Липецкий государственный технический университет

Кафедра прикладной математики

Отчет по лабораторной работе №2 «Процессы в операционной системе Linux.»

Студент		Стукановский А.О	
	подпись, дата	фамилия, инициалы	
Группа		ПМ-18	
Руководитель			
доц., к.п.н. кафедры АСУ		Кургасов В. В.	
ученая степень, ученое звание	подпись, дата	фамилия, инициалы	

Содержание

Цель работы	3
Задание кафедры	3
Выполнение работы	5
Часть I:	5
Загрузка пользователем	5
Поиск файла с образом ядра. Определение по имени файла номера версии Linux	5
Просмотр процессов ps –f	5
Создание с помощью редактора vi двух сценариев loop и loop2	5
Запустить loop2 на переднем плане	7
Остановка процесса через сигнал STOP	7
Последовательный просмотр процессов через ps –f	8
Использование сигнала kill -9 PID, чтобы Убить процесс loop2	8
Запуск в фоне процесса loop: sh loop	8
Завершение процесса loop командой kill -15 PID	9
Третий запуск в фоне процесса loop2. Командой kill -9 PID без остановки убить данный	
процесс	9
Запуск экземпляра оболочки: bash	9
Запуск нескольких процессов в фоне.	10
Часть II:	11
Запуск в консоли на выполнение трех задач(две в интерактивном режиме, одна - в фо-	
новом)	11
Перевод одной из задач, выполняющихся в интерактивном режиме, в фоновый режим	11
Перевод задач из фонового режима в интерактивный и наоборот	11
Создание именованного канала для архивирования и осуществление передачи в канал	12
Часть III (вариант 8):	12
Сохранение в файл мгновенного состояния процессов системы указанного пользователя.	12
Послать сигнал SIGINT (по имени и по номеру сигнала) всем процессам, запущенным	
командой vi. Сообщить, успешно ли был послан сигнал	13
Измените на 3 единицы приоритеты процессов, владельцем которых является текущий	
пользователь.	13
Вывод	15
Список литературы	16

Цель работы

Ознакомиться на практике с понятием процесса в операционной системе. Приобрести опыт и навыки управления процессами в операционной системе Linux.

Задание кафедры

Часть I:

- 1) Загрузиться пользователем.
- 2) Найти файл с образом ядра. Выяснить по имени файла номер версии Linux.
- 3) Посмотреть процессы ря –f. Прокомментировать. Для этого почитать man ря.
- 4) Написать с помощью редактора vi два сценария loop и loop2. Текст сценариев: Loop: while true; do true; done Loop2: while true; do true; done
- 5) Запустить loop2 на переднем плане: sh loop2.
- 6) Остановить, послав сигнал STOP.
- 7) Посмотреть последовательно несколько раз ps -f. Записать сообщение, объяснить.
- 8) Убить процесс loop2, послав сигнал kill -9 PID. Записать сообщение. Прокомментировать.
- 9) Запустить в фоне процесс loop: sh loop. Не останавливая, посмотреть несколько раз: ps –f. Записать значение, объяснить.
- 10) Завершить процесс loop командой kill -15 PID. Записать сообщение, прокомментировать.
- 11) Третий раз запустить в фоне. Не останавливая убить командой kill -9 PID.
- 12) Запустить еще один экземпляр оболочки: bash.
- 13) Запустить несколько процессов в фоне. Останавливать их и снова запускать. Записать результаты просмотра командой ps-f.

Часть II:

- 1) Запустить в консоли на выполнение три задачи, две в интерактивном режиме, одну в фоновом.
- 2) Перевести одну из задач, выполняющихся в интерактивном режиме, в фоновый режим.
- 3) Провести эксперименты по переводу задач из фонового режима в интерактивный и наоборот.
- Создать именованный канал для архивирования и осуществить передачу в канал списка файлов домашнего каталога вместе с подкаталогами (ключ -R),
 одного каталога вместе с файлами и подкаталогами.
- 5) В отчете предоставьте все шаги ваших действий. То есть следует привести следующее: текст задания, а следом за ним снимок экрана консоли с результатами выполнения задания. Кроме того, перед скриншотом следует привести текстовую запись использованных команд.

- 1) Сохранить в файл мгновенное состояние процессов системы указанного пользователя.
- 2) Послать сигнал SIGINT (по имени и по номеру сигнала) всем процессам, запущенным командой vi. Сообщить, успешно ли был послан сигнал.
- 3) Измените на 3 единицы приоритеты процессов, владельцем которых является текущий пользователь.
- 4) В отчете предоставьте все шаги ваших действий. То есть следует привести следующее: текст задания, а следом за ним снимок экрана консоли с результатами выполнения задания. Кроме того, перед скриншотом следует привести текстовую запись использованных команд. Кратко поясните результаты выполнения всех команд.

Выполнение работы

Часть I:

Загрузка пользователем.

Для начала авторизуемся пользователем Artem (рисунок 1).

```
ubuntuserver login: artem
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0–51–generic x86_64)
```

Рисунок 1.

Поиск файла с образом ядра. Определение по имени файла номера версии Linux.

Файл ядра расположен в директории /boot и называется vmlinuz-<версия> (рисунок 2). Как видно, таких файлов несколько, поэтому узнаем точную версию ядра по средствам команды uname -r.

```
artem@ubuntuserver:~$ ls /boot
System.map-5.4.0-48-generic initrd.img vmlinuz
System.map-5.4.0-51-generic initrd.img-5.4.0-48-generic vmlinuz-5.4.0-48-generic
config-5.4.0-48-generic initrd.img-5.4.0-51-generic vmlinuz-5.4.0-51-generic
config-5.4.0-51-generic initrd.img.old vmlinuz.old
grub lost+found
artem@ubuntuserver:~$ uname -r
5.4.0-51-generic
```

Рисунок 2.

Просмотр процессов ps –f.

Для того, чтобы просмотреть удобный листинг процессов активного пользователя, воспользуемся командой ps -f, результат на рисуноке 3.

Утилита р
я выводит снимок процессов. Позволяет найти процессы по имени, пользователю или терминалу.

```
artem@ubuntuserver:~$ ps -f
UID PID PPID C STIME TTY TIME CMD
artem 992 834 0 15:30 tty1 00:00:00 -bash
artem 1124 992 0 15:44 tty1 00:00:00 ps -f
artem@ubuntuserver:~$ _
```

Рисунок 3.

Создание с помощью редактора vi двух сценариев loop и loop2.

С помощью редактора vi создадим два сценария loop и loop2, где сценарий loop содержит (рисунок 4), а loop2 - (рисунок 5). Для создания сценариев используется команда vi <название $>_{(67)}$.



Рисунок 4.

Рисунок 5.

artem@ubuntuserver:~\$ vi loop

Рисунок 6.

"loop" [New] 1L, 26C written artem@ubuntuserver:~\$ vi loop2_

Рисунок 7.

После создания, сценарии остаются в корневой папке пользователя, создавшего их.

Запустить loop2 на переднем плане.

Запустим сценарий loop2 на исполнение. Воспользуемся командой sh <имя_{>(8)}.

artem@ubuntuserver:~\$ sh loop2_

Рисунок 8.

Остановка процесса через сигнал STOP.

Сигнал STOP приостанавливает процесс: в таком состоянии процесс не удаляется из таблицы процессов, но и не выполняется. Приостановить процесс можно комбинацией клавиш "Ctrl + Z"(рисунок 9).

```
Hello
Hello
^Z
[1]+ Stopped sh loop2
artem@ubuntuserver:~$
```

Рисунок 9.

Последовательный просмотр процессов через ps -f.

Выполнив команду рs -f несколько раз, можем обнаружить(рисунок 10), что процесс до сих пор отображается в списке процесов, однако время выполнения программы остаётся неизменным.

```
artem@ubuntuserver:'
UID PID
artem 992
                        PPID
834
                                 STIME
15:30
15:51
                                                         TIME CMD
                                                    00:00:00
                                                              -bash
                                                    00:00:01 sh loop2
artem
                         992
                                  15:51 ttq1
                                                    00:00:00 ps -f
artem
artem@ubuntuserver:
                        $ ps
               PID
992
                                 STIME
15:30
UID
                        PPID
                                                         TIME CMD
artem
artem
              1130
                                                    00:00:01 sh loop2
                                                    00:00:00 ps -f
                         992
                                  15:52 tty1
artem
artem@ubuntuserver:
                        $ ps
               PID
992
UID
                        PPID
                                                         TIME CMD
                         834
artem
                                  15:30
                                                    00:00:00 -bash
artem
              1130
                         992
                                  15:51
                                                    00:00:01 sh loop2
                                                    00:00:00 ps -f
artem
              1133
                         992
                                 15:52 tty1
 artem@ubuntuserver:
```

Рисунок 10.

Использование сигнала kill -9 PID, чтобы Убить процесс loop2.

На рисунке 10 можно заметить, что в таблице процессов содержится процесс loop2 с PID 1130. Воспользовавшись командой kill -9 1130 мы сможем завершить процесс.

```
TIME CMD
00:00:00 -bash
UID
              ΡID
                       PPID
                               STIME TTY
                               15:30 tty1
artem
                       834
             1130
                       992
                               15:51 tty1
artem
                                                 00:00:01 sh loop2
                               15:53 tty1
artem
             1134
                                                 00:00:00 ps -f
artem@ubuntuserver:
                                  1130
                                  sh loop2
     Killed
```

Рисунок 11.

Так же, из рисунка видно, что консоль сообщает нам о том, что процесс был завершён.

Запуск в фоне процесса loop: sh loop.

Для того, чтобы запустить процесс сценария параллельно, достаточно добавить в конец командной строки символ "": sh $uma_{(12)}$.

```
em@ubuntuserver:~$ sh loop&
[1] 1135
artem@ubuntuserver:
                       ′$ ps
              PID
992
1135
                       PPID
834
HTD
                              C STIME TTY
                                                       TIME CMD
                             0 15:30 tty1
92 15:54 tty1
artem
                                                  00:00:00 -bash
artem
                        992
                            92
                                                  00:00:03 sh loop
                                15:54
                                                  00:00:00 ps -f
artem
              1136
                        992
                                       tty1
artem@ubuntuserver:
                       $ ps
                       PPID
834
UID
               ΡID
                              C STIME
                                                       TIME CMD
                             0 15:30
99 15:54
artem
               992
                                                  00:00:00 -bash
              1135
                             99
                                       tty1
                                                  00:00:05 sh loop
artem
                        992
                        992
                                15:54
                                                  00:00:00 ps -f
artem
              1137
artem@ubuntuserver:
                       $ ps
UID
               ΡID
                       PPID
                              C STIME
                                                       TIME CMD
               992
                        834
                             0 15:30
95 15:54
artem
                                       tty1
                                                  00:00:00 -bash
              1135
                                                  00:00:07 sh loop
artem
                        992
                                       tty1
                                                  00:00:00 ps -f
                        992
                                15:54
              1138
artem
artem@ubuntuserver:
```

Рисунок 12.

Завершение процесса loop командой kill -15 PID.

На рисунке 13 приведена таблица процессов, где присутствует исполняемый в фоновом режиме процесс loop.

```
tem@ubuntuserver:
artem@ubuntuserver:
JID
              ΡID
                                                   TIME CMD
              992
                      834
                              15:30
                                               00:00:00 -bash
artem
                                    tty1
                                              00:00:00 ps -f
                              15:56 tty1
            1139
                       992
artem
      Terminated
                                sh loop
 rtem@ubuntuserver:~$
```

Рисунок 13.

Сигнал номер 15 (TERM) служит для прерывания работы задания.

Третий запуск в фоне процесса loop2. Командой kill -9 PID без остановки убить данный процесс.

Через команду sh loop вновь запускается в фоновом режиме процесс loop. Для удаления процесса в этот раз используем команду kill -9 <PID> (рисунок 14).

```
tem@ubuntuserver:~$ sh loop&
[1] 1140
artem@ubuntuserver:
                     '$ ps
                     PPID
HTD
             ΡID
                              STIME TTY
                                                   TIME CMD
                                              00:00:00 -bash
                      834
artem
                                    tty1
                          96 15:58 tty1
artem
             1140
                      992
                                              00:00:04 sh loop
artem
            1141
                            0 15:58 tty1
                                              00:00:00 ps -f
                     ′$ kill -9 1140
artem@ubuntuserver:
artem@ubuntuserver:
                     '$ ps
             PID
992
                                                   TIME CMD
                           C STIME TTY
UID
                     PPID
                      834
artem
                              15:30 tty1
                                              00:00:00 -bash
  tem
             1142
                      992
                            0 15:58 tty1
                                              00:00:00 ps -f
```

Рисунок 14.

Запуск экземпляра оболочки: bash.

Чтобы запустить новый экземпляр оболочки bash, используем команду bash (рисунок 15).

```
artem@ubuntuserver:
                        ′$ bash
artem@ubuntuserver:
                        $ ps
                              C STIME TTY
0 15:30 tty1
0 16:00 tty1
               PID
                        PPID
HTD
                                                         TIME CMD
artem
                         834
                                                    00:00:00 -bash
              1148
                                                    00:00:00 bash
artem
                         992
artem
              1154
                        1148
                               0 16:00 tty1
                                                    00:00:00 ps -f
```

Рисунок 15.

Запуск нескольких процессов в фоне.

На рисунке 16 осуществляется запуск в фоновом режиме трёх процессов loop в фоновом режиме. Через команду kill -19 <PID> процесс приостанавливается, что видно в полноформатном листинге ps -f (рисунок 16).

```
rtem@ubuntuserver:~$ sh loop&
[1] 1155
artem@ubuntuserver:~$ sh loop&
[2] 1156
artem@ubuntuserver:~$ sh loop&
[3] 1157
artem@ubuntuserver:
UID PID
artem 992
                          ~$ ps −f
PPID C
834 O
                                     STIME TTY
                                                               TIME CMD
                          16:00 tty1
16:00 tty1
1148 47 16:01 tty1
1148 38 16:01 tty1
1148 32 16:01 tty1
1148 0 16:01
                                     15:30
                                             tty1
                                                          00:00:00 -bash
                1148
                                                          00:00:00 bash
artem
                                                          00:00:03 sh loop
artem
                1156
1157
                                                         00:00:02 sh loop
00:00:01 sh loop
artem
artem
artem
                1158
                                                          00:00:00 ps -f
artem@ubuntuserver:
artem@ubuntuserver:
                          '$ ps -f
                 PID
992
                                     STIME TTY
15:30 tty
UID
                          PPID
                                                               TIME CMD
                            834
                                             tty1
                                                          00:00:00 -bash
artem
                1148
artem
                                     16:00
                                                          00:00:00 bash
                          1148 31 16:01 tty1
1148 36 16:01 tty1
1148 35 16:01 tty1
                1155
                                                          00:00:09 sh loop
artem
                1156
                                                          00:00:10 sh loop
artem
                                                          00:00:09 sh loop
artem
                1159
artem
                           1148
                                                          00:00:00 ps -f
[1]+ Stopped
artem@ubuntuserver:^
                                        sh loop
                          ′$ ps −f
UID
                 PID
992
                          PPID
834
                                     STIME TTY
15:30 tty
                                                               TIME CMD
artem
                                                                     -bash
                                             tty1
artem
                1148
                            992
                                     16:00
                                                          00:00:00 bash
                                     16:01 tty1
16:01 tty1
16:01 tty1
                1155
                                                          00:00:09 sh loop
artem
                          1148
                1156
                          1148 38
                                                          00:00:12 sh
artem
                                                                          loop
                1157
                                                          00:00:12 sh loop
                           1148 37
artem
artem
                           1148
                                     16:01 tty1
                                                          00:00:00 ps -f
 rtem@ubuntuserver:'
```

Рисунок 16.

Далее для возобнавления работы используем kill -18 <PID>. Листинг, подтверждающий это, приведён на рисунке 17.

```
artem@ubuntuserver:~$ kill –18 1155
                      '$ ps –
PPID
artem@ubuntuserver:
              PID
992
                               STIME
ШID
                                                     TIME CMD
artem
                       834
                               15:30
                                                00:00:00
artem
             1148
                               16:00
                                                00:00:00 bash
                                      tty1
                      1148
             1155
                            16
                               16:01
                                      tty1
                                                00:00:10 sh loop
artem
                               16:01 tty1
                                                00:00:28 sh loop
             1156
                      1148 42
artem
                      1148 42
                                                00:00:27 sh loop
artem
             1157
                               16:01 tty1
             1161
                      1148
                               16:02
                                                00:00:00 ps
artem
artem@ubuntuserver
                      $ ps
                      PPID
834
                               STIME
                                                     TIME CMD
UID
              PID
                                                00:00:00
                               15:30
                                                          -bash
artem
             1148
                               16:00
                                                00:00:00 bash
artem
                       992
              1155
                      1148
                                                00:00:12
                                                          sh loop
artem
                      1148 42
                               16:01
                                                00:00:29 sh loop
artem
                                                00:00:29 sh loop
                      1148 41
                               16:01
artem
             1157
                                                00:00:00 ps -f
             1162
                      1148
                               16:02
artem
rtem@ubuntuserver:
```

Рисунок 17.

Часть II:

Запуск в консоли на выполнение трех задач(две в интерактивном режиме, одна - в фоновом).

Для начала создадим 2 интерактивных процесса. Используем sh loop и будем приостанавливать каждый из них с помошью "Ctrl+Z в резуль. Так же запустим один процесс в фоне sh loop. Результат приведён на рисунке 18.

```
tem@ubuntuserver:~$ sh loop
     Stopped
                                 sh loop
artem@ubuntuserver:~$ sh loop
                                 sh loop
[2]+ Stopped
artem@ubuntuserver:~$ sh loop&
[3] 933
artem@ubuntuserver:
JID
             PID
                     PPID
                              STIME
                                                    TIME CMD
              922
                                               00:00:00 -bash
artem
                                     ttu1
                              19:04
19:04
                      922
                           13
                                               00:00:03 sh loop
artem
                                     tty1
              932
artem
                                     tty1
                                               00:00:00 sh
                                                            loop
                           93
                                               00:00:02 sh loop
artem
                               19:04
artem
```

Рисунок 18.

Перевод одной из задач, выполняющихся в интерактивном режиме, в фоновый режим.

Используя команду bg переведём процессы 1 и 2 (которые в данный момент стоят на паузе) в фоновый режим (рисунок 19).

```
artem@ubuntuserver:~$ bg 1 2
[1]– sh loop &
[2]+ sh loop &
```

Рисунок 19.

Перевод задач из фонового режима в интерактивный и наоборот.

В случае перевода процесса из фонового в интерактивный режим, программа сразу начинает взаимодействие с пользователем, в связи с чем её приходится ставит на паузу (рисунок 20).

```
tem@ubuntuserver:~$ fg 3
  loop
     Stopped
                               sh loop
artem@ubuntuserver:~$ bg
[3] + sh loop &
artem@ubuntuserver:
             PID
                             STIME
                                                  TIME CMD
                                             00:00:00
artem
             922
                                                      -bash
artem
                                             00:00:17 sh loop
                                             00:00:15 sh loop
artem
                             19:04
                                             00:00:29 sh loop
artem
                             19:04
                                             00:00:00 ps -f
artem
             936
 rtem@ubuntuserver:
```

Рисунок 20.

Создание именованного канала для архивирования и осуществление передачи в канал.

Для начала создадим поток pipe используя команду mkfifo. После передадив в неё список файлов домашнего каталога с подкаталогами, используя ключ -R (рисунок 21)

```
artem@ubuntuserver:~$ ls
1000 loop loop2 result.txt snap test.sh text.txt
artem@ubuntuserver:~$ mkfifo pipe
artem@ubuntuserver:~$ ls
1000 loop loop2 pipe result.txt snap test.sh text.txt
artem@ubuntuserver:~$ ls -R > pipe
```

Рисунок 21.

А теперь передадим туда лишь файлы того каталога, в котором находимся, используем ls - l > pipe (рисунок 22).

```
artem@ubuntuserver:~$ ls –l > pipe
```

Рисунок 22.

Часть III (вариант 8):

Сохранение в файл мгновенного состояния процессов системы указанного пользователя.

Для умпрощения сохранения в файл состояния процессов системы указанного пользователя был написан следующий сценарий (рисунок 23).

```
read n_

(rm text.txt)

(ps -f -u $n) >> text.txt
```

Рисунок 23.

При его запуске интерактивная программа требует от вас ввода имени пользователя, информацию о процессах которого вы хотите сохранить. После ввода имени пользователя, программа создаёт txt документ с именем text (рисунок 24).

```
artem@ubuntuserver:~$ ls
loop loop2 result.txt snap test.sh testresult.txt
artem@ubuntuserver:~$ ./test.sh
artem
rm: cannot remove 'text.txt': No such file or directory
artem@ubuntuserver:~$ ls
loop loop2 result.txt snap test.sh testresult.txt text.txt
artem@ubuntuserver:~$ vi text.txt_
```

Рисунок 24.

Открыв текстовый файл мы увидим довольно подробный листинг (ps -f) для заданного пользователя, это видно на рисунке 25.

UID	PID	PPID	C STIME TTY	TIME CMD
artem	903	1	0 18:15 ?	00:00:00 /lib/systemd/systemd ––user
artem	909	903	0 18:15 ?	00:00:00 (sd-pam)
artem	914	639	0 18:15 tty1	00:00:00 -bash
artem	1007	914	0 18:38 tty1	00:00:00 -bash
artem	1009	1007	0 18:38 tty1	00:00:00 ps -f -u artem

Рисунок 25.

Послать сигнал SIGINT (по имени и по номеру сигнала) всем процессам, запущенным командой vi. Сообщить, успешно ли был послан сигнал.

У нас имеются два активных процесса vi (рисунок 26).

artem@ubu	ntuserver	:~\$ ps	-f			
UID	PID	PPID	C STIME	TTY	TIME	CMD
artem	1383	1249	0 19:58	tty1	00:00:00	-bash
artem	1402	1383	0 20:00	tty1	00:00:00	vi text.txt
artem	1404	1383	0 20:01	tty1	00:00:00	vi loop2
artem	1407	1383	0 20:01	tty1	00:00:00	grepcolor
artem	1420	1383	0 20:11	ttu1	00:00:00	ps -f

Рисунок 26.

Попробуем послать сигнал SIGINT двумя способами(по номеру и имени сигнала) на два этих процесса разом. Из рисунка 27 видно, что данный сигнал не оказывает никакого влияния на процессы vi text.txt и vi loop2.

```
tem@ubuntuserver:
                               -2 1402 1404
artem@ubuntuserver:
                       $ ps
              PID
UID
                      PPID
                               STIME
                                                      TIME CMD
artem
                                19:58
                                      tty1
                                                 00:00:00 -bash
                               20:00
artem
             1402
                                      tty1
                               20:01 tty1
                                                 00:00:00 vi loop2
artem
             1404
                                                 00:00:00 grep --color=aut
00:00:00 ps -f
                               20:01 tty1
              1407
artem
                       1383
                               20:08 tty1
              1415
artem
                       1383
artem@ubuntuserver:
                       $ kill
artem@ubuntuserver:
                       $ ps
                      PPID
                                                 TIME CMD
00:00:00 -bash
UID
              PID
                               STIME
             1383
artem
                       1249
artem
              1402
                       1383
                                20:00
                                      tty1
                                                 00:00:00 vi text.txt
artem
              1404
                       1383
                               20:01
                                                 00:00:00 vi loop2
                               20:01
                                                 00:00:00 grep --color=aut
              1407
artem
                       1383
                                      ttu1
 rtem
                       1383
                                20:09
                                                 00:00:00 ns
```

Рисунок 27.

Измените на 3 единицы приоритеты процессов, владельцем которых является текущий пользователь.

На рисунке 28 представлен листинг процессов пользователя artem с указанием их приоритетов.

```
NI ADDR SZ WCHAN TTY
0 – 1833 do_wai tty1
0 – 1833 do_sig tty1
1833 do_sig tty1
tem@ubuntuserver
                                                                                              TIME CMD
00:00:00 bash
00:00:00 bash
     UID
                   PID
                              PPID
                                           PRI
                                            80
                   922
                               641
S
T
T
    1000
                                            80
                   944
                   945
                                                                                               00:00:00 bash
    1000
                                            80
                                                                                              00:03:29 sh
00:03:28 sh
                                                                                tty1
                   975
                                            80
    1000
                   976
                                922
                                            80
                                                              652
                                                                                tty1
                                                            652 -
5079 do_sig
1432 do_sig
R
    1000
                   977
978
                               922
922
922
                                                                                               00:03:28 sh
                                            80
80
                                                     0
    1000
                                                                                tty1
                   979
                                                                                ttý1
                                                                                               00:00:00 bash
    1000
                                            80
     1000
                                                                                               00:00:00 ps
```

Рисунок 28.

Используя команду renice увеличим значение приоритета (NI) на 3 для всех процессов пользователя artem. Результат представлен на рисунке 29.

```
artem@ubuntuserver:~$ renice 3 –u artem
1000 (user ID) old priority 0, new priority 3
artem@ubuntuserver:~$ ps –l
                                               PPID
641
922
922
922
                               PID
922
                                                                 PRI
83
            UID
                                                                             NI ADDR SZ WCHAN
3 – 1833 do_wai
                                                                                                                                          TIME CMD
00:00:00 bash
                                                                                                                                          00:00:00 bash
00:00:00 bash
00:00:00 bash
00:03:40 sh
00:03:39 sh
00:03:39 sh
                               944
945
975
                                                                   83
83
           1000
1000
                                                                                           1833 do_sig
                                                                                            1833
                                                                                                     do_sig
                                                                                                                      tty1
           1000
                                                                    83
                                                                                          652 - tty1
652 - tty1
5079 do_sig tty1
1432 do_sig tty1
                                                 922
922
922
922
922
                               976
977
                                                          33
33
                                                                   83
83
           1000
    R
R
T
                             978
979
1007
                                                                                                                                          00:00:00 vi
00:00:00 bash
00:00:00 ps
           1000
1000
                                                                   83
83
           1000
                                                                                           1888
                                                                                                                       ttÿ1
 artem@ubuntuserver:
```

Рисунок 29.

Вывод

B ходе лабораторной работы были усвоены понятия процесса в операционной системе и приобретены навыки управления процессами в операционной системе Linux.

Список литературы

- [1] Львовский, С.М. Набор и верстка в системе І
*ТеХ [Текст] / С.М. Львовский. М.: МЦНМО, 2006. 448 с.