

3 Практика 1 раздел

1

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

```
passwd: password updated successfully
root@eltex-practice2-pg1-v5:~# su - user1
user1@eltex-practice2-pg1-v5:~$
```

2

Подсчитайте количество процессов, имеющих несколько потоков

выполнения

```
root@eltex-practice2-pg1-v5:~# su - user1
user1@eltex-practice2-pg1-v5:~$ ps -eo pid,nlwp | awk '$2 > 1 {count++} END {print count}'
11
```

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;

```
'd' or <Space> toggles display, 's' sets sort. Use 'q' or <Esc> to end!

* PID      = Process Id          ioW      = I/O Bytes Written
* RUSER    = Real User Name      ioWop    = I/O Write Operations
* USER     = Effective User Name AGID      = Autogroup Identifier
* PR       = Priority           AGNI      = Autogroup Nice Value
* NI       = Nice Value         STARTED   = Start Time from boot
VIRT       = Virtual Image (KiB) ELAPSED   = Elapsed Running Time
RES        = Resident Size (KiB) %CUU      = CPU Utilization
SHR        = Shared Memory (KiB) %CUC      = Utilization + child
* S        = Process Status     nsCGROUP  = CGRP namespace Inode
* %CPU     = CPU Usage          nsTIME   = TIME namespace Inode
* %MEM     = Memory Usage (RES)
* TIME+    = CPU Time, hundredths
* COMMAND  = Command Name/Line
PPID      = Parent Process pid
UID       = Effective User Id
RUID      = Real User Id
SUID      = Saved User Id
```

4

В другом терминальном окне выполните команду passwd и оставьте ее в

состоянии запроса текущего пароля

```
> user1@eltex-practice2-pg1-v5: ~
Tasks: 115 total, 1 running, 114 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.0 sy, 0.0 ni, 100.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 22.1/2411.4 [|||||||||||||||||]
MiB Swap: 0.0/3185.0 [ ]

  PID RUSER   USER      PR  NI  S   %CPU %MEM    TIME+  COMMAND
  67259 user1    user1     20   0  S    0.0  0.2    0:00.01 bash

System Resource Required

Last login: Sat Oct 25 09:46:08 2025 from 172.16.8.4
root@eltex-practice2-pg1-v5:~# su - user1
user1@eltex-practice2-pg1-v5:~$ passwd
Changing password for user1.
Current 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;
- найдите процесс, запущенный командой passwd;
- отправьте этому процессу сигналы 15 (SIGTERM), 2 (SIGINT), 3 (SIGQUIT), 9(SIGKILL)

```
Last login: Sat Oct 25 09:58:27 2025 from 172.16.8.4
root@eltex-practice2-pg1-v5:~# PID=$(ps aux | grep passwd | grep -v grep | awk '{print $2}')
root@eltex-practice2-pg1-v5:~# echo "PID процесса passwd: $PID"
PID процесса passwd: 67269
root@eltex-practice2-pg1-v5:~#

root@eltex-practice2-pg1-v5:~# echo "PID процесса passwd: $PID"
PID процесса passwd: 67269
root@eltex-practice2-pg1-v5:~# echo "Отправка SIGTERM (15)..."
Отправка SIGTERM (15)...
root@eltex-practice2-pg1-v5:~# kill -15 $PID
root@eltex-practice2-pg1-v5:~# sleep 2
root@eltex-practice2-pg1-v5:~# echo "Отправка SIGINT (2)..."
Отправка SIGINT (2)...
root@eltex-practice2-pg1-v5:~# kill -2 $PID
root@eltex-practice2-pg1-v5:~# sleep 2
root@eltex-practice2-pg1-v5:~# echo "Отправка SIGQUIT (3)..."
Отправка SIGQUIT (3)...
root@eltex-practice2-pg1-v5:~# kill -3 $PID
root@eltex-practice2-pg1-v5:~# sleep 2

root@eltex-practice2-pg1-v5:~#
root@eltex-practice2-pg1-v5:~# echo "Отправка SIGKILL (9)..."
Отправка SIGKILL (9)...
root@eltex-practice2-pg1-v5:~# kill -9 $PID
root@eltex-practice2-pg1-v5:~#
```

```

root@eltex-practice2-pg1-v5:~# su - user1
user1@eltex-practice2-pg1-v5:~$ passwd
Changing password for user1.
Current password: Killed
user1@eltex-practice2-pg1-v5:~$

```

```

user1@eltex-practice2-pg1-v5: ~
Tasks: 120 total, 1 running, 119 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.0 sy, 0.0 ni, 100.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 21.6/2389.9 [|||||||||||||||||]
MiB Swap: 0.0/3185.0 [

```

PID	RUSER	USER	PR	NI	S	%CPU	%MEM	TIME+	COMMAND
67259	user1	user1	20	0	S	0.0	0.2	0:00.01	bash

6

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

```

user1@eltex-practice2-pg1-v5:~$ vim ~/file_task3.txt
[1]+  Stopped                  vim ~/file_task3.txt
user1@eltex-practice2-pg1-v5:~$

```

7

Выполните команду `sleep 600`, нажмите Ctrl-Z и выполните команду `jobs`

```

user1@eltex-practice2-pg1-v5:~$ sleep 600
^Z
[2]+  Stopped                  sleep 600
user1@eltex-practice2-pg1-v5:~$ jobs
[1]-  Stopped                  vim ~/file_task3.txt
[2]+  Stopped                  sleep 600
user1@eltex-practice2-pg1-v5:~$

```

8

Последнее задание (`sleep 600`) сделайте фоновым

```

user1@eltex-practice2-pg1-v5:~$ bg %2
[2]+  sleep 600 &
user1@eltex-practice2-pg1-v5:~$ jobs
[1]+  Stopped                  vim ~/file_task3.txt
[2]-  Running                  sleep 600 &
user1@eltex-practice2-pg1-v5:~$

```

9

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

```

[2]-  Running                  sleep 600 &
user1@eltex-practice2-pg1-v5:~$ renice -n 10 -p $(jobs -p %2)
67385 (process ID) old priority 0, new priority 10
user1@eltex-practice2-pg1-v5:~$

```

10 Проверьте, что число NICE у этого задания изменилось

```

user1@eltex-practice2-pg1-v5:~$ ps -o pid,ni,comm -p $(jobs -p %2)
  PID  NI COMMAND
  67385  10 sleep
user1@eltex-practice2-pg1-v5:~$

```

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

```
user1@eltex-practice2-pg1-v5:~$ fg %1
vim ~/file_task3.txt
```

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

```
user1@eltex-practice2-pg1-v5:~$ kill -15 $(jobs -p %2)
user1@eltex-practice2-pg1-v5:~$ jobs
[2]+  Terminated                  sleep 600
user1@eltex-practice2-pg1-v5:~$
```

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

```
user1@eltex-practice2-pg1-v5:~$
Не взломаешьpractice2-pg1-v5:~$ ^C
```

самому себе

```
user1@eltex-practice2-pg1-v5:~$ kill -SIGINT $$
Не взломаешь
user1@eltex-practice2-pg1-v5:~$
```

```
Не взломаешь
user1@eltex-practice2-pg1-v5:~$ kill -SIGQUIT $$
Не взломаешь
user1@eltex-practice2-pg1-v5:~$
```

2 Раздел

1

Создайте скрипт на языке bash с именем template_task.sh, делающий следующее:

-

При запуске проверяет, что имя скрипта не совпадает с template_task.sh, если совпадает - выходит с уведомлением «я бригадир, сам не работаю»

-

При запуске дописывает в файл report_имя_скрипта_без_полного_пути.log в рабочем каталоге информацию: [PID] ДАТА ВРЕМЯ Скрипт запущен

-

Генерирует случайное число от 30 до 1800 и ждет такое количество секунд

-

Дописывает в файл report_имя_скрипта_без_полного_пути.log сообщение:

[PID] ДАТА ВРЕМЯ Скрипт завершился, работал N минут

```
user1@eltex-practice2-pg1-v5:~$ ./template_task.sh
я бригадир, сам не работаю
user1@eltex-practice2-pg1-v5:~$
```

```

user1@eltex-practice2-pg1-v5: $ user1@eltex-practice2-pg1-v5: $
user1@eltex-practice2-pg1-v5: $ ./worker1.sh
Скрипт будет работать 64 секунд
Скрипт завершен, информация записана в report_worker1.log
user1@eltex-practice2-pg1-v5: $

```

```

user1@eltex-practice2-pg1-v5: $ ls -la report_*.log
-rw-rw-r-- 1 user1 user1 112 Oct 25 10:39 report_.log
-rw-rw-r-- 1 user1 user1 574 Oct 25 11:17 report_worker1.log
user1@eltex-practice2-pg1-v5: $ cat report_worker1.log
[67438] 2025-10-25 10:52:28 Скрипт запущен
[67446] 2025-10-25 10:54:07 Скрипт запущен
[67477] 2025-10-25 11:01:09 Скрипт запущен
[67482] 2025-10-25 11:03:31 Скрипт запущен
[67493] 2025-10-25 11:05:19 Скрипт запущен
[67500] 2025-10-25 11:06:29 Скрипт запущен
[67500] 2025-10-25 11:06:59 Скрипт завершен, работал 0 минут
[67542] 2025-10-25 11:16:14 Скрипт запущен
[67542] 2025-10-25 11:17:18 Скрипт завершен, работал 1 минут
user1@eltex-practice2-pg1-v5: $

```

2

Создайте скрипт на языке bash с именем observer.sh, читающий файл конфигурации со списком скриптов observer.conf, проверяющим их наличие в списке работающих процессов поиском в /proc и запускающих их в отключенном от терминала режиме (nohup) в случае отсутствия в нем. Информация о перезапуске дописывайте в файл observer.log

```

user1@eltex-practice2-pg1-v5: $ nano observer.sh
user1@eltex-practice2-pg1-v5: $ user1@eltex-practice2-pg1-v5: $ chmod +x observer.sh
user1@eltex-practice2-pg1-v5: $ nano observer.conf
user1@eltex-practice2-pg1-v5: $ ./observer.sh
user1@eltex-practice2-pg1-v5: $ cat observer.log
[2025-10-25 11:20:31] Observer запущен
[2025-10-25 11:20:31] ЗАПУСК: worker1.sh не найден, запускаю...
[2025-10-25 11:20:31] УСПЕХ: worker1.sh запущен с PID 67691
[2025-10-25 11:20:31] ЗАПУСК: worker2.sh не найден, запускаю...
[2025-10-25 11:20:31] УСПЕХ: worker2.sh запущен с PID 67827
[2025-10-25 11:20:31] СКРИПТ worker1.sh уже запущен
[2025-10-25 11:20:32] СКРИПТ worker2.sh уже запущен
[2025-10-25 11:20:32] ЗАПУСК: processor.sh не найден, запускаю...
[2025-10-25 11:20:32] УСПЕХ: processor.sh запущен с PID 68212
[2025-10-25 11:20:32] Observer завершил работу
user1@eltex-practice2-pg1-v5: $
user1@eltex-practice2-pg1-v5: $ ps aux | grep -E "worker1|worker2|processor"
user1      67691  0.0  0.1  7340  3712 pts/1    S   11:20   0:00 bash /home/user1/worker1.sh
user1      67827  0.0  0.1  7340  3456 pts/1    S   11:20   0:00 bash /home/user1/worker2.sh
user1      68212  0.0  0.1  7340  3712 pts/1    S   11:20   0:00 bash /home/user1/processor.sh
user1      68220  0.0  0.0  6680  2304 pts/1    S+  11:20   0:00 grep --color=auto -E worker1|worker2|processor
user1@eltex-practice2-pg1-v5: $ tail -f observer.log
[2025-10-25 11:20:31] Observer запущен
[2025-10-25 11:20:31] ЗАПУСК: worker1.sh не найден, запускаю...
[2025-10-25 11:20:31] УСПЕХ: worker1.sh запущен с PID 67691
[2025-10-25 11:20:31] ЗАПУСК: worker2.sh не найден, запускаю...
[2025-10-25 11:20:31] УСПЕХ: worker2.sh запущен с PID 67827
[2025-10-25 11:20:31] СКРИПТ worker1.sh уже запущен
[2025-10-25 11:20:32] СКРИПТ worker2.sh уже запущен
[2025-10-25 11:20:32] ЗАПУСК: processor.sh не найден, запускаю...
[2025-10-25 11:20:32] УСПЕХ: processor.sh запущен с PID 68212
[2025-10-25 11:20:32] Observer завершил работу

```

3

Настройте запуск observer.sh посредством cron по расписанию – 1 раз в минуту

4

Создайте несколько символических ссылок на файл template_task.sh c

различными именами (рабочие задачи), добавьте в файл конфигурации

observer.conf соответствующие записи об этих задачах, включая исходный

файл template_task.sh

```
user1@eltex-practice2-pg1-v5:~$ ls -la worker*.sh processor.sh analyzer.sh
lrwxrwxrwx 1 user1 user1 16 Oct 25 11:34 analyzer.sh -> template_task.sh
lrwxrwxrwx 1 user1 user1 16 Oct 25 11:34 processor.sh -> template_task.sh
lrwxrwxrwx 1 user1 user1 16 Oct 25 11:34 worker1.sh -> template_task.sh
lrwxrwxrwx 1 user1 user1 16 Oct 25 11:34 worker2.sh -> template_task.sh
user1@eltex-practice2-pg1-v5:~$ cat > observer.conf << 'EOF'
> # Конфигурация observer - список скриптов для мониторинга
> /home/user1/worker1.sh
> /home/user1/worker2.sh
> /home/user1/processor.sh
> /home/user1/analyzer.sh
> /home/user1/template_task.sh
> EOF
user1@eltex-practice2-pg1-v5:~$ ls -la worker*.sh processor.sh analyzer.sh
.conf lrwxrwxrwx 1 user1 user1 16 Oct 25 11:34 analyzer.sh -> template_task.sh
lrwxrwxrwx 1 user1 user1 16 Oct 25 11:34 processor.sh -> template_task.sh
lrwxrwxrwx 1 user1 user1 16 Oct 25 11:34 worker1.sh -> template_task.sh
lrwxrwxrwx 1 user1 user1 16 Oct 25 11:34 worker2.sh -> template_task.sh
user1@eltex-practice2-pg1-v5:~$ cat observer.conf
# Конфигурация observer - список скриптов для мониторинга
/home/user1/worker1.sh
/home/user1/worker2.sh
/home/user1/processor.sh
/home/user1/analyzer.sh
/home/user1/template_task.sh
user1@eltex-practice2-pg1-v5:~$
```

```

user1@elitex-practice2-pg1-v5: $ crontab -r
user1@elitex-practice2-pg1-v5: $ echo "Cron остановлен"
Cron остановлен
user1@elitex-practice2-pg1-v5: $ crontab -l
no crontab for user1
user1@elitex-practice2-pg1-v5: $ █
1rwxrwxrwx 1 user1 user1 16 Oct 25 11:34 worker2.sh -> template_task.sh
user1@elitex-practice2-pg1-v5: $ cat observer.conf
# Конфигурация observer - список скриптов для мониторинга
/home/user1/worker1.sh
/home/user1/worker2.sh
/home/user1/processor.sh
/home/user1/analyzer.sh
/home/user1/template_task.sh
user1@elitex-practice2-pg1-v5: $ echo "Ждем для накопления логов"
Ждем для накопления логов
user1@elitex-practice2-pg1-v5: $ sleep 300
user1@elitex-practice2-pg1-v5: $ tar -czf practice_report.tar.gz \
> template_task.sh \
> observer.sh \
> observer.conf \
> report_*.log \
> observer.log \
> worker1.sh \
> worker2.sh \
> processor.sh \
> analyzer.sh
user1@elitex-practice2-pg1-v5: $ tar -tzf practice_report.tar.gz
template_task.sh
observer.sh
observer.conf
report_analyzer.log
report_.log
report_processor.log
report_worker1.log
report_worker2.log
observer.log
worker1.sh
worker2.sh
processor.sh
analyzer.sh
user1@elitex-practice2-pg1-v5: $ ls -la *.log
-rw-rw-r-- 1 user1 user1 11743 Oct 25 11:47 observer.log
-rw-rw-r-- 1 user1 user1 203 Oct 25 11:43 report_analyzer.log
-rw-rw-r-- 1 user1 user1 112 Oct 25 10:39 report_.log
-rw-rw-r-- 1 user1 user1 204 Oct 25 11:41 report_processor.log
-rw-rw-r-- 1 user1 user1 778 Oct 25 11:40 report_worker1.log
-rw-rw-r-- 1 user1 user1 351 Oct 25 11:44 report_worker2.log
user1@elitex-practice2-pg1-v5: $ tail -n 20 observer.log
[2025-10-25 11:45:01] СКРИПТ analyzer.sh уже запущен
[2025-10-25 11:45:01] ЗАПУСК: template_task.sh не найден, запускаю...
[2025-10-25 11:45:01] УСПЕХ: template_task.sh запущен с PID 80854
[2025-10-25 11:45:01] Observer завершил работу
[2025-10-25 11:46:01] Observer запущен
[2025-10-25 11:46:01] СКРИПТ worker1.sh уже запущен
[2025-10-25 11:46:01] СКРИПТ worker2.sh уже запущен
[2025-10-25 11:46:02] СКРИПТ processor.sh уже запущен
[2025-10-25 11:46:02] СКРИПТ analyzer.sh уже запущен
[2025-10-25 11:46:02] ЗАПУСК: template_task.sh не найден, запускаю...
[2025-10-25 11:46:02] УСПЕХ: template_task.sh запущен с PID 81527
[2025-10-25 11:46:02] Observer завершил работу
[2025-10-25 11:47:01] Observer запущен
[2025-10-25 11:47:01] СКРИПТ worker1.sh уже запущен
[2025-10-25 11:47:01] СКРИПТ worker2.sh уже запущен
[2025-10-25 11:47:01] СКРИПТ processor.sh уже запущен
[2025-10-25 11:47:01] СКРИПТ analyzer.sh уже запущен
[2025-10-25 11:47:01] ЗАПУСК: template_task.sh не найден, запускаю...
[2025-10-25 11:47:01] УСПЕХ: template_task.sh запущен с PID 82205
[2025-10-25 11:47:01] Observer завершил работу

```

5

Соберите статистику работы в виде набора файлов report_*.log, observer.log, приложите их вместе с исходными текстами скриптов в качестве отчета в виде сжатого архива tar. Не забудьте остановить процесс, удалив задачу в cron!


```
[2025-10-25 11:47:01] Observer завершил работу
user1@eltex-practice2-pg1-v5:~$ crontab -r
user1@eltex-practice2-pg1-v5:~$ echo "Cron остановлен"
Cron остановлен
user1@eltex-practice2-pg1-v5:~$ crontab -l
no crontab for user1
user1@eltex-practice2-pg1-v5:~$ ls -la practice_report.tar.gz
-rw-rw-r-- 1 user1 user1 2895 Oct 25 11:46 practice_report.tar.gz
user1@eltex-practice2-pg1-v5:~$
```

```
-rw-rw-r-- 1 user1 user1 2895 Oct 25 11:46 practice_report.tar.gz
user1@eltex-practice2-pg1-v5:~$ tar -tzf practice_report.tar.gz
template_task.sh
observer.sh
observer.conf
report_analyzer.log
report_.log
report_processor.log
report_worker1.log
report_worker2.log
observer.log
worker1.sh
worker2.sh
processor.sh
analyzer.sh
user1@eltex-practice2-pg1-v5:~$
```

```
user1@eltex-practice2-pg1-v5:~$ ps aux | grep -E "worker|processor|analyzer|template_task" | grep -v grep
root      4  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-rcu_g]
root      5  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-rcu_p]
root      6  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-slub_]
root      7  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-netns]
root     10  0.0  0.0      0  0 ?    I<    Oct15   0:49 [kworker/0:0H-kblockd]
root     12  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-mm_pe]
root     26  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/1:0H-events_highpri]
root     28  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-inet_]
root     34  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-write]
root     39  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-kinte]
root     40  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-kbloc]
root     41  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-blkcg]
root     43  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-tpm_d]
root     44  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-ata_s]
root     45  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-md]
root     46  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-md_bi]
root     47  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-edac-]
root     48  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-devfr]
root     50  0.0  0.0      0  0 ?    I<    Oct15   1:06 [kworker/1:1H-kblockd]
root     53  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-kthro]
root     54  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-acpi_]
root     56  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-scsi_]
root     58  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-scsi_]
root     60  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-scsi_]
root     64  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-mld]
root     66  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-ipv6_]
root     74  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-kstrp]
root     76  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/u5:0]
root     81  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-crypt]
root     93  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-charg]
root    149  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-ttm]
root    174  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-kdmf1]
root    200  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-raid5]
root    240  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-ext4-]
root    357  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-kmpat]
root    358  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-kmpat]
root    519  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-ext4-]
root    545  0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-cfg80]
root    1155 0.0  0.0      0  0 ?    I<    Oct15   0:00 [kworker/R-tls-s]
root   30701 0.0  0.0      0  0 ?    I<    Oct16   0:00 [kworker/0:1H-kblockd]
root   67077 0.0  0.0      0  0 ?    I    09:42   0:00 [kworker/1:2-events]
root   67367 0.0  0.0      0  0 ?    I    10:08   0:00 [kworker/u4:0-events_power_efficient]
root   68222 0.0  0.0      0  0 ?    I    11:23   0:00 [kworker/u4:2-events_power_efficient]
root   68259 0.0  0.0      0  0 ?    I    11:25   0:00 [kworker/0:1-events]
root   76136 0.0  0.0      0  0 ?    I    11:38   0:00 [kworker/0:2-cgroup_destroy]
root   76137 0.0  0.0      0  0 ?    I    11:38   0:00 [kworker/u4:1-events_unbound]
root   76794 0.0  0.0      0  0 ?    I    11:40   0:00 [kworker/1:1-cgroup_destroy]
root   82216 0.0  0.0      0  0 ?    I    11:50   0:00 [kworker/1:0-cgroup_destroy]
root   82228 0.0  0.0      0  0 ?    I    11:53   0:00 [kworker/1:3]
root   82229 0.0  0.0      0  0 ?    I    11:53   0:00 [kworker/0:0]
root   82230 0.0  0.0      0  0 ?    I    11:53   0:00 [kworker/0:3-events]
root   82251 0.0  0.0      0  0 ?    I    11:56   0:00 [kworker/u4:3-events_power_efficient]
user1@eltex-practice2-pg1-v5:~$
```