

Министерство образования и науки Российской Федерации  
Федеральное государственное автономное образовательное учреждение высшего образования  
«Пермский национальный исследовательский политехнический университет»  
Электротехнический факультет  
Кафедра «Информационные технологии и автоматизированные системы»

ОТЧЕТ  
по лабораторной работе №5  
Тема: «Знакомство с процессами»

Выполнил: студент группы РИС-19-16

Миннахметов Э.Ю. \_\_\_\_\_

Проверил: ст. преподаватель кафедры  
ИТАС

Шереметьев В. Г. \_\_\_\_\_

Дата \_\_\_\_\_

Пермь, 2021

## Цель работы

Познакомиться с понятием процесса. Научиться получать список имеющихся в системе процессов и управлять их состоянием.

## Ход работы

1. Прочитайте теоретический материал по лабораторной работе.
2. Ознакомьтесь с работой команд, приведенных в Таблице 2. Изучите для содержащихся в Таблице 2 команд страницы справочного руководства.
3. Создайте файл `proc1`, содержащий список процессов пользователя `root`, отсортированный по идентификатору родительского процесса. Используйте команду `ps` и изученные ранее утилиты.

```
eldarian@lester:~$ ps -fu root --sort ppid > proc1
eldarian@lester:~$ cat proc1 | head -10
UID      PID      PPID  C  STIME TTY          TIME CMD
root         1         0  0  16:07 ?        00:00:01 /sbin/init splash
root         2         0  0  16:07 ?        00:00:00 [kthreadd]
root        614         1  0  16:07 ?        00:00:00 /lib/systemd/systemd-journald
root        642         1  0  16:07 ?        00:00:00 /lib/systemd/systemd-udev
root        986         1  0  16:07 ?        00:00:00 /sbin/mount.ntfs /dev/sdb
root       1048         1  0  16:07 ?        00:00:00 /usr/lib/accountsservice/
root       1049         1  0  16:07 ?        00:00:00 /usr/sbin/acpid
root       1055         1  0  16:07 ?        00:00:00 /usr/sbin/cron -f
root       1056         1  0  16:07 ?        00:00:00 /usr/sbin/cupsd -l
eldarian@lester:~$
```

4. Получите информацию о процессах вашего пользователя, имеющих статус работоспособный.

```
eldarian@lester:~$ ps -eo stat,pid,command | grep -Ei "^R"
R      181 [kworker/7:1-events]
R      397 [kworker/u48:4+events_unbound]
Rl     3015 /usr/lib/firefox/firefox -contentproc -childID 14 -isForBrowser -prefsLen 5812 -prefMapSize 2453
10    -jsInit 278680 -parentBuildID 20211028161635 -appdir /usr/lib/firefox/browser 2415 true tab
R+     4420 ps -eo stat,pid,command
```

5. Добавьте к файлу `proc1` сведения о процессе в данный момент потребляющий большой процесс ресурсов центрального процессора.

```
eldarian@lester:~$ ps -eo pid,%cpu,%mem,command --sort=-pcpu | head -2 >> proc1
eldarian@lester:~$ cat proc1 | tail -10
root      8414         2  0  17:10 ?        00:00:00 [kworker/9:0-events]
root      8415         2  0  17:10 ?        00:00:00 [kworker/20:2-events]
root      8548         2  0  17:13 ?        00:00:00 [kworker/1:0-events]
root      8549         2  0  17:13 ?        00:00:00 [kworker/0:1-events]
root      8806         2  0  17:19 ?        00:00:00 [kworker/12:0-events]
root      8807         2  0  17:19 ?        00:00:00 [kworker/4:0-events]
root      1260       1236  3  15:32 tty1      00:03:47 /usr/lib/xorg/Xorg -nolisten tcp -auth /var/run/sddm/{ed49d8d1-29ea-41d3-bc
0c-ae5c6add6888} -background none -noreset -displayfd 17 -seat seat0 vt1
root      1744       1236  0  15:33 ?        00:00:00 /usr/lib/x86_64-linux-gnu/sddm/sddm-helper --socket /tmp/sddm-authfa7e18c7-
edae-44f9-a688-65824b4c62c7 --id 1 --start /usr/bin/startplasma-x11 --user eldarian
PID %CPU %MEM COMMAND
7186 27.9 0.7 /usr/lib/firefox/firefox -contentproc -childID 4 -isForBrowser -prefsLen 5430 -prefMapSize 245251 -jsInit 278
680 -parentBuildID 20211028161635 -appdir /usr/lib/firefox/browser 6982 true tab
eldarian@lester:~$
```

6. Запустите утилиту `top`. Изучите содержимое информационных полей, предоставляемых утилитой. Получите информацию о степени использования ресурсов системы, количестве пользователей, времени работы системы.

```
top - 17:34:45 up 2:02, 1 user, load average: 0,94, 0,99, 0,87
Tasks: 392 total, 1 running, 391 sleeping, 0 stopped, 0 zombie
%Cpu(s): 2,1 us, 1,5 sy, 0,0 ni, 96,4 id, 0,0 wa, 0,0 hi, 0,0 si, 0,0 st
MiB Mem : 64245,7 total, 56289,6 free, 3072,2 used, 4883,9 buff/cache
MiB Swap: 0,0 total, 0,0 free, 0,0 used. 60304,1 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
7186	eldarian	20	0	3043816	526652	115416	S	28,9	0,8	10:36.36	Isolated Web Co
7333	eldarian	20	0	273124	90148	55812	S	17,9	0,1	2:43.05	RDD Process
6982	eldarian	20	0	4412116	614776	344936	S	17,6	0,9	6:55.64	GeckoMain
6667	root	20	0	0	0	0	D	17,3	0,0	1:03.07	kworker/u48:0+events_unbound
1869	eldarian	20	0	2591100	126736	89928	S	4,7	0,2	5:29.53	kwin_x11
1260	root	20	0	584560	100876	68256	S	4,0	0,2	4:11.92	Xorg
8147	eldarian	20	0	2888076	297984	140652	S	4,0	0,5	1:12.71	Isolated Web Co
4970	eldarian	20	0	57,0g	248548	117768	S	2,3	0,4	2:51.39	Discord
4917	eldarian	20	0	686028	102148	66744	S	2,0	0,2	1:59.29	Discord
7266	eldarian	20	0	2511708	200640	109512	S	1,3	0,3	0:45.94	Isolated Web Co
4677	eldarian	20	0	36,7g	158604	99928	S	1,0	0,2	0:47.55	Discord
1753	eldarian	9	-11	1595472	20740	15892	S	0,7	0,0	0:57.84	pulseaudio
7190	eldarian	20	0	2488436	144920	98660	S	0,7	0,2	0:17.10	Isolated Web Co
8382	eldarian	20	0	1595776	155136	97564	S	0,7	0,2	1:54.07	plasma-systemmo
223	root	20	0	0	0	0	I	0,3	0,0	0:06.00	kworker/10:1-events
1053	root	20	0	6980	2768	1892	S	0,3	0,0	0:00.63	mount.ntfs
1120	message+	20	0	9084	6172	4012	S	0,3	0,0	0:03.71	dbus-daemon
5906	root	20	0	0	0	0	I	0,3	0,0	0:03.66	kworker/23:2-events
6585	root	20	0	0	0	0	I	0,3	0,0	0:02.08	kworker/11:0-events
6871	root	20	0	0	0	0	I	0,3	0,0	0:02.10	kworker/2:0-events
7054	eldarian	20	0	191928	42416	35012	S	0,3	0,1	0:00.62	Socket Process
7479	eldarian	20	0	1025972	103720	83636	S	0,3	0,2	0:11.54	konsole
7664	eldarian	20	0	2408052	98068	82240	S	0,3	0,1	0:01.32	Isolated Web Co

Степень использования ресурсов системы – 0.96, 0.99, 0.87.

Количество пользователей – 1.

Время работы системы – 2 часа 2 минуты.

7. Ознакомьтесь со справочной информацией утилиты `top`. Изучите действия, выполняемые по нажатию на клавиши, «u», «f», «R», «d», «L». Настройте вывод утилиты `top` в соответствии с заданием 3, с периодом обновления 1с.

**u** — выбор пользователя, запустившего процесс.

**f** — выбор выводимых свойств процессов (в окне **d** — выбрать/убрать, **s** — выбрать в качестве сортируемого поля).

**R** — обратная сортировка.

**d** — настройка продолжительности показа состояния перед обновлением.

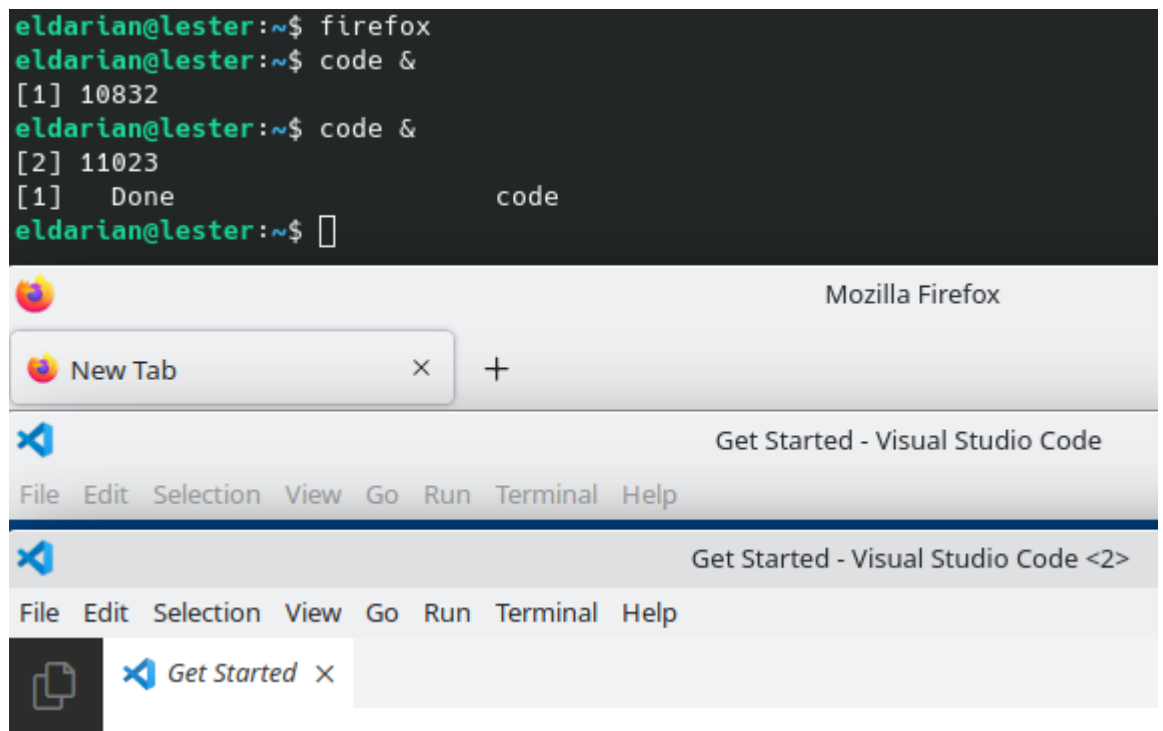
**L** — поиск в выводе состояния.

8. В окне терминала запустите программу firefox. Используя команду pstree получите информацию о дереве процессов и их идентификаторах. Найдите поддерево для процесса firefox, изучите список составляющих его процессов.

```
GeckoMain—Isolated Web Co—37*[{Isolated Web Co}]
          —3*[{Isolated Web Co—29*[{Isolated Web Co}]]
          —Isolated Web Co—27*[{Isolated Web Co}]
          —6*[{Isolated Web Co—26*[{Isolated Web Co}]]
          —Isolated Web Co—43*[{Isolated Web Co}]
          —Privileged Cont—26*[{Privileged Cont}]
          —RDD Process—10*[{RDD Process}]
          —Socket Process—4*[{Socket Process}]
          —3*[{Web Content—18*[{Web Content}]]
          —WebExtensions—26*[{WebExtensions}]
          —141*[{GeckoMain}]
```

9. В новом окне терминала запустите программу firefox. Затем два раза запустите утилиту gedit фоновом режиме.

Вместо программы gedit запускаю другой текстовый редактор с графическим интерфейсом – VS Code.



10.Получите список заданий текущей сессии терминала.

```
eldarian@lester:~$ firefox &  
[1] 11235  
eldarian@lester:~$ code &  
[2] 11250  
[1] Done                                firefox  
eldarian@lester:~$ code &  
[3] 11319  
[2] Done                                code  
eldarian@lester:~$ jobs  
[3]+ Done                                code  
eldarian@lester:~$
```

11. Используя команды `fg` и `bg` и сочетания клавиш «Ctrl + z» и «Ctrl + c» научитесь перемещать задачи из фона на передний план и наоборот.

Последовательность команд:

***top*** – запуск утилиты на переднем плане

«***Ctrl + Z***» - остановка процесса и отправка на фон

***fg*** – вернуть на передний план и возобновить работу

«***Ctrl + Z***» - остановка процесса и отправка на фон

***bg*** – возобновить работу в фоновом режиме

12. Получите список сигналов для команды kill. Завершите запущенные процессы с помощью команды kill с помощью сигналов SIGKILL и SIGTERM.

```
eldarian@lester:~$ kill -l
 1) SIGHUP      2) SIGINT      3) SIGQUIT     4) SIGILL      5) SIGTRAP
 6) SIGABRT     7) SIGBUS     8) SIGFPE      9) SIGKILL     10) SIGUSR1
11) SIGSEGV    12) SIGUSR2    13) SIGPIPE    14) SIGALRM     15) SIGTERM
16) SIGSTKFLT  17) SIGCHLD    18) SIGCONT    19) SIGSTOP     20) SIGTSTP
21) SIGTTIN    22) SIGTTOU    23) SIGURG     24) SIGXCPU     25) SIGXFSZ
26) SIGVTALRM  27) SIGPROF    28) SIGWINCH   29) SIGIO       30) SIGPWR
31) SIGSYS     34) SIGRTMIN   35) SIGRTMIN+1 36) SIGRTMIN+2 37) SIGRTMIN+3
38) SIGRTMIN+4 39) SIGRTMIN+5 40) SIGRTMIN+6 41) SIGRTMIN+7 42) SIGRTMIN+8
43) SIGRTMIN+9 44) SIGRTMIN+10 45) SIGRTMIN+11 46) SIGRTMIN+12 47) SIGRTMIN+13
48) SIGRTMIN+14 49) SIGRTMIN+15 50) SIGRTMAX-14 51) SIGRTMAX-13 52) SIGRTMAX-12
53) SIGRTMAX-11 54) SIGRTMAX-10 55) SIGRTMAX-9  56) SIGRTMAX-8  57) SIGRTMAX-7
58) SIGRTMAX-6  59) SIGRTMAX-5  60) SIGRTMAX-4  61) SIGRTMAX-3  62) SIGRTMAX-2
63) SIGRTMAX-1  64) SIGRTMAX

eldarian@lester:~$ top &
[2] 13282
eldarian@lester:~$ kill -SIGKILL 13282

[2]+  Stopped                  top
eldarian@lester:~$ top &
[3] 13283
[2]  Killed                  top
eldarian@lester:~$ kill -SIGTERM 13283

[3]+  Stopped                  top
eldarian@lester:~$
```

13. Выполните команду killall -u имя\_текущего\_пользователя.

После команды все программы закрылись и графическая оболочка перезагрузилась.

## Ответы на контрольные вопросы

1. Какие способы получения информации о процессах в системе вы знаете?  
*ps, top, pstree*

2. Как можно управлять выводом утилиты top?

*u* — выбор пользователя, запустившего процесс.

*f* — выбор выводимых свойств процессов (в окне *d* — выбрать/убрать, *s* — выбрать в качестве сортируемого поля).

*R* — обратная сортировка.

*d* — настройка продолжительности показа состояния перед обновлением.

*L* — поиск в выводе состояния.

3. Какие сигналы отправляются сочетаниями клавиш «Ctrl + z» и «Ctrl + c»?

«*Ctrl + Z*» - остановка процесса и отправка на фон.

«*Ctrl + C*» - завершение процесса.

4. Какой процесс в системе запускается первым?

*systemd*

5. Какие типы процессов вы знаете?

Системные процессы — часть ядра и всегда находятся в оперативной памяти.

Демоны — фоновые приложения, не требующие графическую оболочку. Настраивается автозапуск утилитой *systemctl*.

Прикладные процессы — графическая оболочка, консольные утилиты и программы.

6. Что такое дескриптор процесса?

Номер процесса, на который можно ссылаться (*PID*).