

Лабораторная работа №6

Отчет

Устинова Виктория Вадимовна

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Выводы	14
5	Ответы на контрольные вопросы	15

Список иллюстраций

3.1	командой jobs смотрим какие есть сейчас процессы	7
3.2	Останавливаем их с помощью ctrl +c	7
3.3	вводим exit чтобы закрыть терминал	8
3.4	Мы видим что команда все еще запущена 3805, выходим нажав q .	8
3.5	Находим и убиваем , теперь там нет ее	8
3.6	Вводим команду и меняем приоритет на 5	9
3.7	удаляем (kill -9) корневой оболочки	9
3.8	Произошла смена приоритета	9
3.9	Завершаем процессы с помощью fg 1 2 3 и ctrl+c	10
3.10	Заново запускаем программу ues с теми же параметрами, затем завершаем её выполнение.	10
3.11	jobs проверили, с помощью fg 1 и ctrl +z останавливаем процесс и затем переводим в фоновый режим с помощью bg	11
3.12	Запускаем и выходим, с помощью команды top смотрим, что он все еще запущен 4726	11
3.13	Запускаем и убиваем процессы с помощью двух команд	12
3.14	С помощью команд посылаем сигнал и запускаем процессы с подавлением потока, завершаем их всех	12
3.15	выполняем помледние команды и приоритет поменялся на 5 . . .	13

Список таблиц

1 Цель работы

Получить навыки управления процессами операционной системы.

2 Задание

1. Продемонстрируйте навыки управления заданиями операционной системы (см. раз-дел 6.4.1).
2. Продемонстрируйте навыки управления процессами операционной системы (см. раз-дел 6.4.2).
3. Выполните задания для самостоятельной работы (см. раздел 6.5

3 Выполнение лабораторной работы

Вводим команды и запускаем процессы, останавливаем процесс `ctrl +z` и продолжили запускать процесс в фоновом режиме с помощью `bg`(рис. 3.1).

```
[vvustinova@vvustinova ~]$ su -
Пароль:
[root@vvustinova ~]# sleep 3600 &
[1] 3721
[root@vvustinova ~]# dd if=/dev/zero of=/dev/null &
[2] 3730
[root@vvustinova ~]# sleep 7200
^Z
[3]+  Остановлен    sleep 7200
[root@vvustinova ~]# jobs
[1]-  Запущен       sleep 3600 &
[2]-  Запущен       dd if=/dev/zero of=/dev/null &
[3]+  Остановлен    sleep 7200
[root@vvustinova ~]# bg 3
[3]+ sleep 7200 &
[root@vvustinova ~]# jobs
[1]-  Запущен       sleep 3600 &
[2]-  Запущен       dd if=/dev/zero of=/dev/null &
[3]+  Запущен       sleep 7200 &
```

Рис. 3.1: командой `jobs` смотрим какие есть сейчас процессы

Для перемещения задания 1 на передний план введите `Fg 1` и остановите все процессы(рис. 3.2).

```
[root@vvustinova ~]# fg 1
sleep 3600
^C
[root@vvustinova ~]# jobs
[2]-  Запущен       dd if=/dev/zero of=/dev/null &
[3]+  Запущен       sleep 7200 &
[root@vvustinova ~]# fg 2
dd if=/dev/zero of=/dev/null
^C245189584+0 записей получено
245189583+0 записей отправлено
125537066496 байт (126 GB, 117 GiB) скопирован, 199,07 s, 631 MB/s
[root@vvustinova ~]# fg 3
sleep 7200
^C
[root@vvustinova ~]#
```

Рис. 3.2: Останавливаем их с помощью `ctrl +c`

Откройте второй терминал и под учётной записью своего пользователя введите в нём след.команду(рис. 3.3).

```

root@vvustinova:~ x
vvustinova@vvustinova:~ x
[vvustinova@vvustinova ~]$ dd if=/dev/zero of=/dev/null &
[1] 3805
[vvustinova@vvustinova ~]$ exit

```

Рис. 3.3: вводим exit чтобы закрыть терминал

На другом терминале под учётной записью своего пользователя запустите top(рис. 3.4).

```

[root@vvustinova ~]# top
top - 13:22:54 up 38 min, 2 users, load average: 1,00, 1,07, 0,92
Tasks: 240 total, 2 running, 238 sleeping, 0 stopped, 0 zombie
%Cpu(s): 3,9 us, 7,1 sy, 0,0 ni, 88,7 id, 0,0 wa, 0,0 hi, 0,3 si, 0,0 st
MiB Mem : 3910,7 total, 469,1 free, 1705,3 used, 2026,0 buff/cache
MiB Swap: 3072,0 total, 3072,0 free, 0,0 used, 2205,4 avail Mem

  PID USER      PR  NI   VIRT   RES    SHR S  %CPU  %MEM    TIME+  COMMAND
 3805 vvustin+  20   0 220988   1792   1792 R 100,0   0,0   2:09.60 dd
 2233 vvustin+  20   0 5703380 377888 132700 S  18,6   9,4   6:15.26 gnome-s+
 3445 vvustin+  20   0 835812  48740  38020 S   3,7   1,2   0:06.01 gnome-t+
 3828 root       20   0 225904   4352   3584 R   0,7   0,1   0:00.22 top

```

Рис. 3.4: Мы видим что команда все еще запущена 3805, выходим нажав q

Вновь запустите top и в нём используйте k , чтобы убить задание dd. После этого выйдите из top.(рис. 3.5).

```

[root@vvustinova ~]# top
top - 13:23:33 up 39 min, 2 users, load average: 1,78, 1,30, 1,00
Tasks: 238 total, 1 running, 237 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0,4 us, 0,3 sy, 0,0 ni, 99,1 id, 0,0 wa, 0,0 hi, 0,2 si, 0,0 st
MiB Mem : 3910,7 total, 471,2 free, 1703,2 used, 2022,9 buff/cache
MiB Swap: 3072,0 total, 3072,0 free, 0,0 used, 2207,5 avail Mem

  PID USER      PR  NI   VIRT   RES    SHR S  %CPU  %MEM    TIME+  COMMAND
 2233 vvustin+  20   0 5707556 377888 132700 S  10,3   9,4   6:23.30 gnome-s+
 3445 vvustin+  20   0 835812  48740  38020 S   1,0   1,2   0:06.52 gnome-t+
  17 root       20   0      0      0      0 I   0,3   0,0   0:10.57 rcu_pre+
 560 root       20   0      0      0      0 S   0,3   0,0   0:04.38 xfsaild+
1037 root       20   0 483704  28944  15360 S   0,3   0,7   0:03.52 tuned

```

Рис. 3.5: Находим и убиваем , теперь там нет ее

Вводим команды из туиса и команда ps aux | grep dd показывает нам строки где есть dd.Используйте PID одного из процессов dd, чтобы изменить приоритет.(рис. 3.6).


```

[root@vvustinova ~]# su -
[root@vvustinova ~]# dd if=/dev/zero of=/dev/null &
[1] 3924
[root@vvustinova ~]# dd if=/dev/zero of=/dev/null &
[2] 3928
[root@vvustinova ~]# dd if=/dev/zero of=/dev/null &
[3] 3937
[root@vvustinova ~]# ps aux | grep dd
root      2  0.0  0.0   0   0 ?        S   12:44   0:00 [kthreadd]
vvustin+ 2406  0.0  0.7 881864 30136 ?        Ssl 12:51   0:00 /usr/libexec/evolution-ad
-factory
root      3924 98.4  0.0 220988 1792 pts/0    R   13:27   1:48 dd if=/dev/zero of=/dev/n
root      3928 98.8  0.0 220988 1792 pts/0    R   13:27   1:34 dd if=/dev/zero of=/dev/n
root      3937 98.5  0.0 220988 1792 pts/0    R   13:28   1:17 dd if=/dev/zero of=/dev/n
root      3988  0.0  0.0 221820 2560 pts/0    R+  13:29   0:00 grep --color=auto dd
[root@vvustinova ~]# renice -n 5 3924
3924 (process ID) old priority 0, new priority 5

```

Рис. 3.6: Вводим команду и меняем приоритет на 5

Найдите PID корневой оболочки, из которой были запущены процессы dd, и введите команду(рис. 3.7).

```

3477 pts/0  Ss      0:00 | \_ bash
3679 pts/0  S        0:00 | \_ su -
3690 pts/0  S        0:00 | \_ -bash
3889 pts/0  S        0:00 | \_ su -
3892 pts/0  S        0:00 | \_ -bash
3924 pts/0  RN      3:30 | \_ dd if=/dev/zero of=/dev/null
3928 pts/0  R       3:16 | \_ dd if=/dev/zero of=/dev/null
3937 pts/0  R       2:59 | \_ dd if=/dev/zero of=/dev/null
4015 pts/0  R+      0:00 | \_ ps fax
4016 pts/0  S+      0:00 | \_ grep --color=auto -B5 dd
[root@vvustinova ~]# kill -9 3892
-bash: kill: : недопустимое указание сигнала
[root@vvustinova ~]# kill -9 3892
Убито
[root@vvustinova ~]#
[root@vvustinova ~]#

```

Рис. 3.7: удаляем (kill -9) корневой оболочки

Запускаем команду триджды как фоновое, затем меняем -5 на -15(рис. 3.8).

```

[root@vvustinova ~]# dd if=/dev/zero of=/dev/null &
[2] 4340
[root@vvustinova ~]#

[root@vvustinova ~]#
[root@vvustinova ~]# ^C
[root@vvustinova ~]# dd if=/dev/zero of=/dev/null &&
>
>
> q

^C36254041+0 записей получено
36254040+0 записей отправлено
18562068480 байт (19 GB, 17 GiB) скопирован, 34,2928 s, 541 MB/s

[root@vvustinova ~]# dd if=/dev/zero of=/dev/null &
[3] 4362
[root@vvustinova ~]# renice -n -5 4336
4336 (process ID) old priority 0, new priority -5
[root@vvustinova ~]# renice -n -15 4336
4336 (process ID) old priority -5, new priority -15

```

Рис. 3.8: Произошла смена приоритета

Завершите все процессы dd, которые вы запустили(рис. 3.9).

```
[root@vvustinova ~]# fg 1
dd if=/dev/zero of=/dev/null
^C317285968+0 записей получено
317285968+0 записей отправлено
162450415616 байт (162 GB, 151 GiB) скопирован, 304,18 s, 534 MB/s

[root@vvustinova ~]# fg 2
dd if=/dev/zero of=/dev/null
^C318240881+0 записей получено
318240881+0 записей отправлено
162939331072 байт (163 GB, 152 GiB) скопирован, 306,133 s, 532 MB/s

[root@vvustinova ~]# fg 3
dd if=/dev/zero of=/dev/null
^C210259419+0 записей получено
210259419+0 записей отправлено
107652822528 байт (108 GB, 100 GiB) скопирован, 208,077 s, 517 MB/s
```

Рис. 3.9: Завершаем процессы с помощью fg 1 2 3 и ctrl+c

Запускаем программу yes в фоновом с подавлением потока, а после на переднем с подавлением потока и останавливаем.(рис. 3.10).

```
[root@vvustinova ~]# yes > /dev/null &
[1] 4575
[root@vvustinova ~]# yes > /dev/null
^Z
[2]+  Остановлен    yes > /dev/null
[root@vvustinova ~]# yes > /dev/null
^C
[root@vvustinova ~]#
```

Рис. 3.10: Заново запускаем программу yes с теми же параметрами, затем завершаем её выполнение.

Проверяем состояние процессов.Переведите процесс, который у вас выполняется в фоновом режиме, на передний план, затем остановите его.Переведите любой ваш процесс с подавлением потока вывода в фоновый режим.(рис. 3.11).

```

[root@vvustinova ~]# jobs
[1]  Запущен          yes > /dev/null &
[2]- Остановлен      yes > /dev/null
[3]+ Остановлен      yes
[root@vvustinova ~]# fg 1
yes > /dev/null
^Z
[1]+ Остановлен      yes > /dev/null
[root@vvustinova ~]# bg 2
[2] yes > /dev/null &
[root@vvustinova ~]# jobs
[1]+ Остановлен      yes > /dev/null
[2]  Запущен          yes > /dev/null &
[3]- Остановлен      yes
[root@vvustinova ~]#

```

Рис. 3.11: jobs проверили, с помощью fg 1 и ctrl +z останавливаем процесс и затем переводим в фоновый режим с помощью bg

Запустите процесс в фоновом режиме таким образом, чтобы он продолжил свою работу даже после отключения от терминала. Закройте окно и заново запустите консоль. Убедитесь, что процесс продолжил свою работу. (рис. 3.12).

```

[root@vvustinova ~]# jobs
[root@vvustinova ~]# nohup yes > /dev/null &
[1] 4726
[root@vvustinova ~]# nohup: ввод игнорируется и поток ошибок перенаправляется на стандартный вывод
[~]# exit
[~]#
[~]# top
top - 14:29:17 up 1:44, 2 users, load average: 4.59, 4.63, 4.32
Tasks: 242 total, 6 running, 236 sleeping, 0 stopped, 0 zombie
%Cpu(s): 2.2 us, 6.3 sy, 0.6 ni, 90.3 id, 0.0 wa, 0.0 hi, 0.6 si, 0.0 st
MiB Mem : 3910.7 total, 395.1 free, 1734.7 used, 2071.6 buff/cache
MiB Swap: 3072.0 total, 3072.0 free, 0.0 used, 2176.0 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM    TIME+  COMMAND
  3928 root       20   0   220988    1792    1792 R   100.0   0.0   61:12.75 dd
  3924 root       25   5   220988    1792    1792 R   100.0   0.0   61:14.12 dd
  4578 root       20   0   220948    1792    1792 R   100.0   0.0    9:47.25 yes
  3937 root       20   0   220988    1792    1792 R   100.0   0.0   60:52.17 dd
  4726 root       20   0   220948    1792    1792 R   100.0   0.0    5:08.82 yes

```

Рис. 3.12: Запускаем и выходим, с помощью команды top смотрим, что он все еще запущен 4726

Запустите ещё три программы yes в фоновом режиме с подавлением потока вывода. Убейте два процесса: для одного используйте его PID, а для другого — его идентификатор конкретного задания. (рис. 3.13).

```
[vvustinova@vvustinova ~]$ yes >/dev/null &
[1] 4853

[vvustinova@vvustinova ~]$
[vvustinova@vvustinova ~]$
[vvustinova@vvustinova ~]$ yes >/dev/null &
[2] 4869
[vvustinova@vvustinova ~]$ yes >/dev/null &
[3] 4874
[vvustinova@vvustinova ~]$ fg 1
yes > /dev/null
^C
[vvustinova@vvustinova ~]$ kill -9 4869
[2]-  Убито          yes > /dev/null
[vvustinova@vvustinova ~]$
```

Рис. 3.13: Запускаем и убиваем процессы с помощью двух команд

Попробуйте послать сигнал 1 (SIGHUP) процессу, запущенному с помощью `nohup`, и обычному процессу. Запустите ещё несколько программ `yes` в фоновом режиме с подавлением потока выводов. Завершите их работу одновременно, используя команду `killall`. (рис. 3.14).

```
[vvustinova@vvustinova ~]$ ps aux | grep yes
root      4578  83.8  0.0 220948 1792 pts/0    R   14:16   17:46  yes
root      4726  96.8  0.0 220948 1792 pts/0    R   14:24   13:02  yes
vvustin+  4874  96.1  0.0 220948 1792 pts/0    R   14:32    4:46  yes
vvustin+  4934   0.0  0.0 221820 2560 pts/0    S+  14:37    0:00  grep --color=auto yes
[vvustinova@vvustinova ~]$ sudo kill -1 4726
[vvustinova@vvustinova ~]$ sudo kill -1 4874
[3]+  Обрыв терминальной линии          yes > /dev/null
[vvustinova@vvustinova ~]$ yes > /dev/null &
[1] 5024
[vvustinova@vvustinova ~]$ yes > /dev/null &
[2] 5031
[vvustinova@vvustinova ~]$ yes > /dev/null &
[3] 5038
[vvustinova@vvustinova ~]$ killall yes
bash: killall: команда не найдена...
Аналогичная команда: 'killall'
[vvustinova@vvustinova ~]$ killall yes
yes(4578): Операция не позволена
yes(4726): Операция не позволена
[1]  Завершено      yes > /dev/null
[3]+ Завершено      yes > /dev/null
[2]+ Завершено      yes > /dev/null
[vvustinova@vvustinova ~]$ nice -n 5 yes > dev/null &
```

Рис. 3.14: С помощью команд посылаем сигнал и запускаем процессы с подавлением потока, завершаем их всех

Запустите программу `yes` в фоновом режиме с подавлением потока вывода. Используя утилиту `nice`, запустите программу `yes` с теми же параметрами и с приоритетом, большим на 5. (рис. 3.15).

```
[vvustinova@vvustinova ~]$ nice -n 5 yes > /dev/null &  
[1] 5145  
[vvustinova@vvustinova ~]$  
[vvustinova@vvustinova ~]$ ps -l | grep yes  
0 R 1000 5145 3477 98 85 5 - 55237 - pts/0 00:05:54 yes  
[vvustinova@vvustinova ~]$
```

Рис. 3.15: выполняем помледние комнады и приоритет поменялся на 5

4 Выводы

У нас получилось приобрести навыки управления процессами операционной системы.

5 Ответы на контрольные вопросы

1. `jobs`
2. Чтобы остановить текущее задание оболочки и продолжить его выполнение в фоновом режиме, можно использовать клавиши `Ctrl+ Z`. Это приостановит задание, после чего можно набрать команду `bg (background)` чтобы переместить его в фон.
3. Для отмены (прерывания) текущего задания оболочки можно использовать комбинацию клавиш `Ctrl +C`.
4. Если доступ к оболочке пользователя невозможен, чтобы отменить задание, можно использовать команду `kill`. Сначала нужно узнать PID (Process ID) этого задания, например, с помощью `ps` или `top`. Затем, выполнить `kill`. Если процесс не завершается, можно использовать более сильный сигнал `kill -9`. Важно: `-9` следует использовать только в крайнем случае, так как это приводит к немедленному завершению процесса без возможности сохранения данных или корректной очистки.
5. Для отображения отношений между родительскими и дочерними процессами можно использовать команду `ps tree`.
6. Чтобы изменить приоритет процесса с идентификатором 1234 на более высокий, можно использовать команду `renice -n -5 1234` (замените `-5` на нужное значение, чем меньше число, тем выше приоритет, но требуются

права администратора). Обычно, нормальные пользователи могут только понижать приоритет.

7. Самый простой способ остановить все запущенные процессы `dd` сразу - использовать команду `killall dd`.
8. Чтобы остановить команду с именем `mysommand`, можно использовать команду `killall mysommand`. Если таких команд запущено несколько, команда остановит все.
9. В `top`, чтобы убить процесс, нужно нажать клавишу `k`, а затем ввести PID процесса, который нужно завершить, и сигнал (по умолчанию - 15, можно использовать 9 в крайнем случае).
10. Чтобы запустить команду с достаточно высоким приоритетом, но не рискуя нехваткой ресурсов для других процессов, можно использовать команду `nice -n -10 mysommand`. Аргумент `-n -10` задает приоритет. Чем ниже число (до -20, требуются права `root`), тем выше приоритет, но слишком высокий приоритет может замедлить работу системы. Оптимально использовать значение, немного выше нуля. Не рекомендуется запускать пользовательские процессы с высочайшим приоритетом.