

Липецкий государственный технический университет

Факультет автоматизации и информатики

Кафедра Автоматизированных систем управления

ЛАБОРАТОРНАЯ РАБОТА №3

по дисциплине «Операционная система Linux»

Процессы в операционной системе Linux

Вариант 5

Студент

Титов В.А.

Группа АС-20

Руководитель

Кургасов В.В.

к.п.н.

Липецк 2022

Цель работы

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

1 этап

```
root@10:/home/vlad# su - vlad
vlad@10:~$
```

Рис. 1 – Загружаемся пользователем

```
vlad@10:~$ cat /etc/debian_version
11.5
vlad@10:~$ _
```

Рис. 2 – Выясняем версию ОС Debian

Для отображения процессов мы используем команду `ps -f`.

Параметр `-f` используется для выбора информации, отображаемой `ps`.

Так же, с помощью параметра `-f` отображается формат контекста безопасности. Делаем полноформатный листинг.

```
vlad@10:~$ ps -f
UID          PID    PPID  C  STIME TTY          TIME CMD
vlad         518      412  0  10:32 tty1        00:00:00 -bash
vlad         649      648  0  11:50 tty1        00:00:00 -bash
vlad         658      649  0  11:55 tty1        00:00:00 vi 1
vlad         660      649  0  11:59 tty1        00:00:00 vi 1
vlad         666      649  0  12:06 tty1        00:00:00 vi 1
vlad         671      649  0  12:08 tty1        00:00:00 vi 1.txt
vlad         730      729  0  12:40 tty1        00:00:00 -bash
vlad         766      765  0  13:01 tty1        00:00:00 -bash
vlad        1062     1061  0  15:00 tty1        00:00:00 -bash
vlad        1229     1228  0  17:23 tty1        00:00:00 -bash
vlad        1234     1229  0  17:26 tty1        00:00:00 ps -f
vlad@10:~$
```

Рис. 3 – Просмотр процессов

Вызвав несколько раз команду `ps -f`, можем заметить, что ключевое слово `C` изменяется.

Ключевое `C` обозначает загрузку процессора. В настоящее время это целочисленное значение процента использования за время существования процесса.

```
vlad 666 649 0 06:36 tty1 00:00:00 vi 1
vlad 671 649 0 06:38 tty1 00:00:00 vi 1.txt
vlad 730 729 0 07:09 tty1 00:00:00 -bash
vlad 766 765 0 07:31 tty1 00:00:00 -bash
vlad 1062 1061 0 09:30 tty1 00:00:00 -bash
vlad 1229 1228 0 11:52 tty1 00:00:00 -bash
vlad 1306 1229 25 12:15 tty1 00:00:55 sh loop2
vlad 1312 1229 0 12:18 tty1 00:00:00 ps -f
vlad@10:~$ ps -f
UID      PID  PPID  C  STIME TTY      TIME CMD
vlad 518 412 0 05:02 tty1 00:00:00 -bash
vlad 649 648 0 06:20 tty1 00:00:00 -bash
vlad 658 649 0 06:24 tty1 00:00:00 vi 1
vlad 660 649 0 06:29 tty1 00:00:00 vi 1
vlad 666 649 0 06:36 tty1 00:00:00 vi 1
vlad 671 649 0 06:38 tty1 00:00:00 vi 1.txt
vlad 730 729 0 07:09 tty1 00:00:00 -bash
vlad 766 765 0 07:31 tty1 00:00:00 -bash
vlad 1062 1061 0 09:30 tty1 00:00:00 -bash
vlad 1229 1228 0 11:52 tty1 00:00:00 -bash
vlad 1306 1229 24 12:15 tty1 00:00:55 sh loop2
vlad 1313 1229 0 12:19 tty1 00:00:00 ps -f
vlad@10:~$ ps -f
UID      PID  PPID  C  STIME TTY      TIME CMD
vlad 518 412 0 05:02 tty1 00:00:00 -bash
vlad 649 648 0 06:20 tty1 00:00:00 -bash
vlad 658 649 0 06:24 tty1 00:00:00 vi 1
vlad 660 649 0 06:29 tty1 00:00:00 vi 1
vlad 666 649 0 06:36 tty1 00:00:00 vi 1
vlad 671 649 0 06:38 tty1 00:00:00 vi 1.txt
vlad 730 729 0 07:09 tty1 00:00:00 -bash
vlad 766 765 0 07:31 tty1 00:00:00 -bash
vlad 1062 1061 0 09:30 tty1 00:00:00 -bash
vlad 1229 1228 0 11:52 tty1 00:00:00 -bash
vlad 1306 1229 21 12:15 tty1 00:00:55 sh loop2
vlad 1314 1229 0 12:19 tty1 00:00:00 ps -f
vlad@10:~$
```

Рис. 8 – Просмотр последовательности командой `ps -f`

```
vlad@10:~$ ps -f
UID      PID  PPID  C  STIME TTY      TIME CMD
vlad 518 412 0 05:02 tty1 00:00:00 -bash
vlad 649 648 0 06:20 tty1 00:00:00 -bash
vlad 658 649 0 06:24 tty1 00:00:00 vi 1
vlad 660 649 0 06:29 tty1 00:00:00 vi 1
vlad 666 649 0 06:36 tty1 00:00:00 vi 1
vlad 671 649 0 06:38 tty1 00:00:00 vi 1.txt
vlad 730 729 0 07:09 tty1 00:00:00 -bash
vlad 766 765 0 07:31 tty1 00:00:00 -bash
vlad 1062 1061 0 09:30 tty1 00:00:00 -bash
vlad 1229 1228 0 11:52 tty1 00:00:00 -bash
vlad 1306 1229 5 12:15 tty1 00:00:55 sh loop2
vlad 1329 1328 0 12:33 tty1 00:00:00 -bash
vlad 1333 1329 0 12:33 tty1 00:00:00 ps -f
vlad@10:~$ kill -9 1306
vlad@10:~$ ps -f
UID      PID  PPID  C  STIME TTY      TIME CMD
vlad 518 412 0 05:02 tty1 00:00:00 -bash
vlad 649 648 0 06:20 tty1 00:00:00 -bash
vlad 658 649 0 06:24 tty1 00:00:00 vi 1
vlad 660 649 0 06:29 tty1 00:00:00 vi 1
vlad 666 649 0 06:36 tty1 00:00:00 vi 1
vlad 671 649 0 06:38 tty1 00:00:00 vi 1.txt
vlad 730 729 0 07:09 tty1 00:00:00 -bash
vlad 766 765 0 07:31 tty1 00:00:00 -bash
vlad 1062 1061 0 09:30 tty1 00:00:00 -bash
vlad 1229 1228 0 11:52 tty1 00:00:00 -bash
vlad 1329 1328 0 12:33 tty1 00:00:00 -bash
vlad 1334 1329 0 12:33 tty1 00:00:00 ps -f
vlad@10:~$
```

Рис. 9 – Убиваем процесс `loop2`

Ключ C равен 99, что означает загрузку процессора на 99%.

```
vlad@10:~$ sh loop&
[1] 1338
vlad@10:~$ ps -f
UID          PID    PPID  C  STIME TTY          TIME CMD
vlad          518      412  0  05:02 tty1        00:00:00 -bash
vlad          649      648  0  06:20 tty1        00:00:00 -bash
vlad          658      649  0  06:24 tty1        00:00:00 vi 1
vlad          660      649  0  06:29 tty1        00:00:00 vi 1
vlad          666      649  0  06:36 tty1        00:00:00 vi 1
vlad          671      649  0  06:38 tty1        00:00:00 vi 1.txt
vlad          730      729  0  07:09 tty1        00:00:00 -bash
vlad          766      765  0  07:31 tty1        00:00:00 -bash
vlad         1062     1061  0  09:30 tty1        00:00:00 -bash
vlad         1229     1228  0  11:52 tty1        00:00:00 -bash
vlad         1329     1328  0  12:33 tty1        00:00:00 -bash
vlad         1338     1329  99 12:35 tty1        00:00:04 sh loop
vlad         1339     1329  0  12:35 tty1        00:00:00 ps -f
```

Рис. 10 – В фоне запускаем процесс loop

Можно заметить, что загрузка процессора не меняется и равна 99%.

```
vlad          766      765  0  07:31 tty1        00:00:00 -bash
vlad         1062     1061  0  09:30 tty1        00:00:00 -bash
vlad         1229     1228  0  11:52 tty1        00:00:00 -bash
vlad         1329     1328  0  12:33 tty1        00:00:00 -bash
vlad         1338     1329  99 12:35 tty1        00:00:04 sh loop
vlad         1339     1329  0  12:35 tty1        00:00:00 ps -f
vlad@10:~$ ps -f
UID          PID    PPID  C  STIME TTY          TIME CMD
vlad          518      412  0  05:02 tty1        00:00:00 -bash
vlad          649      648  0  06:20 tty1        00:00:00 -bash
vlad          658      649  0  06:24 tty1        00:00:00 vi 1
vlad          660      649  0  06:29 tty1        00:00:00 vi 1
vlad          666      649  0  06:36 tty1        00:00:00 vi 1
vlad          671      649  0  06:38 tty1        00:00:00 vi 1.txt
vlad          730      729  0  07:09 tty1        00:00:00 -bash
vlad          766      765  0  07:31 tty1        00:00:00 -bash
vlad         1062     1061  0  09:30 tty1        00:00:00 -bash
vlad         1229     1228  0  11:52 tty1        00:00:00 -bash
vlad         1329     1328  0  12:33 tty1        00:00:00 -bash
vlad         1338     1329  99 12:35 tty1        00:00:26 sh loop
vlad         1340     1329  0  12:35 tty1        00:00:00 ps -f
vlad@10:~$ ps -f
UID          PID    PPID  C  STIME TTY          TIME CMD
vlad          518      412  0  05:02 tty1        00:00:00 -bash
vlad          649      648  0  06:20 tty1        00:00:00 -bash
vlad          658      649  0  06:24 tty1        00:00:00 vi 1
vlad          660      649  0  06:29 tty1        00:00:00 vi 1
vlad          666      649  0  06:36 tty1        00:00:00 vi 1
vlad          671      649  0  06:38 tty1        00:00:00 vi 1.txt
vlad          730      729  0  07:09 tty1        00:00:00 -bash
vlad          766      765  0  07:31 tty1        00:00:00 -bash
vlad         1062     1061  0  09:30 tty1        00:00:00 -bash
vlad         1229     1228  0  11:52 tty1        00:00:00 -bash
vlad         1329     1328  0  12:33 tty1        00:00:00 -bash
vlad         1338     1329  99 12:35 tty1        00:01:53 sh loop
vlad         1341     1329  0  12:37 tty1        00:00:00 ps -f
vlad@10:~$ _
```

Рис. 11 – Просмотр последовательности командой ps -f, не останавливая процесс

Можно заметить, вызвав атрибут -15 мы завершаем процесс.

```
vlad@10:~$ kill -15 1338
vlad@10:~$ ps -f
UID          PID    PPID  C  STIME TTY          TIME CMD
vlad         518      412  0  05:02 tty1        00:00:00 -bash
vlad         649      648  0  06:20 tty1        00:00:00 -bash
vlad         658      649  0  06:24 tty1        00:00:00 vi 1
vlad         660      649  0  06:29 tty1        00:00:00 vi 1
vlad         666      649  0  06:36 tty1        00:00:00 vi 1
vlad         671      649  0  06:38 tty1        00:00:00 vi 1.txt
vlad         730      729  0  07:09 tty1        00:00:00 -bash
vlad         766      765  0  07:31 tty1        00:00:00 -bash
vlad        1062     1061  0  09:30 tty1        00:00:00 -bash
vlad        1229     1228  0  11:52 tty1        00:00:00 -bash
vlad        1329     1328  0  12:33 tty1        00:00:00 -bash
vlad        1344     1329  0  12:44 tty1        00:00:00 ps -f
[1]+  Завершено      sh loop
vlad@10:~$ _
```

Рис. 12 – Убиваем процесс loop

Вызвав атрибут -9 мы убиваем процесс loop.

```
vlad@10:~$ sh loop&
[1] 1346
vlad@10:~$ ps -f
UID          PID    PPID  C  STIME TTY          TIME CMD
vlad         518      412  0  05:02 tty1        00:00:00 -bash
vlad         649      648  0  06:20 tty1        00:00:00 -bash
vlad         658      649  0  06:24 tty1        00:00:00 vi 1
vlad         660      649  0  06:29 tty1        00:00:00 vi 1
vlad         666      649  0  06:36 tty1        00:00:00 vi 1
vlad         671      649  0  06:38 tty1        00:00:00 vi 1.txt
vlad         730      729  0  07:09 tty1        00:00:00 -bash
vlad         766      765  0  07:31 tty1        00:00:00 -bash
vlad        1062     1061  0  09:30 tty1        00:00:00 -bash
vlad        1229     1228  0  11:52 tty1        00:00:00 -bash
vlad        1329     1328  0  12:33 tty1        00:00:00 -bash
vlad        1346     1329  99 12:48 tty1        00:00:04 sh loop
vlad        1347     1329  0  12:48 tty1        00:00:00 ps -f
vlad@10:~$ kill -9 1346
vlad@10:~$ ps -f
UID          PID    PPID  C  STIME TTY          TIME CMD
vlad         518      412  0  05:02 tty1        00:00:00 -bash
vlad         649      648  0  06:20 tty1        00:00:00 -bash
vlad         658      649  0  06:24 tty1        00:00:00 vi 1
vlad         660      649  0  06:29 tty1        00:00:00 vi 1
vlad         666      649  0  06:36 tty1        00:00:00 vi 1
vlad         671      649  0  06:38 tty1        00:00:00 vi 1.txt
vlad         730      729  0  07:09 tty1        00:00:00 -bash
vlad         766      765  0  07:31 tty1        00:00:00 -bash
vlad        1062     1061  0  09:30 tty1        00:00:00 -bash
vlad        1229     1228  0  11:52 tty1        00:00:00 -bash
vlad        1329     1328  0  12:33 tty1        00:00:00 -bash
vlad        1348     1329  0  12:48 tty1        00:00:00 ps -f
[1]+  Убито          sh loop
vlad@10:~$
```

Рис. 13 – Вызовем в фоне процесс loop, не останавливая убьем его командой kill -9 PID


```

vlad@10:~$ ps -f
UID          PID     PPID  C  STIME TTY          TIME CMD
vlad         518       412  0  05:02 tty1        00:00:00 -bash
vlad         649       648  0  06:20 tty1        00:00:00 -bash
vlad         658       649  0  06:24 tty1        00:00:00 vi 1
vlad         660       649  0  06:29 tty1        00:00:00 vi 1
vlad         666       649  0  06:36 tty1        00:00:00 vi 1
vlad         671       649  0  06:38 tty1        00:00:00 vi 1.txt
vlad         730       729  0  07:09 tty1        00:00:00 -bash
vlad         766       765  0  07:31 tty1        00:00:00 -bash
vlad        1062      1061  0  09:30 tty1        00:00:00 -bash
vlad        1229      1228  0  11:52 tty1        00:00:00 -bash
vlad        1329      1328  0  12:33 tty1        00:00:00 -bash
vlad        1350      1329  0  12:51 tty1        00:00:00 bash
vlad        1355      1350 29  12:53 tty1        00:00:21 bash
vlad        1356      1350  5  12:54 tty1        00:00:02 bash loop
vlad        1358      1350  7  12:54 tty1        00:00:01 bash loop2
vlad        1360      1350  0  12:55 tty1        00:00:00 ps -f
vlad@10:~$

```

Рис. 14 – Запустим несколько процессов в фоне с помощью оболочки
bash

```

vlad         518       412  0  05:02 tty1        00:00:00 -bash
vlad         649       648  0  06:20 tty1        00:00:00 -bash
vlad         658       649  0  06:24 tty1        00:00:00 vi 1
vlad         660       649  0  06:29 tty1        00:00:00 vi 1
vlad         666       649  0  06:36 tty1        00:00:00 vi 1
vlad         671       649  0  06:38 tty1        00:00:00 vi 1.txt
vlad         730       729  0  07:09 tty1        00:00:00 -bash
vlad         766       765  0  07:31 tty1        00:00:00 -bash
vlad        1062      1061  0  09:30 tty1        00:00:00 -bash
vlad        1229      1228  0  11:52 tty1        00:00:00 -bash
vlad        1329      1328  0  12:33 tty1        00:00:00 -bash
vlad        1350      1329  0  12:51 tty1        00:00:00 bash
vlad        1355      1350  9  12:53 tty1        00:00:21 bash
vlad        1356      1350  1  12:54 tty1        00:00:02 bash loop
vlad        1358      1350  0  12:54 tty1        00:00:01 bash loop2
vlad        1361      1350  2  12:56 tty1        00:00:02 bash loop
vlad        1362      1350  0  12:57 tty1        00:00:00 ps -f
vlad@10:~$ ps -f
UID          PID     PPID  C  STIME TTY          TIME CMD
vlad         518       412  0  05:02 tty1        00:00:00 -bash
vlad         649       648  0  06:20 tty1        00:00:00 -bash
vlad         658       649  0  06:24 tty1        00:00:00 vi 1
vlad         660       649  0  06:29 tty1        00:00:00 vi 1
vlad         666       649  0  06:36 tty1        00:00:00 vi 1
vlad         671       649  0  06:38 tty1        00:00:00 vi 1.txt
vlad         730       729  0  07:09 tty1        00:00:00 -bash
vlad         766       765  0  07:31 tty1        00:00:00 -bash
vlad        1062      1061  0  09:30 tty1        00:00:00 -bash
vlad        1229      1228  0  11:52 tty1        00:00:00 -bash
vlad        1329      1328  0  12:33 tty1        00:00:00 -bash
vlad        1350      1329  0  12:51 tty1        00:00:00 bash
vlad        1355      1350  8  12:53 tty1        00:00:21 bash
vlad        1356      1350  1  12:54 tty1        00:00:02 bash loop
vlad        1358      1350  0  12:54 tty1        00:00:01 bash loop2
vlad        1361      1350  2  12:56 tty1        00:00:02 bash loop
vlad        1363      1350  0  12:57 tty1        00:00:00 ps -f
vlad@10:~$

```

Рис. 15 – После нескольких запусков и остановок процессов
рассмотрим результаты командой ps -f

2 этап

1. Запустить в консоли на выполнение три задачи, две в интерактивном режиме, одну - в фоновом. Запустим задачи с помощью команд (рис. 16):

- sh loop.sh (интерактивный режим)
- sh loop.sh (интерактивный режим)
- sh loop.sh (фоновый режим)

```
vlad@10:~$ sh loop
^Z
[3]+  Остановлен      sh loop
vlad@10:~$ sh loop
^Z
[4]+  Остановлен      sh loop
vlad@10:~$ sh loop&
[5] 846
vlad@10:~$ ps -f
UID          PID    PPID  C  STIME TTY          TIME CMD
vlad          538      455  0   08:29 tty1        00:00:00 -bash
vlad          768      767  0   13:29 tty1        00:00:00 -bash
vlad          781      768  0   13:32 tty1        00:00:00 sh
vlad          829      768  0   14:03 tty1        00:00:00 top
vlad          844      768  22  14:19 tty1        00:00:07 sh loop
vlad          845      768  5   14:20 tty1        00:00:01 sh loop
vlad          846      768  87  14:20 tty1        00:00:04 sh loop
vlad          847      768  0   14:20 tty1        00:00:00 ps -f
vlad@10:~$
```

Рис. 16 – Запуск задач

С помощью команды jobs получим список процессов в текущей оболочке (рис. 17):

- jobs (список процессов в текущей оболочке)

```
vlad@10:~$ jobs -l
[1]      781  Остановлено (ввод с терминала)                sh
[2]      829  Остановлено (сигнал)                             top
[3]-     844  Остановлено      sh loop
[4]+     845  Остановлено      sh loop
[5]      846  Запущен           sh loop &
vlad@10:~$ _
```

Рис. 17 – Использование команды jobs

2. Перевести одну из задач, выполняющихся в интерактивном режиме, в фоновый режим. С помощью команды bg и номера, присвоенного задаче командной оболочкой при остановке ее исполнения, переведем задачу из интерактивного режима в фоновый:

- bg %1 (возобновление задачи 1)

```

vlad@10:~$ fg 5
sh loop
^Z
[5]+  Остановлен      sh loop
vlad@10:~$ jobs
[1]  Остановлен      sh
[2]  Остановлен      top
[3]  Запущен         sh loop &
[4]  Запущен         sh loop &
[5]+  Остановлен      sh loop
[6]-  Остановлен      sh loop
vlad@10:~$ fg 3
sh loop
^Z
[3]+  Остановлен      sh loop
vlad@10:~$ fg 4
sh loop
^Z
[4]+  Остановлен      sh loop
vlad@10:~$ jobs
[1]  Остановлен      sh
[2]  Остановлен      top
[3]-  Остановлен      sh loop
[4]+  Остановлен      sh loop
[5]  Остановлен      sh loop
[6]  Остановлен      sh loop
vlad@10:~$ _

```

Рис. 18 – Перевод задачи в фоновый режим

3. 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.

Создадим именованный канал для архивирования с помощью команды `mkfifo`. Посмотрим, что получилось в результате работы команды с помощью `ls -l`.

- `mkfifo myBlog` (создание именованного канала с именем «myBlog»)
- `ls -l myBlog` (проверка создания файла)
- `gzip -9 -c < myBlog > out.gz` (передача дом. каталога)
- `zcat out.gz` (просмотр сжатых файлов)
- `tar -cvf out.tar /home > myBlog` (передача каталога с подкаталогами и файлами)
- `zcat out.gz` (просмотр сжатых файлов)

```

vlad@10:~$ mkfifo Chanel
vlad@10:~$ ls -l Chanel
prw-r--r-- 1 vlad vlad 0 ноя 30 17:15 Chanel
vlad@10:~$ _

```

Рис. 19 – Создание именованного канала и его проверка

```

vlad@10:~$ su
Пароль:
root@10:/home/vlad# gzip -9 -c < Chanel > out.gz &
[1] 901
root@10:/home/vlad# ls -R > Chanel
[1]+  Завершён      gzip -9 -c < Chanel > out.gz
root@10:/home/vlad# zcat out.gz
.:
Chanel
loop
loop2
loop3
out.gz
root@10:/home/vlad#

```

Рис. 20 – Передача списка файлов домашнего каталога пользователя
vlad и проверка

```

root@10:/home/vlad# mkdir DIR
root@10:/home/vlad# cd DIR/
root@10:/home/vlad/DIR# mkdir DIR1
root@10:/home/vlad/DIR# mkdir DIR2
root@10:/home/vlad/DIR# mkdir DIR3
root@10:/home/vlad/DIR# touch FILE1
root@10:/home/vlad/DIR#
&      123      cd      DIR2      gzip      -l      ps
-      1306     Chanel  DIR3      h      -la      -R
/      -9      DIR      -f      /home/vlad  ls      stty
<      -all    DIR/     FILE1    htop     mkdir    su
>      -c      DIR1     gtop     kill     out.gz   top
root@10:/home/vlad/DIR# cd DIR1
root@10:/home/vlad/DIR/DIR1# touch FILE2
root@10:/home/vlad/DIR/DIR1# cd ..
root@10:/home/vlad/DIR# tou FILE3
bash: tou: команда не найдена
root@10:/home/vlad/DIR# touch FILE3
root@10:/home/vlad/DIR# ls -l
итого 12
drwxr-xr-x 2 root root 4096 ноя 30 17:22 DIR1
drwxr-xr-x 2 root root 4096 ноя 30 17:21 DIR2
drwxr-xr-x 2 root root 4096 ноя 30 17:21 DIR3
-rw-r--r-- 1 root root   0 ноя 30 17:21 FILE1
-rw-r--r-- 1 root root   0 ноя 30 17:23 FILE3
root@10:/home/vlad/DIR# cd ..
root@10:/home/vlad# gzip -9 -c < Chanel > out.gz &
[1] 916
root@10:/home/vlad# tar -cvf out.tar DIR > Chanel
[1]+  Завершён      gzip -9 -c < Chanel > out.gz
root@10:/home/vlad# _

```

Рис. 21 – Создание нового каталога с файлами и подкаталогами

```
root@10:/home/vlad# gzip -9 -c < Chanel > out.gz &  
[1] 919  
root@10:/home/vlad# tar -cvf out.tar DIR > Chanel  
[1]+  Завершён      gzip -9 -c < Chanel > out.gz  
root@10:/home/vlad# _
```

Рис. 22 – Передача всего каталога с подкаталогами

```
root@10:/home/vlad# zcat out.gz  
DIR/  
DIR/FILE1  
DIR/DIR2/  
DIR/FILE3  
DIR/DIR3/  
DIR/DIR1/  
DIR/DIR1/FILE2  
root@10:/home/vlad# _
```

Рис. 23 – Результаты

3 этап

1. Отобразить информацию о процессах указанного пользователя в виде иерархии, вывод отсортировать по значениям PID.

Используя команду `top` мы сможем увидеть все запущенные процессы.

Для того чтобы выбрать указанного пользователя нужно будет нажать “u”, после чего ввести имя пользователя, процессы которого нужно посмотреть.

Далее для отсортировки по PID нажмем “f” для перехода в управление полями, затем нажав “a” выберем сортировку по PID.

```
top - 18:18:12 up 2:26, 1 user, load average: 1,00, 1,00, 1,00
Tasks: 85 total, 2 running, 80 sleeping, 3 stopped, 0 zombie
%Cpu(s):100,0 us, 0,0 sy, 0,0 ni, 0,0 id, 0,0 wa, 0,0 hi, 0,0 si, 0,0 st
MiB Mem : 1983,2 total, 1801,7 free, 88,7 used, 92,7 buff/cache
MiB Swap: 975,0 total, 975,0 free, 0,0 used. 1774,1 avail Mem
```

PID	PPID	TIME+	%CPU	%MEM	PR	NI	S	VIRT	RES	UID	COMMAND
672	615	0:00.04	0,0	0,2	20	0	R	9972	3700	1000	top
667	615	0:00.04	0,0	0,2	20	0	T	9972	3700	1000	top
637	615	0:00.35	0,0	0,2	20	0	T	9972	3708	1000	top
615	614	0:00.03	0,0	0,2	20	0	S	8032	4752	1000	bash
550	538	142:06.50	99,7	0,0	20	0	R	2484	572	1000	sh
538	455	0:00.07	0,0	0,3	20	0	S	8360	5140	1000	bash
533	532	0:00.00	0,0	0,1	20	0	S	101212	2516	1000	(sd-pam)
532	1	0:00.12	0,0	0,4	20	0	S	15184	7784	1000	systemd

Рис. – Вывод иерархии процессов пользователя отсортированный по PID

2. С помощью сигнала SIGSTOP приостановить выполнение процесса, владельцем которого является текущий пользователь. Через несколько секунд возобновить выполнение процесса.

Запустим в фоновом режиме команду по печати строки “Hello!” с задержкой в секунду.

Далее остановим, используя сигнал SIGSTOP.

Через несколько секунд возобновим выполнение процесса командой SIGCONT.

```
vlad@10:~$ (sleep 60; echo "Hello!")&
[4] 817
vlad@10:~$ kill -SIGSTOP 817
vlad@10:~$ ps -f
UID          PID    PPID  C STIME TTY          TIME CMD
vlad          538      455  0 08:29 tty1        00:00:00 -bash
vlad          768      767  0 13:29 tty1        00:00:00 -bash
vlad          781      768  0 13:32 tty1        00:00:00 sh
vlad          784      768  64 13:32 tty1        00:10:00 sh loop
vlad          815      768  24 13:45 tty1        00:00:36 sh loop3
vlad          817      768  0 13:47 tty1        00:00:00 -bash
vlad          818      817  0 13:47 tty1        00:00:00 sleep 60
vlad          819      768  0 13:48 tty1        00:00:00 ps -f

[4]+  Остановлен    ( sleep 60; echo "Hello!" )
vlad@10:~$ kill -SIGCONT 817
-bash: kill: SIGCONT: недопустимое указание сигнала
vlad@10:~$ kill -SIGSTOP 817
vlad@10:~$ Hello!
```

Рис. – Остановка процесса используя сигнал SIGSTOP и возобновление используя сигнал SIGCONT

3. Определить идентификаторы и имена процессов, не связанных с указанным терминалом.

Команда для определения идентификатора и имени процесса, не связанных с указанным терминалом “ps -ef”.

```
root      284      2  0 08:29 ?        00:00:00 [ttm_swap]
root      287      2  0 08:29 ?        00:00:00 [card0-crtc0]
root      291      2  0 08:29 ?        00:00:00 [card0-crtc1]
root      294      2  0 08:29 ?        00:00:00 [card0-crtc2]
root      297      2  0 08:29 ?        00:00:00 [card0-crtc3]
root      300      2  0 08:29 ?        00:00:00 [card0-crtc4]
root      303      2  0 08:29 ?        00:00:00 [card0-crtc5]
systemd+  304      1  0 08:29 ?        00:00:00 /lib/systemd/systemd-timesyncd
root      308      2  0 08:29 ?        00:00:00 [card0-crtc6]
root      313      2  0 08:29 ?        00:00:00 [card0-crtc7]
root      349      1  0 08:29 ?        00:00:00 /sbin/dhclient -4 -v -i -pf /run/dhclient.enp0s
root      428      1  0 08:29 ?        00:00:00 /usr/sbin/cron -f
message+  429      1  0 08:29 ?        00:00:00 /usr/bin/dbus-daemon --system --address=systemd
root      436      1  0 08:29 ?        00:00:00 /usr/sbin/rsyslogd -n -iNONE
root      438      1  0 08:29 ?        00:00:00 /lib/systemd/systemd-logind
root      440      1  0 08:29 ?        00:00:00 /sbin/wpa_supplicant -u -s -O /run/wpa_supplika
root      455      1  0 08:29 tty1      00:00:00 /bin/login -p --
root      464      1  0 08:29 ?        00:00:00 sshd: /usr/sbin/sshd -D [listener] 0 of 10-100
root      468      1  0 08:29 ?        00:00:01 /usr/sbin/apache2 -k start
www-data  470      468  0 08:29 ?        00:00:00 /usr/sbin/apache2 -k start
www-data  471      468  0 08:29 ?        00:00:00 /usr/sbin/apache2 -k start
vlad      532      1  0 08:29 ?        00:00:00 /lib/systemd/systemd --user
vlad      533      532  0 08:29 ?        00:00:00 (sd-pam)
vlad      538      455  0 08:29 tty1      00:00:00 -bash
root      600      538  0 10:23 tty1      00:00:00 su
root      601      600  0 10:23 tty1      00:00:00 bash
root      603      601  0 10:23 tty1      00:00:00 top
root      746      2  0 12:48 ?        00:00:00 [kworker/u2:2-events_unbound]
root      759      2  0 13:17 ?        00:00:00 [kworker/u2:0-flush-8:0]
root      767      601  0 13:29 tty1      00:00:00 su - vlad
vlad      768      767  0 13:29 tty1      00:00:00 -bash
vlad      781      768  0 13:32 tty1      00:00:00 sh
root      811      2  0 13:40 ?        00:00:01 [kworker/0:0-events]
root      814      2  0 13:45 ?        00:00:00 [kworker/0:1-ata_sff]
root      820      2  0 13:51 ?        00:00:00 [kworker/0:2-ata_sff]
vlad      823      768  0 13:53 tty1      00:00:00 ps -ef
vlad@10:~$
```

Рис. – Определим идентификаторы и имена процессов, не связанных с указанным терминалом

Вывод

В ходе выполнения данной лабораторной работы мной были получены знания о понятии процесса, приобретен опыт и навыки управления процессами в операционной системе Linux.