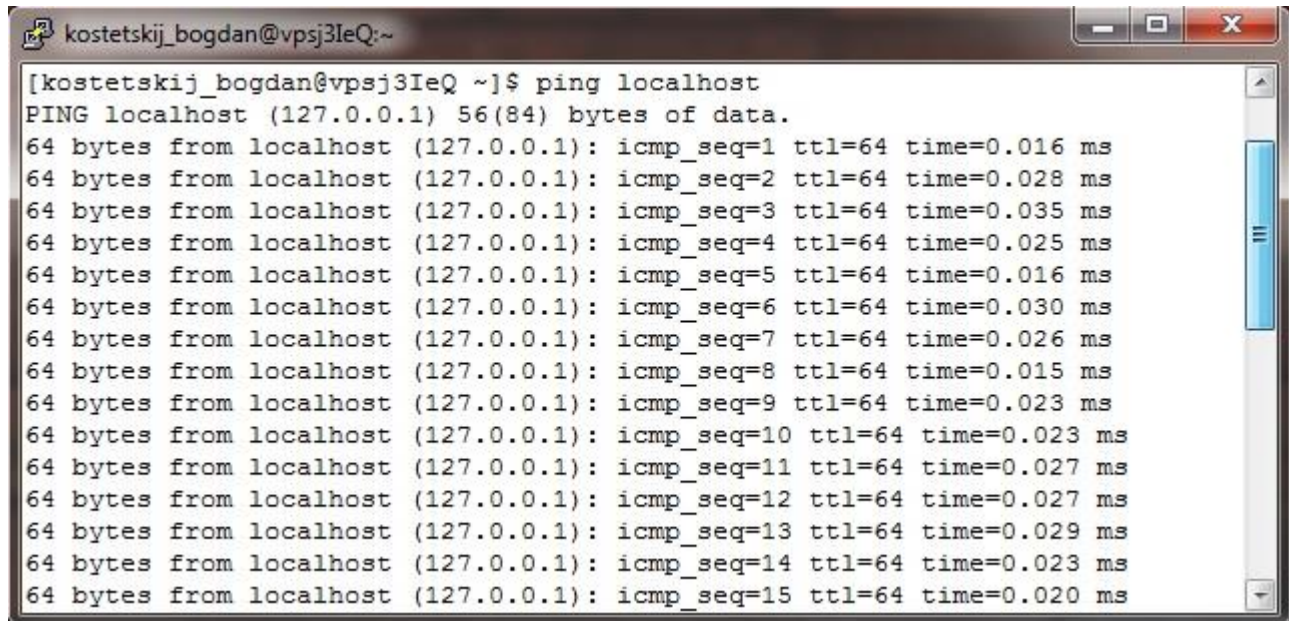
```
kostetskij_bogdan@vpsj3leQ:~  
top - 00:12:11 up 54 days, 7:15, 12 users, load average: 10.05, 8.51, 6.39  
Tasks: 168 total, 11 running, 157 sleeping, 0 stopped, 0 zombie  
%Cpu(s): 85.7 us, 13.1 sy, 1.2 ni, 0.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st  
PiB Mem : 0.0 total, 0.0 free, 0.0 used, 0.0 buff/cache  
PiB Swap: 0.0 total, 0.0 free, 0.0 used, 0.0 avail Mem  
scroll coordinates: y = 1/168 (tasks), x = 1/12 (fields)  
PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND  
29760 apache 20 0 110.8m 3.0m 1.5m S 0.0 0.2 0:00.03 httpd  
28885 apache 20 0 110.8m 3.0m 1.5m S 0.0 0.2 0:00.04 httpd  
27724 ozarchu+ 20 0 127.6m 1.6m 1.2m S 0.0 0.1 0:00.03 ping  
27717 belobro+ 20 0 110.8m 1.5m 1.1m S 1.0 0.1 0:39.77 belobrov.sh  
27573 soft 30 10 273.4m 5.1m 0.9m S 0.0 0.3 0:00.00 php-fpm  
27572 soft 30 10 273.4m 5.1m 0.9m S 0.0 0.3 0:00.00 php-fpm  
27571 soft 30 10 273.4m 5.1m 0.9m S 0.0 0.3 0:00.00 php-fpm  
27570 emps 30 10 273.4m 5.1m 0.9m S 0.0 0.3 0:00.00 php-fpm  
27569 emps 30 10 273.4m 5.1m 0.9m S 0.0 0.3 0:00.00 php-fpm  
27568 emps 30 10 273.4m 5.1m 0.9m S 0.0 0.3 0:00.00 php-fpm  
27567 root 30 10 273.4m 4.9m 0.8m S 0.0 0.3 0:00.00 php-fpm  
27566 root 30 10 273.4m 4.9m 0.8m S 0.0 0.3 0:00.00 php-fpm  
27564 root 30 10 273.4m 5.4m 1.2m S 0.0 0.3 0:03.45 php-fpm  
27557 apache 20 0 110.8m 3.0m 1.6m S 0.0 0.2 0:00.05 httpd  
27556 apache 20 0 110.8m 3.0m 1.6m S 0.0 0.2 0:00.05 httpd  
27549 emps 30 10 21.0m 2.1m 0.8m S 0.0 0.1 0:00.00 nginx  
27547 root 30 10 20.5m 1.2m 0.3m S 0.0 0.1 0:00.00 nginx  
27546 apache 20 0 110.8m 3.0m 1.6m S 0.0 0.2 0:00.05 httpd  
27541 apache 20 0 110.8m 3.0m 1.6m S 0.0 0.2 0:00.04 httpd  
27540 apache 20 0 110.8m 3.0m 1.6m S 0.0 0.2 0:00.04 httpd  
27237 root 20 0 0.0m 0.0m 0.0m S 0.0 0.0 0:00.00 kworker/1:3  
27021 ozarchu+ 20 0 112.8m 2.1m 1.6m S 0.0 0.1 0:00.16 bash  
27020 ozarchu+ 20 0 162.0m 2.4m 0.9m S 0.0 0.1 0:00.04 sshd  
27009 root 20 0 162.0m 6.5m 5.0m S 0.0 0.4 0:00.26 sshd  
26963 root 20 0 0.0m 0.0m 0.0m S 0.0 0.0 0:00.01 kworker/0:3
```

8. Отримайте список процесів, відсортованих за % використання процесора.

```
kostetskij_bogdan@vpsj3leQ:~  
top - 00:13:43 up 54 days, 7:16, 11 users, load average: 9.13, 8.62, 6.63  
Tasks: 165 total, 9 running, 154 sleeping, 1 stopped, 1 zombie  
%Cpu(s): 88.0 us, 10.8 sy, 1.2 ni, 0.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st  
PiB Mem : 0.0 total, 0.0 free, 0.0 used, 0.0 buff/cache  
PiB Swap: 0.0 total, 0.0 free, 0.0 used, 0.0 avail Mem  
scroll coordinates: y = 1/165 (tasks), x = 1/12 (fields)  
PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND  
7548 oracle 20 0 110.6m 1.2m 1.0m R 90.4 0.1 3501:28 test1.sh  
7574 oracle 20 0 110.6m 1.2m 1.0m R 90.0 0.1 3499:47 test2.sh  
27717 belobro+ 20 0 110.8m 1.5m 1.1m S 1.7 0.1 0:44.41 belobrov.sh  
7577 oracle 39 19 110.6m 1.2m 1.0m R 1.3 0.1 52:49.84 test3.sh  
25939 babich_+ 20 0 110.8m 1.5m 1.1m S 1.3 0.1 0:19.68 babich.sh  
13532 belobro+ 20 0 110.8m 1.5m 1.1m S 1.0 0.1 0:15.64 belobrov2.sh  
14315 belobro+ 30 10 110.8m 1.5m 1.1m S 0.7 0.1 0:11.84 belobrov3.sh  
9 root 20 0 0.0m 0.0m 0.0m S 0.3 0.0 141:01.68 rcu_sched  
500 root 20 0 465.3m 1.7m 1.0m R 0.3 0.1 771:01.82 NetworkManager  
7540 kostets+ 20 0 158.5m 2.4m 1.6m R 0.3 0.1 0:00.77 top  
12491 veselko+ 20 0 158.3m 2.3m 1.6m S 0.3 0.1 0:00.72 top  
1 root 20 0 186.7m 2.9m 1.6m S 0.0 0.2 5345:28 systemd  
2 root 20 0 0.0m 0.0m 0.0m S 0.0 0.0 0:01.35 kthreadd  
4 root 0 -20 0.0m 0.0m 0.0m S 0.0 0.0 0:00.00 kworker/0:0H  
6 root 20 0 0.0m 0.0m 0.0m S 0.0 0.0 0:24.38 ksoftirqd/0  
7 root rt 0 0.0m 0.0m 0.0m S 0.0 0.0 0:09.86 migration/0  
8 root 20 0 0.0m 0.0m 0.0m S 0.0 0.0 0:00.00 rcu_bh  
10 root 0 -20 0.0m 0.0m 0.0m S 0.0 0.0 0:00.00 lru-add-drain  
11 root rt 0 0.0m 0.0m 0.0m S 0.0 0.0 0:17.04 watchdog/0  
12 root rt 0 0.0m 0.0m 0.0m S 0.0 0.0 0:19.81 watchdog/1
```

3. Керування станами процесів

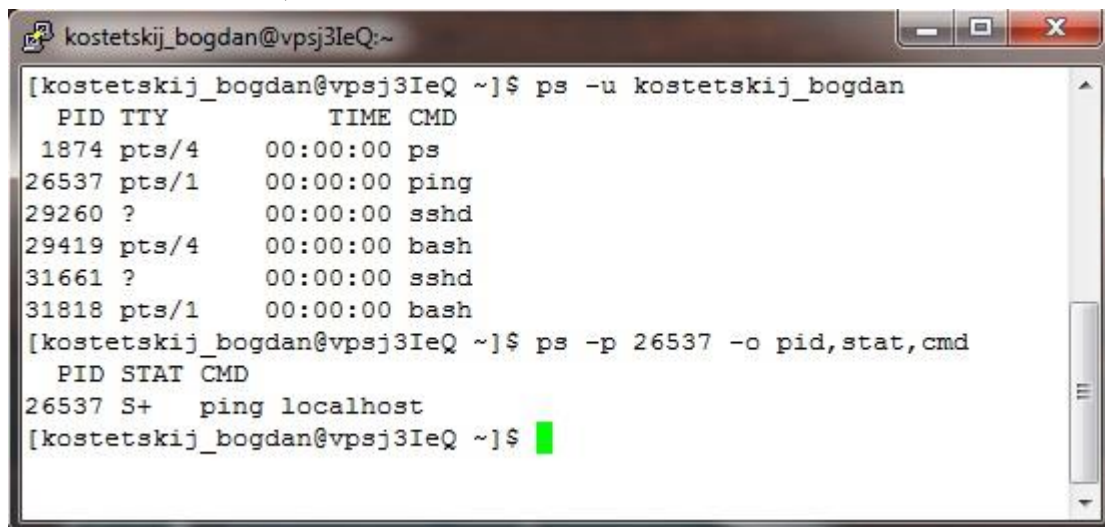
1. У поточному терміналі виконали команду `ping localhost`, але не завершували її роботу.



```
kostetskij_bogdan@vpsj3IeQ:~  
[kostetskij_bogdan@vpsj3IeQ ~]$ ping localhost  
PING localhost (127.0.0.1) 56(84) bytes of data.  
64 bytes from localhost (127.0.0.1): icmp_seq=1 ttl=64 time=0.016 ms  
64 bytes from localhost (127.0.0.1): icmp_seq=2 ttl=64 time=0.028 ms  
64 bytes from localhost (127.0.0.1): icmp_seq=3 ttl=64 time=0.035 ms  
64 bytes from localhost (127.0.0.1): icmp_seq=4 ttl=64 time=0.025 ms  
64 bytes from localhost (127.0.0.1): icmp_seq=5 ttl=64 time=0.016 ms  
64 bytes from localhost (127.0.0.1): icmp_seq=6 ttl=64 time=0.030 ms  
64 bytes from localhost (127.0.0.1): icmp_seq=7 ttl=64 time=0.026 ms  
64 bytes from localhost (127.0.0.1): icmp_seq=8 ttl=64 time=0.015 ms  
64 bytes from localhost (127.0.0.1): icmp_seq=9 ttl=64 time=0.023 ms  
64 bytes from localhost (127.0.0.1): icmp_seq=10 ttl=64 time=0.023 ms  
64 bytes from localhost (127.0.0.1): icmp_seq=11 ttl=64 time=0.027 ms  
64 bytes from localhost (127.0.0.1): icmp_seq=12 ttl=64 time=0.027 ms  
64 bytes from localhost (127.0.0.1): icmp_seq=13 ttl=64 time=0.029 ms  
64 bytes from localhost (127.0.0.1): icmp_seq=14 ttl=64 time=0.023 ms  
64 bytes from localhost (127.0.0.1): icmp_seq=15 ttl=64 time=0.020 ms
```

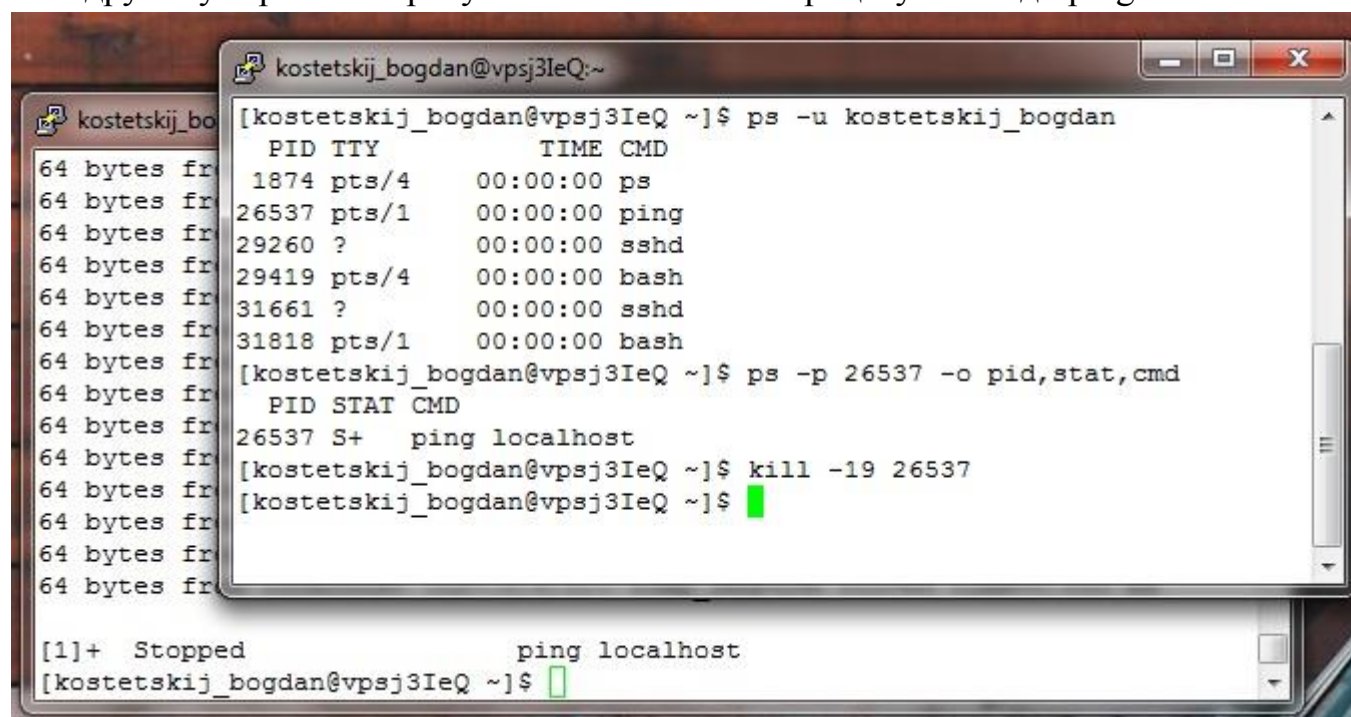
2. Запустили другий термінал доступу до Linux-сервера.

3. У другому терміналі для команди `ping` отримали таблицю її процесу (колонки PID, STAT, CMD).



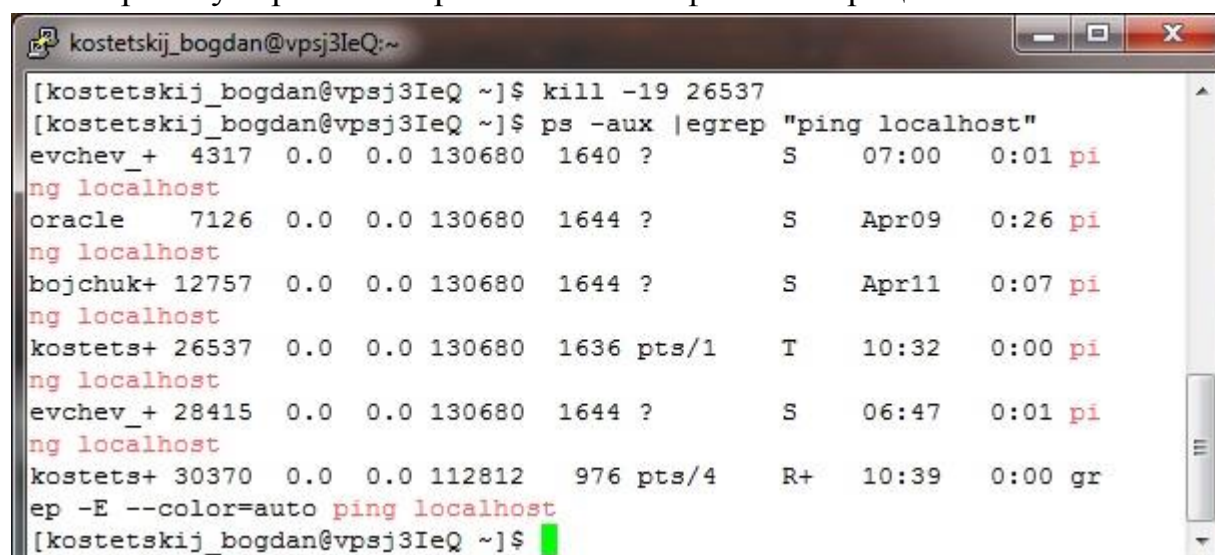
```
kostetskij_bogdan@vpsj3IeQ:~  
[kostetskij_bogdan@vpsj3IeQ ~]$ ps -u kostetskij_bogdan  
  PID TTY          TIME CMD  
 1874 pts/4      00:00:00 ps  
26537 pts/1      00:00:00 ping  
29260 ?           00:00:00 sshd  
29419 pts/4      00:00:00 bash  
31661 ?           00:00:00 sshd  
31818 pts/1      00:00:00 bash  
[kostetskij_bogdan@vpsj3IeQ ~]$ ps -p 26537 -o pid,stat,cmd  
  PID STAT CMD  
26537 S+   ping localhost  
[kostetskij_bogdan@vpsj3IeQ ~]$
```


4. У другому терміналі призупинили виконання процесу команди ping.



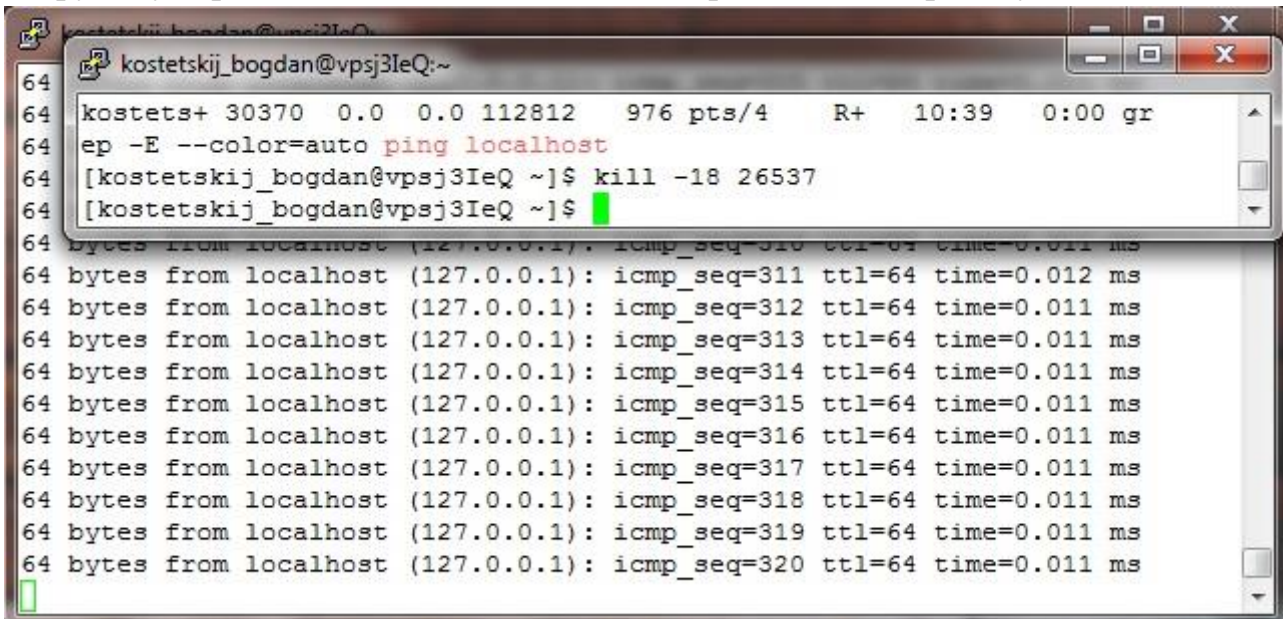
```
kostetskij_bogdan@vpsj3IeQ:~  
[kostetskij_bogdan@vpsj3IeQ ~]$ ps -u kostetskij_bogdan  
  PID TTY          TIME CMD  
 1874 pts/4        00:00:00 ps  
 26537 pts/1        00:00:00 ping  
 29260 ?              00:00:00 sshd  
 29419 pts/4        00:00:00 bash  
 31661 ?              00:00:00 sshd  
 31818 pts/1        00:00:00 bash  
[kostetskij_bogdan@vpsj3IeQ ~]$ ps -p 26537 -o pid,stat,cmd  
  PID STAT CMD  
 26537 S+    ping localhost  
[kostetskij_bogdan@vpsj3IeQ ~]$ kill -19 26537  
[kostetskij_bogdan@vpsj3IeQ ~]$  
  
[1]+  Stopped                  ping localhost  
[kostetskij_bogdan@vpsj3IeQ ~]$
```

5. У першому терміналі отримали список фонових процесів.



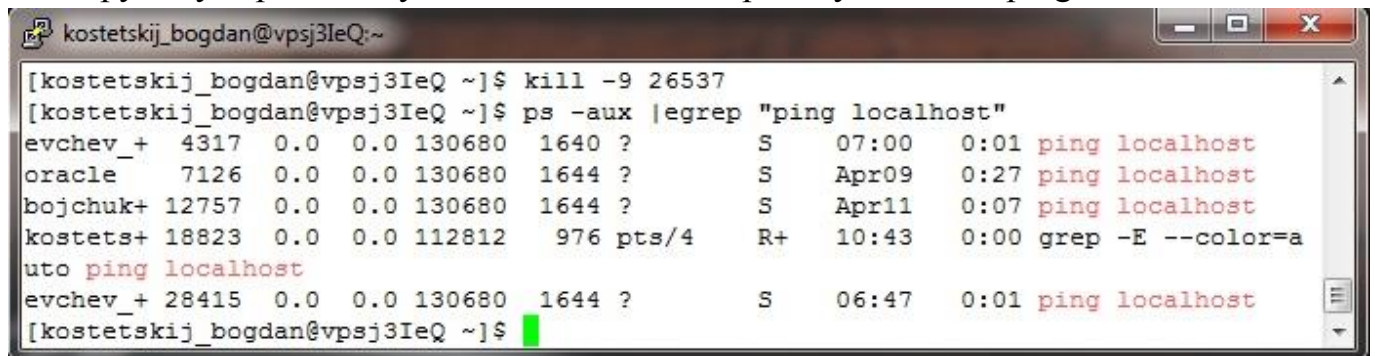
```
kostetskij_bogdan@vpsj3IeQ:~  
[kostetskij_bogdan@vpsj3IeQ ~]$ kill -19 26537  
[kostetskij_bogdan@vpsj3IeQ ~]$ ps -aux | egrep "ping localhost"  
evchev_+ 4317  0.0  0.0 130680 1640 ?        S    07:00   0:01 pi  
ng localhost  
oracle   7126  0.0  0.0 130680 1644 ?        S    Apr09   0:26 pi  
ng localhost  
bojchuk+ 12757  0.0  0.0 130680 1644 ?        S    Apr11   0:07 pi  
ng localhost  
kostets+ 26537  0.0  0.0 130680 1636 pts/1    T    10:32   0:00 pi  
ng localhost  
evchev_+ 28415  0.0  0.0 130680 1644 ?        S    06:47   0:01 pi  
ng localhost  
kostets+ 30370  0.0  0.0 112812  976 pts/4    R+   10:39   0:00 gr  
ep -E --color=auto ping localhost  
[kostetskij_bogdan@vpsj3IeQ ~]$
```

6. У другому терміналі відновили виконання припиненого процесу.



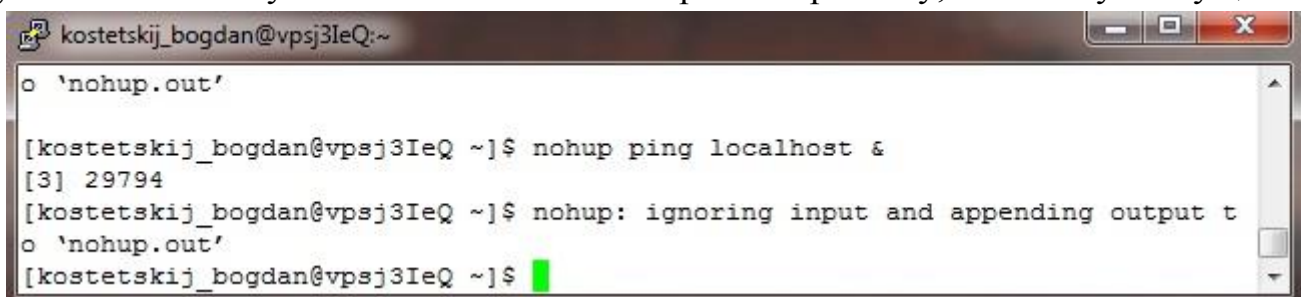
```
kostetskij_bogdan@vpsj3IeQ:~  
64 kostets+ 30370 0.0 0.0 112812 976 pts/4 R+ 10:39 0:00 gr  
64 ep -E --color=auto ping localhost  
64 [kostetskij_bogdan@vpsj3IeQ ~]$ kill -18 26537  
64 [kostetskij_bogdan@vpsj3IeQ ~]$  
64 bytes from localhost (127.0.0.1): icmp_seq=310 ttl=64 time=0.011 ms  
64 bytes from localhost (127.0.0.1): icmp_seq=311 ttl=64 time=0.012 ms  
64 bytes from localhost (127.0.0.1): icmp_seq=312 ttl=64 time=0.011 ms  
64 bytes from localhost (127.0.0.1): icmp_seq=313 ttl=64 time=0.011 ms  
64 bytes from localhost (127.0.0.1): icmp_seq=314 ttl=64 time=0.011 ms  
64 bytes from localhost (127.0.0.1): icmp_seq=315 ttl=64 time=0.011 ms  
64 bytes from localhost (127.0.0.1): icmp_seq=316 ttl=64 time=0.011 ms  
64 bytes from localhost (127.0.0.1): icmp_seq=317 ttl=64 time=0.011 ms  
64 bytes from localhost (127.0.0.1): icmp_seq=318 ttl=64 time=0.011 ms  
64 bytes from localhost (127.0.0.1): icmp_seq=319 ttl=64 time=0.011 ms  
64 bytes from localhost (127.0.0.1): icmp_seq=320 ttl=64 time=0.011 ms
```

7. У другому терміналі зупинили виконання процесу команди ping.



```
kostetskij_bogdan@vpsj3IeQ:~  
[kostetskij_bogdan@vpsj3IeQ ~]$ kill -9 26537  
[kostetskij_bogdan@vpsj3IeQ ~]$ ps -aux |egrep "ping localhost"  
evchev_+ 4317 0.0 0.0 130680 1640 ? S 07:00 0:01 ping localhost  
oracle_ 7126 0.0 0.0 130680 1644 ? S Apr09 0:27 ping localhost  
bojchuk+ 12757 0.0 0.0 130680 1644 ? S Apr11 0:07 ping localhost  
kostets+ 18823 0.0 0.0 112812 976 pts/4 R+ 10:43 0:00 grep -E --color=a  
uto ping localhost  
evchev_+ 28415 0.0 0.0 130680 1644 ? S 06:47 0:01 ping localhost  
[kostetskij_bogdan@vpsj3IeQ ~]$
```

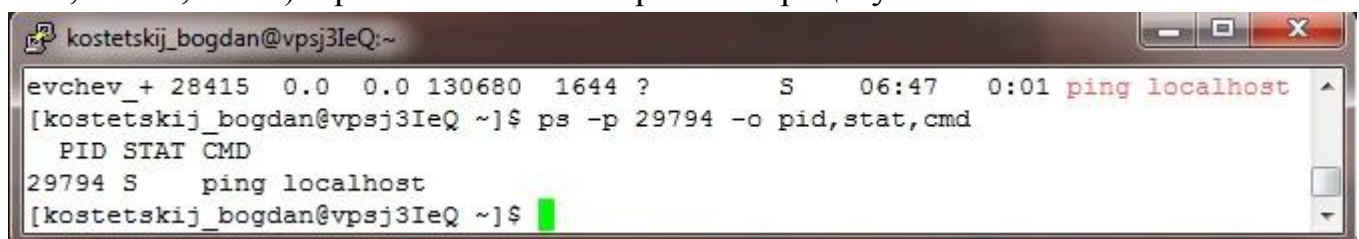
8. У першому терміналі запустили команду ping в фоновому режимі так, щоб він не був автоматично зупинений навіть після закриття терміналу, з якого був запусканий.



```
kostetskij_bogdan@vpsj3IeQ:~  
o 'nohup.out'  
[kostetskij_bogdan@vpsj3IeQ ~]$ nohup ping localhost &  
[3] 29794  
[kostetskij_bogdan@vpsj3IeQ ~]$ nohup: ignoring input and appending output t  
o 'nohup.out'  
[kostetskij_bogdan@vpsj3IeQ ~]$
```

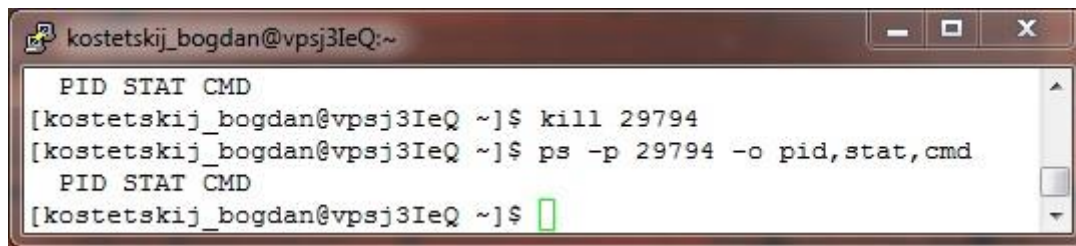
9. Закрили перший термінал.

10. У другому терміналі для команди ping отримали таблицю її процесу (колони PID, STAT, CMD). Зробили висновок про стан процесу.



```
kostetskij_bogdan@vpsj3IeQ:~  
evchev_+ 28415 0.0 0.0 130680 1644 ? S 06:47 0:01 ping localhost  
[kostetskij_bogdan@vpsj3IeQ ~]$ ps -p 29794 -o pid,stat,cmd  
PID STAT CMD  
29794 S ping localhost  
[kostetskij_bogdan@vpsj3IeQ ~]$
```



11. Завершили роботу процесу.



```
kostetskij_bogdan@vpsj3IeQ:~  
PID STAT CMD  
[kostetskij_bogdan@vpsj3IeQ ~]$ kill 29794  
[kostetskij_bogdan@vpsj3IeQ ~]$ ps -p 29794 -o pid,stat,cmd  
PID STAT CMD  
[kostetskij_bogdan@vpsj3IeQ ~]$
```

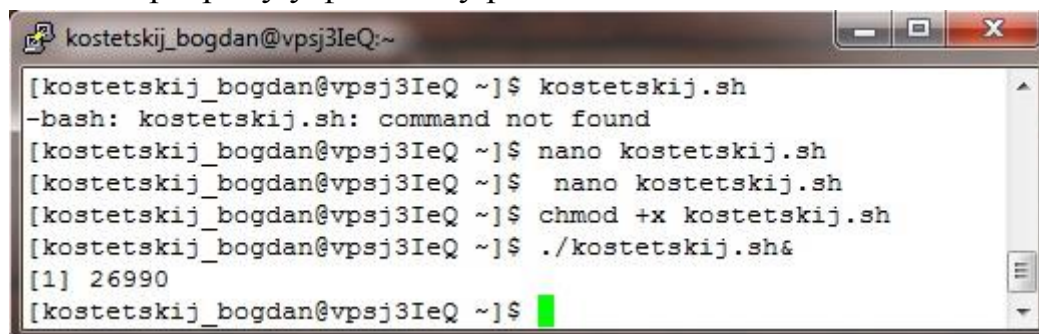
4.Управління пріоритетами процесів

1. Створіть bash-програму, що виконує операцію циклічного складання за формулою: $x = x + n$, де початкове значення x = кількість букв вашого прізвища, n - кількість букв у вашому імені. Ім'я програми збігається з транслітерацією вашого прізвища з розширенням .sh, наприклад, ivanov.sh



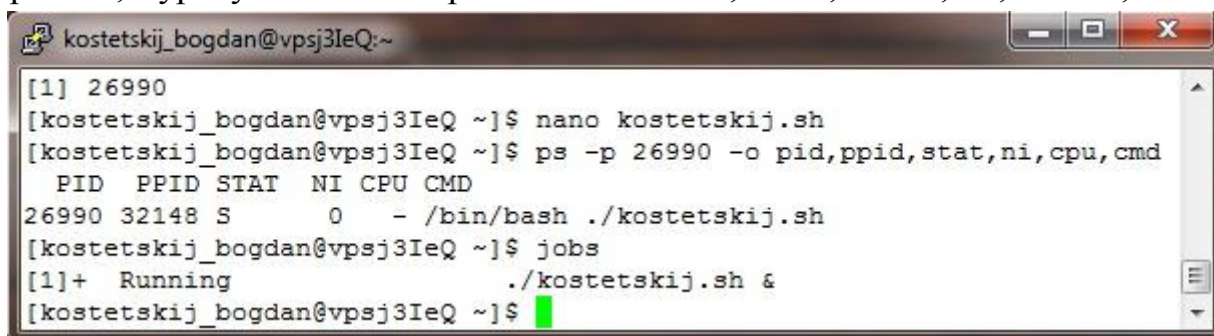
```
GNU nano 2.3.1 File: kostetskij.sh  
  
#!/bin/bash  
x=10  
n=6  
while( true )  
do  
    x=$((x+n))  
done  
  
^G Get H^O Write^R Read ^Y Prev ^K Cut T^C Cur Pos  
^X Exit ^J Justi^W Where^V Next ^U UnCut^T To Spell
```

2. Запустіть bash-програму у фоновому режимі.



```
kostetskij_bogdan@vpsj3IeQ:~  
[kostetskij_bogdan@vpsj3IeQ ~]$ kostetskij.sh  
-bash: kostetskij.sh: command not found  
[kostetskij_bogdan@vpsj3IeQ ~]$ nano kostetskij.sh  
[kostetskij_bogdan@vpsj3IeQ ~]$ nano kostetskij.sh  
[kostetskij_bogdan@vpsj3IeQ ~]$ chmod +x kostetskij.sh  
[kostetskij_bogdan@vpsj3IeQ ~]$ ./kostetskij.sh &  
[1] 26990  
[kostetskij_bogdan@vpsj3IeQ ~]$
```

3. Перегляньте таблицю процесів для запущеного процесу, пов'язаного з bash-програмою, з урахуванням набором колонок: PID, PPID, STAT, NI,% CPU, CMD.



```
kostetskij_bogdan@vpsj3IeQ:~  
[1] 26990  
[kostetskij_bogdan@vpsj3IeQ ~]$ nano kostetskij.sh  
[kostetskij_bogdan@vpsj3IeQ ~]$ ps -p 26990 -o pid,ppid,stat,ni,cpu,cmd  
PID PPID STAT NI CPU CMD  
26990 32148 S 0 - /bin/bash ./kostetskij.sh  
[kostetskij_bogdan@vpsj3IeQ ~]$ jobs  
[1]+ Running ./kostetskij.sh &  
[kostetskij_bogdan@vpsj3IeQ ~]$
```

4. Виконайте команду призупинення запущеного процесу.

```
kostetskij_bogdan@vpsj3IeQ:~  
[kostetskij_bogdan@vpsj3IeQ ~]$ jobs  
[1]+  Running                  ./kostetskij.sh &  
[kostetskij_bogdan@vpsj3IeQ ~]$ kill -19 26990  
[kostetskij_bogdan@vpsj3IeQ ~]$ jobs  
[1]+  Stopped                  ./kostetskij.sh  
[kostetskij_bogdan@vpsj3IeQ ~]$
```

5. Ще раз перегляньте таблицю процесів для призупиненого процесу з урахуванням набором колонок: PID, PPID, STAT, NI, %CPU, CMD. Зробіть висновки про його стан.

```
kostetskij_bogdan@vpsj3IeQ:~  
[kostetskij_bogdan@vpsj3IeQ ~]$ ps -p 26990 -o pid,ppid,stat,ni,cpu,cmd  
PID PPID STAT NI CPU CMD  
26990 32148 T 0 - /bin/bash ./kostetskij.sh  
[kostetskij_bogdan@vpsj3IeQ ~]$
```

6. Виконайте команду продовження виконання припиненого процесу.

```
kostetskij_bogdan@vpsj3IeQ:~  
[kostetskij_bogdan@vpsj3IeQ ~]$ kill -19 26990  
[kostetskij_bogdan@vpsj3IeQ ~]$ kill -18 26990  
[kostetskij_bogdan@vpsj3IeQ ~]$ jobs  
[1]+  Running                  ./kostetskij.sh &  
[kostetskij_bogdan@vpsj3IeQ ~]$
```

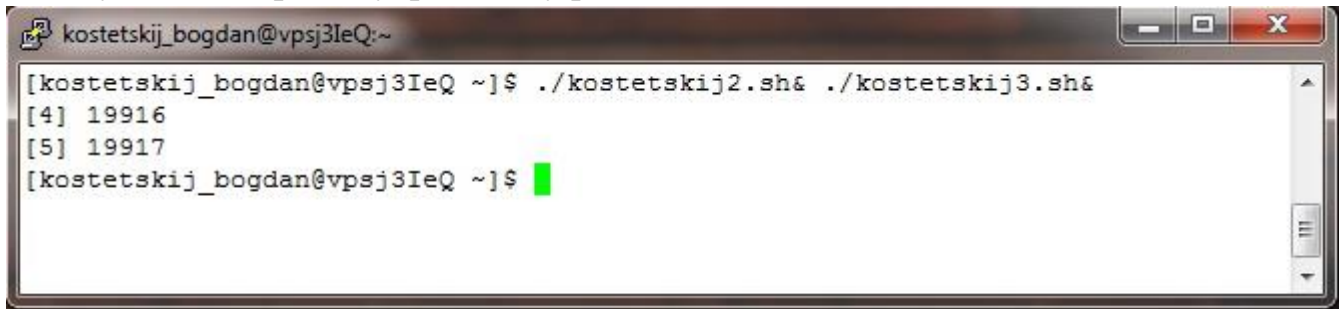
7. Ще раз перегляньте таблицю процесів для процесу, який продовжив виконуватися, з урахуванням набором колонок: PID, PPID, STAT, NI, %CPU, CMD. Зробіть висновки про його стан.

```
kostetskij_bogdan@vpsj3IeQ:~  
[1]+  Running                  ./kostetskij.sh &  
[kostetskij_bogdan@vpsj3IeQ ~]$ ps -p 26990 -o pid,ppid,stat,ni,cpu,cmd  
PID PPID STAT NI CPU CMD  
26990 32148 S 0 - /bin/bash ./kostetskij.sh  
[kostetskij_bogdan@vpsj3IeQ ~]$
```

8. Створіть два файли як символічні посилання на створену bash-програму з іменами як ім'я поточного файлу з додаванням цифр 2 і 3, відповідно, наприклад: ivanov2.sh, ivanov3.sh

```
kostetskij_bogdan@vpsj3IeQ:~  
[kostetskij_bogdan@vpsj3IeQ ~]$ ln -s kostetskij.sh kostetskij2.sh  
[kostetskij_bogdan@vpsj3IeQ ~]$ ln -s kostetskij.sh kostetskij3.sh  
[kostetskij_bogdan@vpsj3IeQ ~]$ ls  
1.csv          kostetskij2.sh      my_creat_file.sh  
accounts.csv   kostetskij3.sh      MyOSParam.sh  
Catalog       KostetskijBogdan.csv nohup.out  
Depozyt.txt    KostetskijBogdan.docx Operating-System.Laboratory-Work-1  
file_csv.sh    KostetskijBogdan.pdf os.lab1.cp1251.html  
file.html      kostetskij_lab_3    os.lab1.utf.html  
folder         kostetskij.sh       pustoy  
hard_link_1    lab.csv             sym_link_1  
hard_link_2    my_create_file.sh  
[kostetskij_bogdan@vpsj3IeQ ~]$
```


9. Запустіть два файли у фоновому режимі.



```
kostetskij_bogdan@vpsj3IeQ:~  
[kostetskij_bogdan@vpsj3IeQ ~]$ ./kostetskij2.sh& ./kostetskij3.sh&  
[4] 19916  
[5] 19917  
[kostetskij_bogdan@vpsj3IeQ ~]$
```

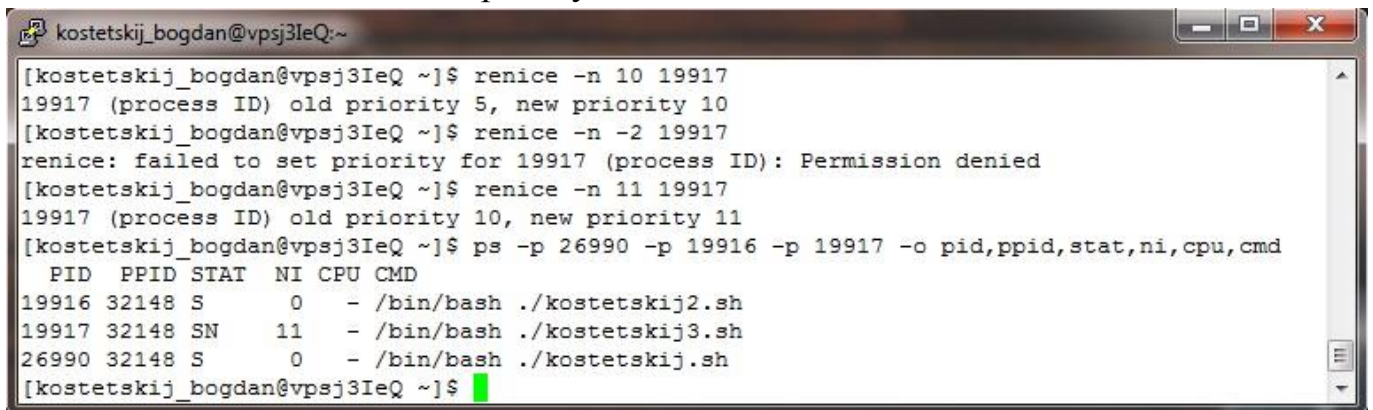
10. Перегляньте таблицю процесів для трьох запущених процесів з урахуванням набором колонок: PID, PPID, STAT, NI, %CPU, CMD. Зробіть висновки за поточними значеннями NI та %CPU.



```
kostetskij_bogdan@vpsj3IeQ:~  
[kostetskij_bogdan@vpsj3IeQ ~]$ ps -p 26990 -p 19916 -p 19917 -o pid,ppid,stat,ni,cpu,cmd  
PID PPID STAT NI CPU CMD  
19916 32148 S 0 - /bin/bash ./kostetskij2.sh  
19917 32148 S 0 - /bin/bash ./kostetskij3.sh  
26990 32148 S 0 - /bin/bash ./kostetskij.sh  
[kostetskij_bogdan@vpsj3IeQ ~]$ jobs  
[1] Running ./kostetskij.sh &  
[3] Running ./kostetskij3.sh &  
[4]- Running ./kostetskij2.sh &  
[5]+ Running ./kostetskij3.sh &  
[kostetskij_bogdan@vpsj3IeQ ~]$
```

11. Зменшити пріоритет виконання одного з трьох процесів.

12. Перегляньте таблицю процесів для трьох запущених файлів з урахуванням набором колонок: PID, PPID, STAT, NI, %CPU, CMD. Зробіть висновки щодо змін значень % CPU для кожного процесу: як вони змінилися?



```
kostetskij_bogdan@vpsj3IeQ:~  
[kostetskij_bogdan@vpsj3IeQ ~]$ renice -n 10 19917  
19917 (process ID) old priority 5, new priority 10  
[kostetskij_bogdan@vpsj3IeQ ~]$ renice -n -2 19917  
renice: failed to set priority for 19917 (process ID): Permission denied  
[kostetskij_bogdan@vpsj3IeQ ~]$ renice -n 11 19917  
19917 (process ID) old priority 10, new priority 11  
[kostetskij_bogdan@vpsj3IeQ ~]$ ps -p 26990 -p 19916 -p 19917 -o pid,ppid,stat,ni,cpu,cmd  
PID PPID STAT NI CPU CMD  
19916 32148 S 0 - /bin/bash ./kostetskij2.sh  
19917 32148 SN 11 - /bin/bash ./kostetskij3.sh  
26990 32148 S 0 - /bin/bash ./kostetskij.sh  
[kostetskij_bogdan@vpsj3IeQ ~]$
```

Висновки: в ході лабораторної роботи було отримано навичок в управлінні процесами в ОС Unix засобами командної оболонки. Майже всі завдання були доволі складні.