

Лабораторная работа № 6

Управление процессами

Сергеев Д. О.

11 октября 2025

Российский университет дружбы народов, Москва, Россия

Информация

- Сергеев Даниил Олегович
- Студент
- Направление: Прикладная информатика
- Российский университет дружбы народов
- 1132246837@pfur.ru

Цель работы

Получить навыки управления процессами операционной системы.

Задание

- Продемонстрируйте навыки управления заданиями операционной системы
- Продемонстрируйте навыки управления процессами операционной системы
- Выполните задания для самостоятельной работы

Ход выполнения лабораторной работы

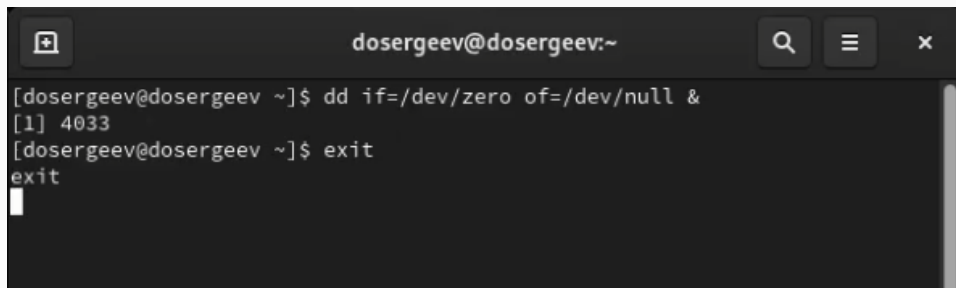
```
[dosergeev@dosergeev ~]$ su -  
Password:  
[root@dosergeev ~]# sleep 3600 &  
[1] 3982  
[root@dosergeev ~]# dd if=/dev/zero of=/dev/null &  
[2] 3990  
[root@dosergeev ~]# sleep 7200  
^Z  
[3]+  Stopped                  sleep 7200  
[root@dosergeev ~]# jobs  
[1]   Running                  sleep 3600 &  
[2]-  Running                  dd if=/dev/zero of=/dev/null &  
[3]+  Stopped                  sleep 7200
```

Рис. 1: Остановка зависшего процесса

Для продолжения выполнения задания 3 в фоновом режиме введем команду `bg 3`. Переместим задание 1 на передний план и введем `fg 1`, после чего отменим задание 1 командой `Ctrl+C`. Проверим статус задач командой `jobs`. Аналогично сделаем для отмены заданий 2 и 3.

```
[root@dosergeev ~]# bg 3
[3]+ sleep 7200 &
[root@dosergeev ~]# jobs
[1]  Running                sleep 3600 &
[2]-  Running                dd if=/dev/zero of=/dev/null &
[3]+  Running                sleep 7200 &
[root@dosergeev ~]# fg 1
sleep 3600
^C
[root@dosergeev ~]# jobs
[2]-  Running                dd if=/dev/zero of=/dev/null &
[3]+  Running                sleep 7200 &
```

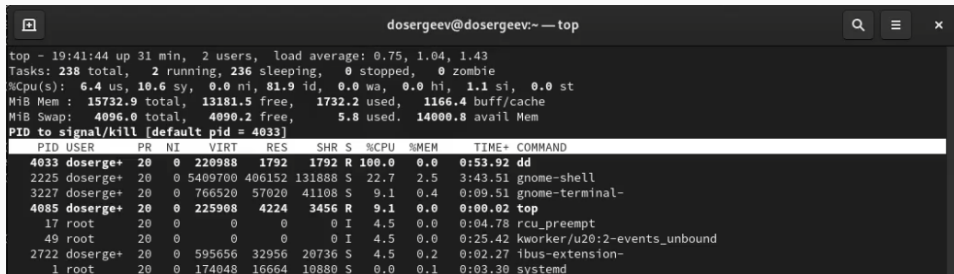
Рис. 2: Переключение режимов процесса



```
dosergeev@dosergeev:~  
[dosergeev@dosergeev ~]$ dd if=/dev/zero of=/dev/null &  
[1] 4033  
[dosergeev@dosergeev ~]$ exit  
exit  
█
```

Рис. 3: Фоновой процесс в другом терминале

запустим top и убьём задание dd, нажав k и указав PID процесса и номер сигнала (9/termkill).



```
dosergeev@dosergeev:~ — top
top - 19:41:44 up 31 min, 2 users, load average: 0.75, 1.04, 1.43
Tasks: 238 total, 2 running, 236 sleeping, 0 stopped, 0 zombie
%Cpu(s): 6.4 us, 10.6 sy, 0.0 ni, 81.9 id, 0.0 wa, 0.0 hi, 1.1 si, 0.0 st
MiB Mem : 15732.9 total, 13181.5 free, 1732.2 used, 1166.4 buff/cache
MiB Swap: 4096.0 total, 4090.2 free, 5.8 used, 14000.8 avail Mem
PID to signal/kill [default pid = 4033]
  PID USER      PR  NI   VIRT    RES    SHR S  %CPU  %MEM    TIME+  COMMAND
 4033 doserge+  20   0 220988   1792   1792 R 100.0   0.0   0:53.92  dd
 2225 doserge+  20   0 5409700 406152 131888 S 22.7   2.5   3:43.51  gnome-shell
 3227 doserge+  20   0 766520   57020 41108 S  9.1   0.4   0:09.51  gnome-terminal-
 4085 doserge+  20   0 225908   4224   3456 R  9.1   0.0   0:00.02  top
    17 root      20   0      0      0      0 I  4.5   0.0   0:04.78  rcu_preempt
    49 root      20   0      0      0      0 I  4.5   0.0   0:25.42  kworker/u20:2-events_unbound
 2722 doserge+  20   0 595656  32956 20736 S  4.5   0.2   0:02.27  ibus-extension-
     1 root      20   0 174048  16664 10880 S  0.0   0.1   0:03.30  systemd
```

Рис. 4: Процесс dd в top

Управление процессами

```
[root@dosergeev ~]# dd if=/dev/zero of=/dev/null &
[1] 4092
[root@dosergeev ~]# dd if=/dev/zero of=/dev/null &
[2] 4093
[root@dosergeev ~]# dd if=/dev/zero of=/dev/null &
[3] 4094
[root@dosergeev ~]# ps aux | grep dd
root      2  0.0  0.0      0   0 ?        S   19:10   0:00 [kthreadd]
root     1096  0.0  0.0 508476 3968 ?        Sl  19:10   0:01 /usr/sbin/VBoxService --pidfile /var/run/vboxadd-service.sh
doserge+ 2393  0.0  0.1 881212 30120 ?        Ssl 19:12   0:00 /usr/libexec/evolution-addressbook-factory
root      4092 100  0.0 220988 1792 pts/0    R   19:42   0:07 dd if=/dev/zero of=/dev/null
root      4093 103  0.0 220988 1792 pts/0    R   19:42   0:06 dd if=/dev/zero of=/dev/null
root      4094 109  0.0 220988 1792 pts/0    R   19:42   0:05 dd if=/dev/zero of=/dev/null
root      4096  0.0  0.0 221796 2560 pts/0    S+  19:42   0:00 grep --color=auto dd
[root@dosergeev ~]# renice -n 5 4092
4092 (process ID) old priority 0, new priority 5
```

Рис. 5: Вывод процессов dd. Изменение приоритета

ps fax | grep -B5 dd

```
2881 ?      Ssl    0:00  \_ /usr/libexec/gvfsd-metadata
3227 ?      Ssl    0:11  \_ /usr/libexec/gnome-terminal-server
3245 pts/0    Ss     0:00      \_ bash
3941 pts/0    S      0:00          \_ su -
3952 pts/0    S      0:00              \_ -bash
4092 pts/0    RN     0:57              \_ dd if=/dev/zero of=/dev/null
4093 pts/0    R      0:56              \_ dd if=/dev/zero of=/dev/null
4094 pts/0    R      0:55              \_ dd if=/dev/zero of=/dev/null
4098 pts/0    R+     0:00              \_ ps fax
4099 pts/0    S+     0:00              \_ grep --color=auto -B5 dd
[root@dosergeev ~]# kill -9 3245
```

Рис. 6: Остановка дочерних процессов

Задания для самостоятельной работы

Запустим команду `dd if=/dev/zero of=/dev/null` три раза на фоне.

```
[dosergeev@dosergeev ~]$ ps axl | grep dd
1  0      2      0  20  0      0      0 -    S    ?      0:00 [kthreadd]
1  0      1096    1  20  0 508476 3968 -    Sl   ?      0:01 /usr/sbin/VBoxService --pidfile /var/run/vboxadd-service.sh
0 1000    2393    2138 20  0 881212 30120 do_pol Ssl  ?      0:00 /usr/libexec/evolution-addressbook-factory
0 1000    4230    4200 15 -5 220988 1792 -    R<   pts/0    1:02 dd if=/dev/zero of=/dev/null
0 1000    4235    4200 20  0 220988 1792 -    R    pts/0    1:01 dd if=/dev/zero of=/dev/null
0 1000    4240    4200 20  0 220988 1792 -    R    pts/0    0:58 dd if=/dev/zero of=/dev/null
0 1000    4271    4200 20  0 221796 2560 pipe_r S+   pts/0    0:00 grep --color=auto dd
[dosergeev@dosergeev ~]$ sudo renice -15 4230
4230 (process ID) old priority -5, new priority -15
[dosergeev@dosergeev ~]$ ps axl | grep dd
1  0      2      0  20  0      0      0 -    S    ?      0:00 [kthreadd]
1  0      1096    1  20  0 508476 3968 -    Sl   ?      0:01 /usr/sbin/VBoxService --pidfile /var/run/vboxadd-service.sh
0 1000    2393    2138 20  0 881212 30120 do_pol Ssl  ?      0:00 /usr/libexec/evolution-addressbook-factory
0 1000    4230    4200 5 -15 220988 1792 -    R<   pts/0    1:40 dd if=/dev/zero of=/dev/null
0 1000    4200    4200 20  0 220988 1792 -    R    pts/0    1:39 dd if=/dev/zero of=/dev/null
0 1000    4240    4200 20  0 220988 1792 -    R    pts/0    1:36 dd if=/dev/zero of=/dev/null
0 1000    4288    4200 20  0 221796 2560 pipe_r S+   pts/0    0:00 grep --color=auto dd
[dosergeev@dosergeev ~]$
```

Рис. 7: Изменение приоритета команды `dd`

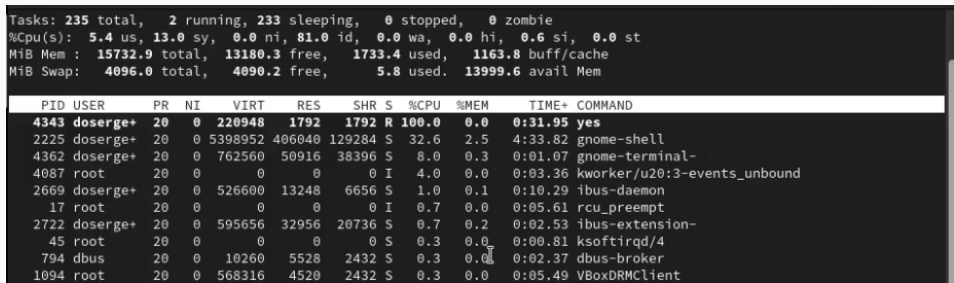
```
[dosergeev@dosergeev ~]$ yes > /dev/null &  
[1] 4301  
[dosergeev@dosergeev ~]$ yes > /dev/null  
^Z  
[2]+  Stopped                  yes > /dev/null  
[dosergeev@dosergeev ~]$ yes > /dev/null  
^C  
[dosergeev@dosergeev ~]$ yes >
```

Рис. 8: Работа с командой yes с подавлением потока

```
[dosergeev@dosergeev ~]$ jobs
[1]    Running                  yes > /dev/null &
[2]-  Stopped                  yes > /dev/null
[3]+  Stopped                  yes
[dosergeev@dosergeev ~]$ bg 2
[2]- yes > /dev/null &
[dosergeev@dosergeev ~]$ jobs
[1]    Running                  yes > /dev/null &
[2]-  Running                  yes > /dev/null &
[3]+  Stopped                  yes
[dosergeev@dosergeev ~]$
```

Рис. 9: Перевод на фоновой режим процесса yes

nohup yes > /dev/null &. Закроем терминал и в новом окне проверим состояние процесса в окне top.



```
Tasks: 235 total,  2 running, 233 sleeping,  0 stopped,  0 zombie
%Cpu(s):  5.4 us, 13.0 sy,  0.0 ni, 81.0 id,  0.0 wa,  0.0 hi,  0.6 si,  0.0 st
MiB Mem : 15732.9 total, 13180.3 free, 1733.4 used, 1163.8 buff/cache
MiB Swap: 4096.0 total, 4090.2 free,  5.8 used, 13999.6 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
4343	doserge+	20	0	220948	1792	1792	R	100.0	0.0	0:31.95	yes
2225	doserge+	20	0	5398952	406040	129284	S	32.6	2.5	4:33.82	gnome-shell
4362	doserge+	20	0	762560	50916	38396	S	8.0	0.3	0:01.07	gnome-terminal-
4087	root	20	0	0	0	0	I	4.0	0.0	0:03.36	kworker/u20:3-events_unbound
2669	doserge+	20	0	526600	13248	6656	S	1.0	0.1	0:10.29	ibus-daemon
17	root	20	0	0	0	0	I	0.7	0.0	0:05.61	rcu_preempt
2722	doserge+	20	0	595656	32956	20736	S	0.7	0.2	0:02.53	ibus-extension-
45	root	20	0	0	0	0	S	0.3	0.0	0:00.81	ksoftirqd/4
794	dbus	20	0	10260	5528	2432	S	0.3	0.0	0:02.37	dbus-broker
1094	root	20	0	568316	4520	2432	S	0.3	0.0	0:05.49	VBoxDRMClient

Рис. 10: Проверка процесса yes после закрытия терминала

Запустим ещё три программы `yes` в фоновом режиме с подавлением потока вывода. Закроем один из них с помощью PID и номера задачи:

- `kill -9 %2`
- `kill -9 4437`

```
[dosergeev@dosergeev ~]$ yes > /dev/null &
[1] 4415
[dosergeev@dosergeev ~]$
[dosergeev@dosergeev ~]$ yes > /dev/null &
[2] 4424
[dosergeev@dosergeev ~]$

[dosergeev@dosergeev ~]$
[dosergeev@dosergeev ~]$ yes > /dev/null &
[3] 4437
[dosergeev@dosergeev ~]$ kill -9 2
bash: kill: (2) - Operation not permitted
[dosergeev@dosergeev ~]$ sudo kill -9 2
[sudo] password for dosergeev:
[dosergeev@dosergeev ~]$ jobs
[1]  Running                  yes > /dev/null &
[2]-  Running                  yes > /dev/null &
[3]+  Running                  yes > /dev/null &
[dosergeev@dosergeev ~]$ kill -9 %2
[2]-  Killed                   yes > /dev/null
[dosergeev@dosergeev ~]$ jobs
[1]-  Running                  yes > /dev/null &
[3]+  Running                  yes > /dev/null &
[dosergeev@dosergeev ~]$ kill -9 4437
[3]+  Killed                   yes > /dev/null
[dosergeev@dosergeev ~]$ jobs
[1]+  Running                  yes > /dev/null &
[dosergeev@dosergeev ~]$
```

Рис. 11: Завершение процесса с помощью PID и номера

Попробуем послать сигнал SIGHUP двум запущенным процессам. Процесс с номером 1 завершился с статусом Hangup, а процесс 4437, ранее запущенный в другом терминале, продолжил свою работу.

```
[dosergeev@dosergeev ~]$ jobs
[1]+  Running                  yes > /dev/null &
[dosergeev@dosergeev ~]$ kill -1 %1
[1]+  Hangup                   yes > /dev/null
[dosergeev@dosergeev ~]$ jobs
[dosergeev@dosergeev ~]$ ps aux | grep yes
doserge+  4343 99.7  0.0 220948 1792 ?        R   19:56   3:40 yes
doserge+  4500  0.0  0.0 221664 2432 pts/0    S+  20:00   0:00 grep --color=auto yes
[dosergeev@dosergeev ~]$ kill -1 4343
[dosergeev@dosergeev ~]$ ps aux | grep yes
doserge+  4343 99.8  0.0 220948 1792 ?        R   19:56   3:58 yes
doserge+  4510  0.0  0.0 221664 2432 pts/0    S+  20:00   0:00 grep --color=auto yes
```

Рис. 12: Попробуем послать сигнал 1 (SIGHUP)

Запустим ещё несколько программ yes. Завершим их работу одной командой: killall yes.

```
[dosergeev@dosergeev ~]$ yes > /dev/null &
[1] 4520
[dosergeev@dosergeev ~]$
[dosergeev@dosergeev ~]$ yes > /dev/null &
[2] 4542
[dosergeev@dosergeev ~]$ killall yes
[1]- Terminated          yes > /dev/null
[2]+ Terminated          yes > /dev/null
[dosergeev@dosergeev ~]$ ps aux | grep yes
doserge+  4554  0.0  0.0 221664  2432 pts/0    S+   20:01   0:00 grep --color=auto yes
[dosergeev@dosergeev ~]$
```

Рис. 13: Завершение группы процессов

```
[dosergeev@dosergeev ~]$ nice -n 5 yes > /dev/null &
[1] 4565
[dosergeev@dosergeev ~]$ yes > /dev/null &
[2] 4571
[dosergeev@dosergeev ~]$ ps axl | grep yes
0 1000 4565 4380 25 5 220948 1792 - RN pts/0 0:27 yes
0 1000 4571 4380 20 0 220948 1792 - R pts/0 0:13 yes
0 1000 4577 4380 20 0 221664 2432 pipe_r S+ pts/0 0:00 grep --color=auto yes
[dosergeev@dosergeev ~]$ renice -n 5 4571
4571 (process ID) old priority 0, new priority 5
[dosergeev@dosergeev ~]$ ps axl | grep yes
0 1000 4565 4380 25 5 220948 1792 - RN pts/0 1:50 yes
0 1000 4571 4380 25 5 220948 1792 - RN pts/0 1:36 yes
0 1000 4589 4380 20 0 221664 2432 pipe_r S+ pts/0 0:00 grep --color=auto yes
[dosergeev@dosergeev ~]$
```

Рис. 14: Абсолютный и относительный приоритет процессов yes

Ответы на контрольные вопросы

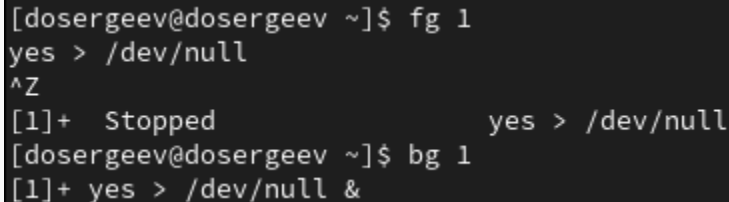
1. Какая команда даёт обзор всех текущих заданий оболочки?

- Можно использовать команду `jobs`

```
[dosergeev@dosergeev ~]$ yes > /dev/null &
[1] 2959
[dosergeev@dosergeev ~]$ yes > /dev/null &
[2] 2964
[dosergeev@dosergeev ~]$ yes > /dev/null &
[3] 2969
[dosergeev@dosergeev ~]$
[dosergeev@dosergeev ~]$ jobs
[1]  Running                  yes > /dev/null &
[2]-  Running                  yes > /dev/null &
[3]+  Running                  yes > /dev/null &
[dosergeev@dosergeev ~]$
```

Рис. 15: Выполнение `jobs`

2. Как остановить текущее задание оболочки, чтобы продолжить его выполнение в фоновом режиме?
- Переходим на текущее задание: `fg <номер>`
 - Используем сочетание клавиш `Ctrl+Z`, чтобы остановить задание
 - Переводим его в фоновой режим: `bg <номер>`



```
[dosergeev@dosergeev ~]$ fg 1
yes > /dev/null
^Z
[1]+  Stopped                  yes > /dev/null
[dosergeev@dosergeev ~]$ bg 1
[1]+ yes > /dev/null &
```

Рис. 16: Переход в фоновой режим

3. Какую комбинацию клавиш можно использовать для отмены текущего задания оболочки?
- Можно использовать сочетание Ctrl+Z

```
[dosergeev@dosergeev ~]$ jobs
[1]  Running                  yes > /dev/null &
[2]-  Running                  yes > /dev/null &
[3]+  Running                  yes > /dev/null &
[dosergeev@dosergeev ~]$ fg 2
yes > /dev/null
^C
[dosergeev@dosergeev ~]$ jobs
[1]-  Running                  yes > /dev/null &
[3]+  Running                  yes > /dev/null &
[dosergeev@dosergeev ~]$
```

Рис. 17: Отмена текущего задания

4. Необходимо отменить одно из начатых заданий. Доступ к оболочке, в которой в данный момент работает пользователь, невозможен. Что можно сделать, чтобы отменить задание?
- Откроем новую оболочку под учётной записью root. Узнаем PID запущенных процессов другой оболочки. Пропишем команду kill для каждого процесса, либо killall -9 <процесс>

```
[root@dosergeev ~]# ps fax | grep bash -B2
 2854 ?        Ssl    0:00  \_ /usr/libexec/xdg-desktop-portal-gtk
 2889 ?        Ssl    0:01  \_ /usr/libexec/gnome-terminal-server
 2907 pts/0    Ss+    0:00  |   \_ bash
 2959 pts/0    R      2:31  |   |   \_ yes
 2969 pts/0    R      2:36  |   |   \_ yes
 3028 pts/1    Ss     0:00  |   \_ bash
 3061 pts/1    S      0:00  |       \_ su -
 3072 pts/1    S      0:00  |           \_ -bash
 3109 pts/1    R+     0:00  |               \_ ps fax
 3110 pts/1    S+     0:00  |                   \_ grep --color=auto bash -B2
[root@dosergeev ~]# kill -9 2959
[root@dosergeev ~]# kill -9 2969
[root@dosergeev ~]# ps aux | grep yes
root      3114  0.0  0.0 221664 2432 pts/1    S+   21:36   0:00 grep --color=auto yes
[root@dosergeev ~]#
```

5. Какая команда используется для отображения отношений между родительскими и дочерними процессами?
- Используется команда `ps fax`.

```
[root@dosergeev ~]# ps fax | grep bash -B2
 2854 ?      Ssl    0:00  \_ /usr/libexec/xdg-desktop-portal-gtk
 2889 ?      Ssl    0:01  \_ /usr/libexec/gnome-terminal-server
 2907 pts/0    Ss+    0:00  |   \_ bash
 2959 pts/0    R      2:31  |   |   \_ yes
 2969 pts/0    R      2:36  |   |   \_ yes
 3028 pts/1    Ss     0:00  |   \_ bash
 3061 pts/1    S      0:00  |       \_ su -
 3072 pts/1    S      0:00  |       \_ -bash
 3109 pts/1    R+     0:00  |           \_ ps fax
 3110 pts/1    S+     0:00  |           \_ grep --color=auto bash -B2
```

Рис. 19: Пример с `ps fax`

6. Какая команда позволит изменить приоритет процесса с идентификатором 1234 на более высокий?
- `renice -n <новый приоритет до 19> <процесс или PID>`
7. В системе в настоящее время запущено 20 процессов dd. Как проще всего остановить их все сразу?
- `killall -9 dd`

8. Какая команда позволяет остановить команду с именем `myscommand`?
 - `kill myscommand`
9. Какая команда используется в `top`, чтобы убить процесс?
 - Используем `k`, затем номер процесса и сигнал 9.
10. Как запустить команду с достаточно высоким приоритетом, не рискуя, что не хватит ресурсов для других процессов?
 - `nice -n <приоритет> <команда>`

Вывод

В результате выполнения лабораторной работы я получил навыки управления задачами и процессами. Узнал как менять режим работы процессов (фоновой и передний), приостанавливать и завершать текущие процессы, запущенные в переднем режиме, изменять их приоритеты с помощью команд `nice` и `renice`, а также научился запускать процессы так, чтобы они работали даже после закрытия текущей оболочки.