

Практика 3, Раздел 1

1. Войдите под пользователем user1 из практики 2 (su - user1)

```
root@eltex-practice2-pg1-v21:~# su user1
```

2. Подсчитайте количество процессов, имеющих несколько потоков

выполнения

```
user1@eltex-practice2-pg1-v21:/root$ ps -eLf | awk '{print $2}' | sort | uniq -c | awk '$1 > 1 {print $0}' | wc -l  
10
```

3. Запустите top и настройте вывод полей с информацией о процессе следующим образом:

- удалите поля VIRT, RES, SHR;
- добавьте поле RUSER и сделайте так, чтобы это поле было показано после поля USER;

```
Fields Management for window 1:Def, whose current sort field is %CPU  
Navigate with Up/Dn, Right selects for move then <Enter> or Left commits,  
'd' or <Space> toggles display, 's' sets sort. Use 'q' or <Esc> to end!  
  
* PID      = Process Id          SWAP    = Swapped Size (KiB)    RSlk     = RES Locked (KiB)  
* USER     = Effective User Name CODE    = Code Size (KiB)      RSsh     = RES Shared (KiB)  
* RUSER    = Real User Name    DATA   = Data+Stack (KiB)    CGNAME   = Control Group name  
* PR       = Priority          nMaj    = Major Page Faults   NU       = Last Used NUMA node  
* NI       = Nice Value        nMin    = Minor Page Faults  LOGID    = Login User Id  
* VIRT     = Virtual Image (KiB) nDRT    = Dirty Pages Count   EXE      = Executable Path  
* RES      = Resident Size (KiB) WCHAN   = Sleeping in Function RSS       = Res Mem (smaps), KiB  
* SHR      = Shared Memory (KiB) Flags    = Task Flags <sched.h> PSS       = Proportion RSS, KiB  
* S        = Process Status    CGROUPS  = Control Groups    PSan     = Proportion Anon, KiB  
* %CPU     = CPU Usage          SUPGIDS  = Supp Groups IDs  PSfd     = Proportion File, KiB  
* %MEM     = Memory Usage (RES) SUPGRPS  = Supp Groups Names PSsh     = Proportion Shrd, KiB  
* TIME+    = CPU Time, hundredths TGID     = Thread Group Id  USS      = Unique RSS, KiB  
* COMMAND  = Command Name/Line OOMa     = OOMEM Adjustment  ioR      = I/O Bytes Read  
* PPID     = Parent Process pid OOMS     = OOMEM Score current ioRop    = I/O Read Operations  
* UID      = Effective User Id  ENVIRON  = Environment vars ioW      = I/O Bytes Written  
* RUID     = Real User Id       vmj      = Major Faults delta ioWop    = I/O Write Operations  
* SUID     = Saved User Id       vMn     = Minor Faults delta AGID     = Autogroup Identifier  
* SUSER    = Saved User Name     USED     = Res+Swap Size (KiB) AGNI     = Autogroup Nice Value  
* GID      = Group Id           nsIPC    = IPC namespace Inode STARTED  = Start Time from boot  
* GROUP    = Group Name         nsMNT    = MNT namespace Inode ELAPSED  = Elapsed Running Time  
* PGPR     = Process Group Id   nsNET    = NET namespace Inode %CUU    = CPU Utilization  
* TTY      = Controlling Tty     nsPID    = PID namespace Inode %CUC    = Utilization + child  
* TPGID    = Tty Process Grp Id nsUSER   = USER namespace Inode nsCGROUP = CGRP namespace Inode  
* SID      = Session Id         nsUTS    = UTS namespace Inode nsTIME   = TIME namespace Inode  
* nTH      = Number of Threads  LXC      = LXC container name  
* P        = Last Used Cpu (SMP) RSan     = RES Anonymous (KiB)  
* TIME     = CPU Time           RSfd     = RES File-based (KiB)
```

PID	USER	RUSER	PR	NI	S	%CPU	%MEM	TIME+	COMMAND
38729	root	root	20	0	S	0.0	0.1	0:48.91	qemu-ga
44372	root	root	20	0	I	0.0	0.0	0:00.17	kworker/0:1-events
44387	root	root	20	0	I	0.0	0.0	0:00.11	kworker/1:0-events
35	root	root	20	0	S	0.0	0.0	0:05.18	kcompactd0
20369	root	root	rt	0	S	0.0	0.7	0:06.85	multipathd
37587	root	root	20	0	S	0.0	0.4	0:01.37	udisksd
44359	root	root	20	0	I	0.0	0.0	0:00.20	kworker/u4:3-events_unbound
44374	root	root	20	0	I	0.0	0.0	0:00.10	kworker/u4:1-events_unbound
44401	root	root	20	0	S	0.0	0.3	0:00.09	sshd
44598	root	root	20	0	S	0.0	0.2	0:00.02	sudo
44626	root	root	20	0	I	0.0	0.0	0:00.01	kworker/u4:0-events_unbound
44627	user1	user1	20	0	R	0.0	0.2	0:00.02	top
1	root	root	20	0	S	0.0	0.4	0:06.95	systemd
2	root	root	20	0	S	0.0	0.0	0:00.02	kthreadd
3	root	root	20	0	S	0.0	0.0	0:00.00	pool_workqueue_release
4	root	root	0	-20	I	0.0	0.0	0:00.00	kworker/R-rcu_g
5	root	root	0	-20	I	0.0	0.0	0:00.00	kworker/R-rcu_p
6	root	root	0	-20	I	0.0	0.0	0:00.00	kworker/R-slub
7	root	root	0	-20	I	0.0	0.0	0:00.00	kworker/R-netns
10	root	root	0	-20	I	0.0	0.0	0:00.52	kworker/0:0H-kblockd
12	root	root	0	-20	I	0.0	0.0	0:00.00	kworker/R-mm_pe
13	root	root	20	0	I	0.0	0.0	0:00.00	rcu_tasks_kthread
14	root	root	20	0	I	0.0	0.0	0:00.00	rcu_tasks_rude_kthread
15	root	root	20	0	I	0.0	0.0	0:00.00	rcu_tasks_trace_kthread

4. В другом терминальном окне выполните команду `passwd` и оставьте ее в состоянии запроса текущего пароля

```
root@eltex-practice2-pg1-v21:~# passwd
New password: 
```

5. Перейдите в терминальное окно с `top` и выполните следующие действия:

- выведите все процессы, для которых реальным пользователем является пользователь, которым вы вошли в сеанс;

```
top - 10:26:19 up 2 days, 2:34, 2 users, load average: 0.08, 0.03, 0.01
Tasks: 116 total, 1 running, 113 sleeping, 2 stopped, 0 zombie
%Cpu(s): 0.1 us, 0.1 sy, 0.0 ni, 99.8 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 3598.2 total, 1844.4 free, 507.7 used, 1537.9 buff/cache
MiB Swap: 3185.0 total, 3185.0 free, 0.0 used. 3090.5 avail Mem
```

PID	USER	RUSER	PR	NI	S	%CPU	%MEM	TIME+	COMMAND
44601	user1	user1	20	0	S	0.0	0.1	0:00.00	bash
44614	user1	user1	20	0	T	0.0	0.2	0:00.03	top
44621	user1	user1	20	0	T	0.0	0.2	0:00.26	top
44627	user1	user1	20	0	R	0.0	0.2	0:00.13	top

- найдите процесс, запущенный командой `passwd`;

```
top - 10:30:19 up 2 days, 2:38, 2 users, load average: 0.00, 0.00, 0.00
Tasks: 119 total, 1 running, 113 sleeping, 5 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.0 sy, 0.0 ni, 98.0 id, 1.9 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 3597.7 total, 1840.8 free, 510.9 used, 1538.0 buff/cache
MiB Swap: 3185.0 total, 3185.0 free, 0.0 used. 3086.8 avail Mem
```

PID	USER	PR	NI	S	%CPU	%MEM	TIME+	COMMAND
44708	root	20	0	S	0.0	0.1	0:00.00	passwd

- отправьте этому процессу сигналы 15 (SIGTERM), 2 (SIGINT), 3(SIGQUIT), 9(SIGKILL)

```
top - 10:40:07 up 2 days, 2:48, 2 users, load average: 0.00, 0.00, 0.00
Tasks: 120 total, 1 running, 113 sleeping, 6 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.1 sy, 0.0 ni, 99.9 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 3595.9 total, 1838.2 free, 511.5 used, 1538.0 buff/cache
MiB Swap: 3185.0 total, 3185.0 free, 0.0 used. 3084.3 avail Mem
```

PID	USER	RUSER	PR	NI	S	%CPU	%MEM	TIME+	COMMAND
-----	------	-------	----	----	---	------	------	-------	---------

```
root@eltex-practice2-pg1-v21:~# passwd
New password: Killed
```

6. Выполните команду `vim ~/file_task3.txt` и нажмите `Ctrl-Z`

```
user1@eltex-practice2-pg1-v21:/root$ vim ~/file task3.txt
```

```
2 files to edit
[8]+  Stopped                  vim ~/file task3.txt
```

- ```
user1@eltex-practice2-pg1-v21:/root$ sleep 600
^Z
[9]+ Stopped sleep 600
user1@eltex-practice2-pg1-v21:/root$ jobs
[1] Stopped top
[2] Stopped top
[3] Stopped top
[4] Stopped top
[5] Stopped top
[6] Stopped top
[7] Stopped sudo top
[8]- Stopped vim ~/file task3.txt
[9]+ Stopped sleep 600
```

- ```
user1@eltex-practice2-pg1-v21:/root$ sleep 600 &
[11] 44740
```

9. Измените число NICE у задания (sleep 600), сделав его равным 10

```

user1@eltex-practice2-pg1-v21:/root$ jobs -l
[1]  44614 Stopped (signal)          top
[2]  44621 Stopped (signal)          top
[3]  44627 Stopped (signal)          top
[4]  44709 Stopped (signal)          top
[5]  44711 Stopped (signal)          top
[6]  44712 Stopped (signal)          top
[7]  44717 Stopped (signal)          sudo top
[8]  44730 Stopped                  vim ~/file task3.txt
[9]- 44733 Stopped                  sleep 600
[10]+ 44739 Stopped                  sleep 600
[11]  44740 Running                  sleep 600 &
user1@eltex-practice2-pg1-v21:/root$ renice 10 44740
44740 (process ID) old priority 0, new priority 10

```

10. Проверьте, что число NICE у этого задания изменилось

```

user1@eltex-practice2-pg1-v21:/root$ ps -o pid,ni,cmd -p $(jobs -p)
  PID  NI  CMD
  44614   0  top
  44621   0  top
  44627   0  top
  44709   0  top
  44711   0  top
  44712   0  top
  44717   0  sudo top
  44730   0  vim /home/user1/file task3.txt
  44733   0  sleep 600
  44739   0  sleep 600
  44740  10  sleep 600

```

11. Сделайте задание vim ~/file_task3.txt активным и выйдите из редактора

```

user1@eltex-practice2-pg1-v21:/root$ vim ~/file_task3.txt
[12]+  Stopped                  vim ~/file_task3.txt

```

12. Отправьте сигнал 15 (SIGTERM) заданию sleep 600 и выполните команду jobs

```

user1@eltex-practice2-pg1-v21:/root$ kill -15 44740
bash: kill: (44740) - No such process
[11]  Terminated                  sleep 600
user1@eltex-practice2-pg1-v21:/root$ jobs
[1]  Stopped                  top
[2]  Stopped                  top
[3]  Stopped                  top
[4]  Stopped                  top
[5]  Stopped                  top
[6]  Stopped                  top
[7]  Stopped                  sudo top
[8]  Stopped                  vim ~/file task3.txt
[9]  Stopped                  sleep 600
[10]- Stopped                  sleep 600
[12]+ Stopped                  vim ~/file task3.txt

```

13. Создайте перехватчик сигналов SIGINT и SIGQUIT внутри командного интерпретатора, который выводит сообщение «Меня голыми руками не возьмёшь!» (используйте встроенную команду trap) и отправьте сигналы самому себе

```
user1@eltex-practice2-pg1-v21:/root$ trap 'echo "Меня голыми руками не возьмёшь!"' SIGINT SIGQUIT
user1@eltex-practice2-pg1-v21:/root$ trap 'echo "Меня голыми руками не возьмёшь!"' SIGINT SIGQUIT
user1@eltex-practice2-pg1-v21:/root$ kill -2 $$
Меня голыми руками не возьмёшь!
user1@eltex-practice2-pg1-v21:/root$ kill -3 $$
Меня голыми руками не возьмёшь!
```