

# **Липецкий государственный технический университет**

Факультет автоматизации и информатики

Автоматизированные системы управления

## **ЛАБОРАТОРНАЯ РАБОТА №3**

по курсу ДПО Интаро - Linux

«Процессы и демоны»

Вариант 4.

Студент

Митина М. В.

Группа     ПИ-20-1

Руководитель  
доц.

Кургасов В.В.

Липецк 2022 г.

## Цель работы

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

## Задание кафедры

*Для части 1:*

- 1) Загрузиться не root, а пользователем.
- 2) Найти файл с образом ядра. Выяснить по имени файла номер версии Linux.
- 3) Посмотреть процессы `ps -f`. Прокомментировать. Для этого почитать `man ps`.
- 4) Написать с помощью редактора `vi` два сценария `loop` и `loop2`. Текст сценариев:
  - a. `Loop: while true; do true; done`
  - b. `Loop2: while true; do true; echo 'Hello'; 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`

*Для части 2:*

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.
4. Создать именованный канал для архивирования и осуществить передачу в канал о списка файлов домашнего каталога вместе с подкаталогами (ключ `-R`), о 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.

*Для части 3:*

#### ***Вариант 4.***

1. Сгенерировать следующую информацию — полный литинг в длинном формате о процессах текущего пользователя: PID, PPID, выделенное время ЦП, время запуска, размер образа.
2. С помощью сигнала SIGTSTP (используя комбинацию клавиш и команду `kill`) приостановить выполнение процесса, владельцем которого является

текущий пользователь. Через несколько секунд возобновить выполнение процесса.

3. Определить идентификатор и имя процесса, созданного последним пользователем root.

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.

## Оглавление

1. Ход работы.....	6
1.1. Часть 1 .....	6
1.2. Часть 2. ....	11
1.3. Часть 3. Вариант 4.....	14
Вывод.....	17

## 1. Ход работы

### 1.1. Часть 1

```
mitina_mv@debian:~$ su - root
Пароль:
root@debian:~# _
```

Рисунок 1. Авторизация под root

```
root@debian ~ [1]# find /boot/vmlinuz-5.10.0-19-amd64
/boot/vmlinuz-5.10.0-19-amd64
```

Рисунок 2. Поиск файла с образом ядра. По названию видим, что ядро текущей системы версии 5.8.0-44-generic.

```
root@debian ~# uname
Linux
root@debian ~# uname -a
Linux debian 5.10.0-19-amd64 #1 SMP Debian 5.10.149-2 (2022-10-21) x86_64 GNU/Linux
```

Рисунок 3. Подтвердим версию ядра ответом команды uname.

```
root@debian ~# ps -f
```

UID	PID	PPID	C	STIME	TTY	TIME	CMD
root	3260	1	0	17:00	tty2	00:00:00	/bin/login -p --
root	3594	3590	0	17:12	tty2	00:00:00	sudo -i
root	3595	3594	0	17:12	tty2	00:00:00	-bash
root	3600	3595	0	17:12	tty2	00:00:00	sudo -s /usr/bin/fish
root	3601	3600	0	17:12	tty2	00:00:00	/usr/bin/fish
root	3876	3601	0	17:34	tty2	00:00:00	ps -f

Рисунок 4. Просмотр процессов командой ps -f.

Команда `ps` показывает нам списки процессов (запущенных в памяти экземпляров программ) и информацию по ним (PID, PPID, Nice, TIME и т.д.). Синтаксис команды: `$ ps опции`. В варианте использования команды по рисунку 3 опция `-f` добавляет форматирование представляемых данных, а именно: вывод максимума доступных данных, например, количество потоков. По умолчанию, команда покажет процессы, принадлежащие текущему пользователю, но это можно изменить через ключ `-u`. На рисунке мы видим следующие колонки:

- **UID** - пользователь, от имени которого запущен процесс;
- **PID** - идентификатор процесса;
- **PPID** - идентификатор родительского процесса;
- **C** - процент времени CPU, используемого процессом;
- **STIME** - время запуска процесса;

- **TTY** - терминал, из которого запущен процесс;
- **TIME** - общее время процессора, затраченное на выполнение процессора;
- **CMD** - команда запуска процессора;



Рисунок 5. в редакторе nano написала содержимое файла loop.


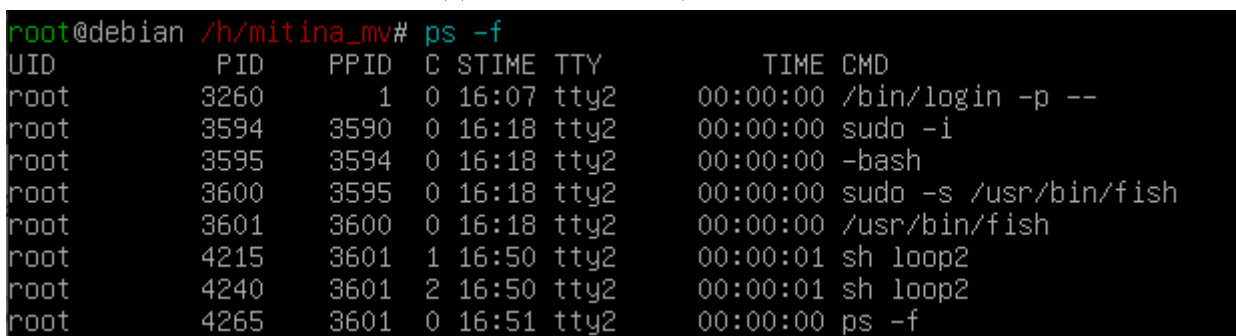


Рисунок 6. в редакторе nano написала содержимое файла loop2.



Рисунок 7. команда запуска файла loop2, выше его вывод и приостановление задачи комбинацией CTRL+Z



UID	PID	PPID	C	STIME	TTY	TIME	CMD
root	3260	1	0	16:07	tty2	00:00:00	/bin/login -p --
root	3594	3590	0	16:18	tty2	00:00:00	sudo -i
root	3595	3594	0	16:18	tty2	00:00:00	-bash
root	3600	3595	0	16:18	tty2	00:00:00	sudo -s /usr/bin/fish
root	3601	3600	0	16:18	tty2	00:00:00	/usr/bin/fish
root	4215	3601	1	16:50	tty2	00:00:01	sh loop2
root	4240	3601	2	16:50	tty2	00:00:01	sh loop2
root	4265	3601	0	16:51	tty2	00:00:00	ps -f

Рисунок 8. просмотр текущих задач машины. Видим два процесса sh loop2 – это запускала я. видим их PID, это поможет их убить.

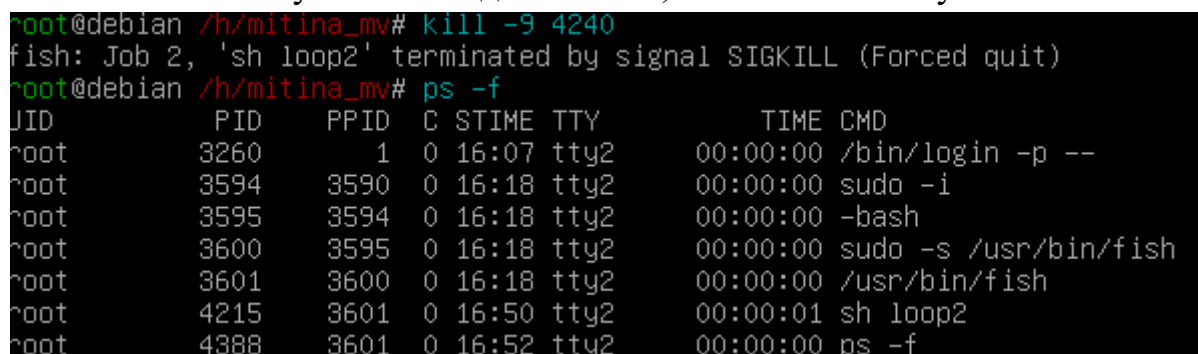


Рисунок 9. убила процесс с PID 4240.

```
root@debian /h/mitina_mv# sh loop&
```

Рисунок 10. запускаем процесс loop в фоне.

```
root@debian /h/mitina_mv# ps -f
UID      PID     PPID  C  STIME TTY          TIME CMD
root      3260         1  0  16:07 tty2        00:00:00 /bin/login -p --
root      3594     3590  0  16:18 tty2        00:00:00 sudo -i
root      3595     3594  0  16:18 tty2        00:00:00 -bash
root      3600     3595  0  16:18 tty2        00:00:00 sudo -s /usr/bin/fish
root      3601     3600  0  16:18 tty2        00:00:00 /usr/bin/fish
root      4215     3601  0  16:50 tty2        00:00:01 sh loop2
root      4424     3601 34  17:07 tty2        00:00:01 sh loop
root      4442     3601  0  17:07 tty2        00:00:00 ps -f
root@debian /h/mitina_mv# ps -f
UID      PID     PPID  C  STIME TTY          TIME CMD
root      3260         1  0  16:07 tty2        00:00:00 /bin/login -p --
root      3594     3590  0  16:18 tty2        00:00:00 sudo -i
root      3595     3594  0  16:18 tty2        00:00:00 -bash
root      3600     3595  0  16:18 tty2        00:00:00 sudo -s /usr/bin/fish
root      3601     3600  0  16:18 tty2        00:00:00 /usr/bin/fish
root      4215     3601  0  16:50 tty2        00:00:01 sh loop2
root      4424     3601 32  17:07 tty2        00:00:13 sh loop
root      4459     3601  0  17:08 tty2        00:00:00 ps -f
```

Рисунок 11. Видим, что процесс sh loop потребляет ресурсы. колонка C – процент времени CPU – видим, что процессы активны и сейчас выполняются.

```
root@debian /h/mitina_mv# kill -15 4424
root@debian /h/mitina_mv# ps -f
UID      PID     PPID  C  STIME TTY          TIME CMD
root      3260         1  0  16:07 tty2        00:00:00 /bin/login -p --
root      3594     3590  0  16:18 tty2        00:00:00 sudo -i
root      3595     3594  0  16:18 tty2        00:00:00 -bash
root      3600     3595  0  16:18 tty2        00:00:00 sudo -s /usr/bin/fish
root      3601     3600  0  16:18 tty2        00:00:00 /usr/bin/fish
root      4215     3601  0  16:50 tty2        00:00:01 sh loop2
root      4522     3601  0  17:30 tty2        00:00:00 ps -f
fish: Job 2, 'sh loop&' terminated by signal SIGTERM (Polite quit request)
```

Рисунок 12. убила процесс sh loop и отобразила результат через команду ps. С сигналом 15 процессы завершаются мягко, сохраняя свои данные, без потерь.

```
root@debian /h/mitina_mv# sh loop&
root@debian /h/mitina_mv# ps -f
UID      PID     PPID  C  STIME TTY          TIME CMD
root      3260         1  0  16:07 tty2        00:00:00 /bin/login -p --
root      3594     3590  0  16:18 tty2        00:00:00 sudo -i
root      3595     3594  0  16:18 tty2        00:00:00 -bash
root      3600     3595  0  16:18 tty2        00:00:00 sudo -s /usr/bin/fish
root      3601     3600  0  16:18 tty2        00:00:00 /usr/bin/fish
root      4215     3601  0  16:50 tty2        00:00:01 sh loop2
root      4550     3601 34  17:35 tty2        00:00:04 sh loop
root      4575     3601  0  17:35 tty2        00:00:00 ps -f
root@debian /h/mitina_mv# kill -9 4550
```

Рисунок 13. запустила в фоне sh loop&, посмотрела его PID и убила процесс с сигналом -9.



```

root@debian /h/mitina_mv# kill -9 4550
root@debian /h/mitina_mv# sleep 1200&
fish: Job 2, 'sh loop&' terminated by signal SIGKILL (Forced quit)
root@debian /h/mitina_mv# sleep 1000&
root@debian /h/mitina_mv# ps -f
UID          PID    PPID  C STIME TTY          TIME CMD
root          3260      1   0 16:07 tty2        00:00:00 /bin/login -p --
root          3594     3590   0 16:18 tty2        00:00:00 sudo -i
root          3595     3594   0 16:18 tty2        00:00:00 -bash
root          3600     3595   0 16:18 tty2        00:00:00 sudo -s /usr/bin/fish
root          3601     3600   0 16:18 tty2        00:00:00 /usr/bin/fish
root          4215     3601   0 16:50 tty2        00:00:01 sh loop2
root          4630     3601   0 17:41 tty2        00:00:00 sleep 1200
root          4654     3601   0 17:41 tty2        00:00:00 sleep 1000
root          4672     3601   0 17:41 tty2        00:00:00 ps -f
root@debian /h/mitina_mv# sh loop
^Z^J
root@debian /h/mitina_mv# ps -f
UID          PID    PPID  C STIME TTY          TIME CMD
root          3260      1   0 16:07 tty2        00:00:00 /bin/login -p --
root          3594     3590   0 16:18 tty2        00:00:00 sudo -i
root          3595     3594   0 16:18 tty2        00:00:00 -bash
root          3600     3595   0 16:18 tty2        00:00:00 sudo -s /usr/bin/fish
root          3601     3600   0 16:18 tty2        00:00:00 /usr/bin/fish
root          4215     3601   0 16:50 tty2        00:00:01 sh loop2
root          4630     3601   0 17:41 tty2        00:00:00 sleep 1200
root          4654     3601   0 17:41 tty2        00:00:00 sleep 1000
root          4694     3601  16 17:42 tty2        00:00:01 sh loop
root          4711     3601   0 17:42 tty2        00:00:00 ps -f

```

Рисунок 14. запустили 2 процесса в фоне, один процесс на переднем плане (его остановили). посмотрели задачи системы и потребляемые ресурсы.

```

root@debian /h/mitina_mv# jobs
Job          Group    CPU    State  Command
5            4694     0%     stopped sh loop
4            4654     0%     running sleep 1000&
3            4630     0%     running sleep 1200&
1            4215     0%     stopped sh loop2
root@debian /h/mitina_mv# fg 5
fg: No suitable job: 5
root@debian /h/mitina_mv [2]# bg 1
bg: Could not find job '1'
root@debian /h/mitina_mv# bg sh loop
bg: 'sh' is not a valid job specifier
bg: 'loop' is not a valid job specifier
root@debian /h/mitina_mv [2]# bg 4694
Send job 5 'sh loop' to background

```

Рисунок 15. посмотрели задачи системы через команду jobs. попытки возобновить задачу с идентификатором 5, последняя запись успешна.

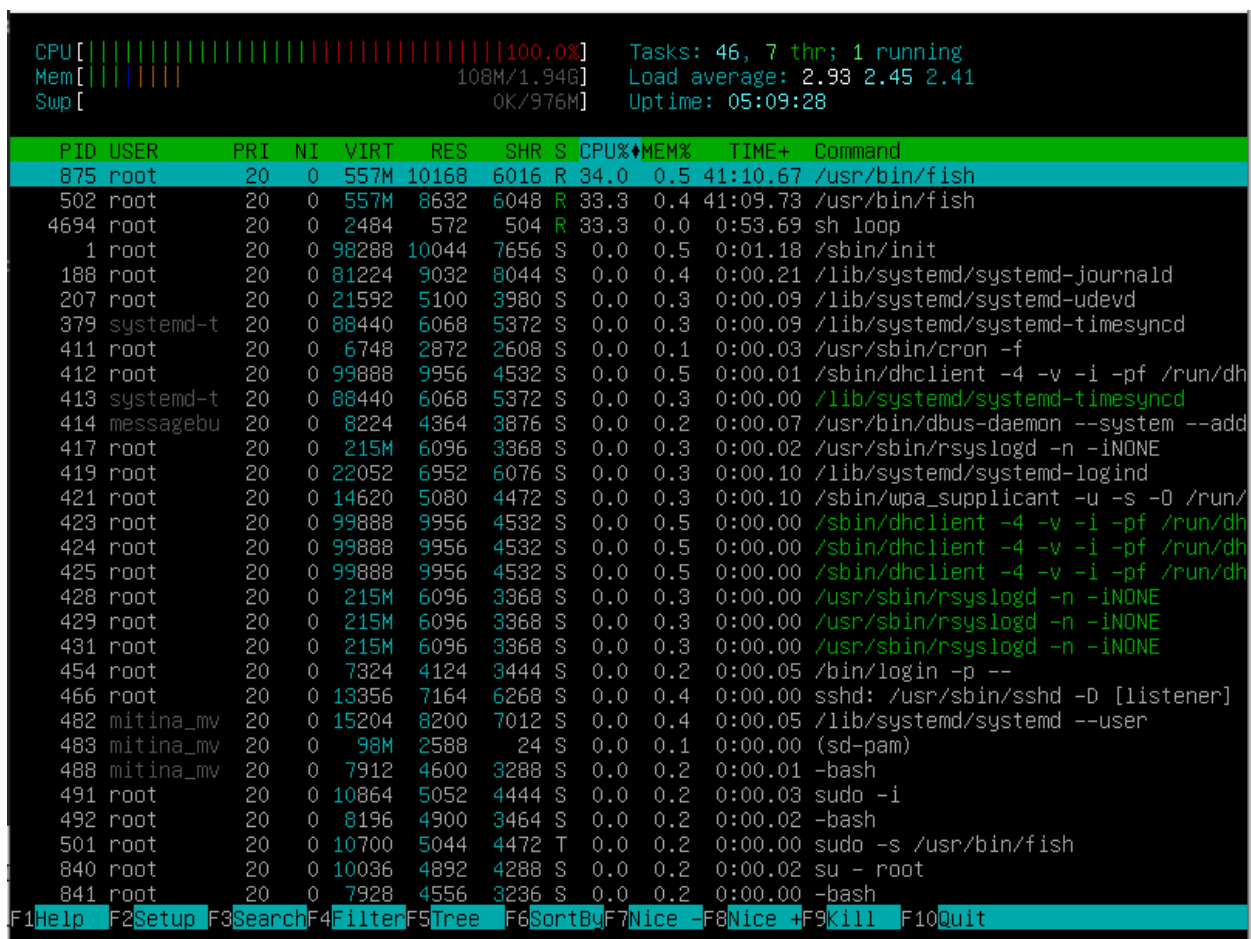


Рисунок 16.установлен пакет htop, просмотр некоторых задач через утилиту.

Утилита htop – монитор процессов – предназначена для детального мониторинга системы, отображающая процессы и состояние системы в реальном времени. Мы можем наблюдать изменение нагрузки CPU и MEM, смену состояний процессов, изменение времени работы и прочего

## 1.2. Часть 2.

```
root@debian /h/mitina_mv# sleep 1200&
root@debian /h/mitina_mv# sh loop
^C^J
Job 4, 'sleep 1000&' has ended

root@debian /h/mitina_mv [SIGINT]# jobs
Job      Group   CPU    State  Command
5        4951    0%     running sleep 1200&
5        4694    33%    running sh loop
3        4630    0%     running sleep 1200&
1        4215    0%     stopped sh loop2

root@debian /h/mitina_mv# ps -f
UID        PID     PPID  C STIME TTY          TIME CMD
root        3260      1  0 16:07 tty2      00:00:00 /bin/login -p --
root        3594    3590  0 16:18 tty2      00:00:00 sudo -i
root        3595    3594  0 16:18 tty2      00:00:00 -bash
root        3600    3595  0 16:18 tty2      00:00:00 sudo -s /usr/bin/fish
root        3601    3600  0 16:18 tty2      00:00:00 /usr/bin/fish
root        4215    3601  0 16:50 tty2      00:00:01 sh loop2
root        4630    3601  0 17:41 tty2      00:00:00 sleep 1200
root        4694    3601 27 17:42 tty2      00:04:41 sh loop
root        4951    3601  0 17:57 tty2      00:00:00 sleep 1200
root        5054    3601  0 17:59 tty2      00:00:00 ps -f

root@debian /h/mitina_mv# fg 4694
Send job 5, 'sh loop' to foreground
^Z^J

root@debian /h/mitina_mv# jobs
Job      Group   CPU    State  Command
5        4694    0%     stopped sh loop
5        4951    0%     running sleep 1200&
3        4630    0%     running sleep 1200&
1        4215    0%     stopped sh loop2
```

Рисунок 17. запуск нескольких процессов, перевод из фонового выполнения на передний план, приостановка задачи.

```
root@debian /h/mitina_mv# ls -R | tar -cvf ath.tar
tar: Робкий отказ от создания пустого архива
Попробуйте «tar --help» или «tar --usage» для
получения более подробного описания.
root@debian /h/mitina_mv [SIGPIPE|2]# tar -cvf ath.tar < ls -R
[W] fish: An error occurred while redirecting file 'ls'
open: Нет такого файла или каталога
root@debian /h/mitina_mv [SIGPIPE|2]# cat lsR.txt < ls -R
[W] fish: An error occurred while redirecting file 'ls'
open: Нет такого файла или каталога
root@debian /h/mitina_mv [SIGPIPE|2]# ls -R > lsR.txt
root@debian /h/mitina_mv# ls
f1.txt  loop  loop2  lsR.txt
```

Рисунок 18. Попытка создать архив списка файлов домашнего каталога с помощью именованного потока, через перенаправление вывода команды `ls -R`.

```

root@debian /h/mitina_mv# tar -cvf ath.tar < lsR.txt
tar: Робкий отказ от создания пустого архива
Попробуйте «tar --help» или «tar --usage» для
получения более подробного описания.
root@debian /h/mitina_mv [2]# tar -cvf ath.tar | lsR.txt
fish: Unknown command: lsR.txt
fish:
tar -cvf ath.tar | lsR.txt
^

root@debian /h/mitina_mv [127]# tar -cvf ath.tar | ls -R
tar: Робкий отказ от создания пустого архива
Попробуйте «tar --help» или «tar --usage» для
получения более подробного описания.
.:
f1.txt loop loop2 lsR.txt
root@debian /h/mitina_mv [2|0]# ls -R > tar -cvf arh.tar
ls: невозможно получить доступ к 'arh.tar': Нет такого файла или каталога
root@debian /h/mitina_mv [2]# tar -cvf ath.tar | cat lsR.txt
.:
f1.txt
loop
loop2
lsR.txt
tar: Робкий отказ от создания пустого архива
Попробуйте «tar --help» или «tar --usage» для
получения более подробного описания.

```

Рисунок 19. создала файл lsR.txt, где сохранила названия файлов, попыталась снова выполнить задачу.

```

root@debian /h/mitina_mv# mkfifo kanal
root@debian /h/mitina_mv# ls -R > kanal
^Z
jdksf
q
:q
tar -cvf
asdasdasd
^C<H> fish: An error occurred while redirecting file 'kanal'
open: Прерван системный вызов

```

Рисунок 20. создала именованный канал, но перенаправление вывода не получилось.

```

mitina_mv@debian:~$ mkfifo mashKanal
mitina_mv@debian:~$ ls -R > mashKanal &
[1] 6163
mitina_mv@debian:~$ tar -cvf arh.tar < mashKanal
tar: Робкий отказ от создания пустого архива
Попробуйте «tar --help» или «tar --usage» для
получения более подробного описания.
[1]+  Завершён          ls --color=auto -R > mashKanal
mitina_mv@debian:~$ tar -cvf arh.tar < mashKanal >
-bash: синтаксическая ошибка рядом с неожиданным маркером «newline»
mitina_mv@debian:~$ gzip -9 -c < mashKanal > outer.gz

^C
mitina_mv@debian:~$ gzip -9 -c < mashKanal > outer.gz &
[1] 6167
mitina_mv@debian:~$ jobs
[1]+  Запущен          gzip -9 -c < mashKanal > outer.gz &
mitina_mv@debian:~$ zcat outer.gz
.:
ath.tar
f1.txt
kanal
kanal2
kanal3
loop
loop2
lsR1.txt
lsR.txt
outer.gz
tar

```

Рисунок 21. отключила оболочку fish, создала именованный канал, перенаправила в него вывод ls -R в фоновом режиме, ошибка с tar повторилась, но ошибки с gzip не возникло. Архив создан, показан через zcat.

### 1.3. Часть 3. Вариант 4.

```

CPU[|||||]100.0% Tasks: 44, 7 thr; 1 running
Mem[|||||]110M/1.94G Load average: 2.00 2.00 2.15
Swp[|||||]0K/976M Uptime: 05:43:05

  PID USER      PRI  NI  VIRT   RES   SHR  S  CPU% MEM%   TIME+  Command
  875 root        20    0  557M  10168  6016 R   50.7   0.5  55:58.11 /usr/bin/fish
    502 root        20    0  557M   8632  6048 R   50.0   0.4  55:57.18 /usr/bin/fish
      1 root        20    0  98288  10044  7656 S    0.0   0.5   0:01.19 /sbin/init
    188 root        20    0  81224   9036  8048 S    0.0   0.4   0:00.22 /lib/systemd/systemd-journald
    207 root        20    0  21592   5100  3980 S    0.0   0.3   0:00.09 /lib/systemd/systemd-udev
    379 systemd-t    20    0  88440   6068  5372 S    0.0   0.3   0:00.10 /lib/systemd/systemd-timesyncd
    411 root        20    0   6748   2872  2608 S    0.0   0.1   0:00.03 /usr/sbin/cron -f
    412 root        20    0  99888   9956  4532 S    0.0   0.5   0:00.01 /sbin/dhclient -4 -v -i -pf /run/dh
    413 systemd-t    20    0  88440   6068  5372 S    0.0   0.3   0:00.00 /lib/systemd/systemd-timesyncd
    414 messagebu   20    0   8224   4364  3876 S    0.0   0.2   0:00.07 /usr/bin/dbus-daemon --system --add
    417 root        20    0   215M   6096  3368 S    0.0   0.3   0:00.02 /usr/sbin/rsyslogd -n -iNONE
    419 root        20    0  22052   6952  6076 S    0.0   0.3   0:00.11 /lib/systemd/systemd-logind
    421 root        20    0  14620   5080  4472 S    0.0   0.3   0:00.12 /sbin/wpa_supplicant -u -s -O /run/
    423 root        20    0  99888   9956  4532 S    0.0   0.5   0:00.00 /sbin/dhclient -4 -v -i -pf /run/dh
    424 root        20    0  99888   9956  4532 S    0.0   0.5   0:00.00 /sbin/dhclient -4 -v -i -pf /run/dh
    425 root        20    0  99888   9956  4532 S    0.0   0.5   0:00.00 /sbin/dhclient -4 -v -i -pf /run/dh
    428 root        20    0   215M   6096  3368 S    0.0   0.3   0:00.00 /usr/sbin/rsyslogd -n -iNONE
    429 root        20    0   215M   6096  3368 S    0.0   0.3   0:00.00 /usr/sbin/rsyslogd -n -iNONE
    431 root        20    0   215M   6096  3368 S    0.0   0.3   0:00.00 /usr/sbin/rsyslogd -n -iNONE
    454 root        20    0   7324   4124  3444 S    0.0   0.2   0:00.05 /bin/login -p --
    466 root        20    0  13356   7164  6268 S    0.0   0.4   0:00.00 sshd: /usr/sbin/sshd -D [listener]
    482 mitina_mv    20    0   15204   8200  7012 S    0.0   0.4   0:00.05 /lib/systemd/systemd --user
    483 mitina_mv    20    0    98M    2588    24 S    0.0   0.1   0:00.00 (sd-pam)
    488 mitina_mv    20    0   7912   4600  3288 S    0.0   0.2   0:00.01 -bash
    491 root        20    0  10864   5052  4444 S    0.0   0.2   0:00.03 sudo -i
    492 root        20    0   8196   4900  3464 S    0.0   0.2   0:00.02 -bash
    501 root        20    0  10700   5044  4472 T    0.0   0.2   0:00.00 sudo -s /usr/bin/fish
    840 root        20    0  10036  4892  4288 S    0.0   0.2   0:00.02 su - root
    841 root        20    0   7928  4556  3236 S    0.0   0.2   0:00.00 -bash
    846 root        20    0  10044  4416  3848 S    0.0   0.2   0:00.00 su - MariaPI201

F1Help F2Setup F3Search F4Filter F5Tree F6SortBy F7Nice F8Nice +F9Kill F10Quit

```

Рисунок 22. процессы, показанные через утилиту htop.

```

CPU[|||||]100.0% Tasks: 44, 7 thr; 1 running
Mem[|||||]110M/1.94G Load average: 2.00 2.00 2.07
Swp[|||||]0K/976M Uptime: 05:52:11

Send signal:
 0 Cancel
 1 SIGHUP
 2 SIGINT
 3 SIGQUIT
 4 SIGILL
 5 SIGTRAP
 6 SIGABRT
 6 SIGIOT
 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
3260 root

  PID USER      PRI  NI  VIRT   RES   SHR  S  CPU% MEM%   TIME+  Command
    502 root        20    0  557M   8632  6048 R   49.7   0.4  1h00:29 /usr/bin/fish
    875 root        20    0  557M  10168  6016 R   49.7   0.5  1h00:30 /usr/bin/fish
      1 root        20    0  98288  10044  7656 S    0.0   0.5   0:01.19 /sbin/init
    188 root        20    0  81224   9036  8048 S    0.0   0.4   0:00.22 /lib/systemd/systemd-journald
    207 root        20    0  21592   5100  3980 S    0.0   0.3   0:00.09 /lib/systemd/systemd-udev
    411 root        20    0   6748   2872  2608 S    0.0   0.1   0:00.03 /usr/sbin/cron -f
    412 root        20    0  99888   9956  4532 S    0.0   0.5   0:00.01 /sbin/dhclient -4 -v -i -pf /run/dh
    417 root        20    0   215M   6096  3368 S    0.0   0.3   0:00.02 /usr/sbin/rsyslogd -n -iNONE
    419 root        20    0  22052   6952  6076 S    0.0   0.3   0:00.11 /lib/systemd/systemd-logind
    421 root        20    0  14620   5080  4472 S    0.0   0.3   0:00.12 /sbin/wpa_supplicant -u -s -O /run/
    423 root        20    0  99888   9956  4532 S    0.0   0.5   0:00.00 /sbin/dhclient -4 -v -i -pf /run/dh
    424 root        20    0  99888   9956  4532 S    0.0   0.5   0:00.00 /sbin/dhclient -4 -v -i -pf /run/dh
    425 root        20    0  99888   9956  4532 S    0.0   0.5   0:00.00 /sbin/dhclient -4 -v -i -pf /run/dh
    428 root        20    0   215M   6096  3368 S    0.0   0.3   0:00.00 /usr/sbin/rsyslogd -n -iNONE
    429 root        20    0   215M   6096  3368 S    0.0   0.3   0:00.00 /usr/sbin/rsyslogd -n -iNONE
    431 root        20    0   215M   6096  3368 S    0.0   0.3   0:00.00 /usr/sbin/rsyslogd -n -iNONE
    454 root        20    0   7324   4124  3444 S    0.0   0.2   0:00.05 /bin/login -p --
    466 root        20    0  13356   7164  6268 S    0.0   0.4   0:00.00 sshd: /usr/sbin/sshd -D [listener]
    491 root        20    0  10864   5052  4444 S    0.0   0.2   0:00.03 sudo -i
    492 root        20    0   8196   4900  3464 S    0.0   0.2   0:00.02 -bash
    501 root        20    0  10700   5044  4472 T    0.0   0.2   0:00.00 sudo -s /usr/bin/fi
    840 root        20    0  10036  4892  4288 S    0.0   0.2   0:00.02 su - root
    841 root        20    0   7928  4556  3236 S    0.0   0.2   0:00.00 -bash
    846 root        20    0  10044  4416  3848 S    0.0   0.2   0:00.00 su - MariaPI201
    854 root        20    0  10036  4872  4264 S    0.0   0.2   0:00.03 su - root
    855 root        20    0   7928  4616  3292 S    0.0   0.2   0:00.01 -bash
    859 root        20    0  10044  4364  3792 S    0.0   0.2   0:00.00 su - MariaPI201
    864 root        20    0  10068  4444  3856 S    0.0   0.2   0:00.03 su - mitina_mv
    874 root        20    0  10864   5200  4588 T    0.0   0.3   0:00.03 sudo -s /usr/bin/fi
    874 root        20    0   7304  4324  3664 S    0.0   0.2   0:00.03 /bin/login -p --

EnterSend EscCancel

```

Рисунок 23. Клавиша F9 позволяет перейти к осуществлению команды kill, и выбрать, какой сигнал отправить процессу.

Сигнал kill -9 прерывает процесс, завершает процесс сигнал -15 и останавливает сигнал -19.

```

CPU[|||||100.0%] Tasks: 44, 7 thr: 1 running
Mem[|||||110M/1.94G] Load average: 2.00 2.00 2.11
Swp[|||||0K/976M] Uptime: 05:46:35

Show processes of: PID USER PRI NI VIRT RES SHR S CPU% MEM% TIME+ Command
All users 502 root 20 0 557M 8632 6048 R 49.7 0.4 57:42.10 /usr/bin/fish
MariaPI201 875 root 20 0 557M 10168 6016 R 49.7 0.5 57:43.03 /usr/bin/fish
messagebus 5842 root 20 0 8276 4176 3416 R 0.7 0.2 0:00.35 htop
mitina_mv 1 root 20 0 98288 10044 7656 S 0.0 0.5 0:01.19 /sbin/init
root 188 root 20 0 81224 9036 8048 S 0.0 0.4 0:00.22 /lib/systemd/systemd-journald
systemd-timesync 207 root 20 0 21592 5100 3980 S 0.0 0.3 0:00.09 /lib/systemd/systemd-udevd
379 systemd-t 20 0 88440 6068 5372 S 0.0 0.3 0:00.10 /lib/systemd/systemd-logind
411 root 20 0 6748 2872 2608 S 0.0 0.1 0:00.03 /usr/sbin/cron -f
412 root 20 0 99888 9956 4532 S 0.0 0.5 0:00.01 /sbin/dhclient -4 -v -i -pf /run/dhclient-4
413 systemd-t 20 0 88440 6068 5372 S 0.0 0.3 0:00.00 /lib/systemd/systemd-logind
414 messagebu 20 0 8224 4364 3876 S 0.0 0.2 0:00.07 /usr/sbin/rsyslogd -n -iNONE
417 root 20 0 215M 6096 3368 S 0.0 0.3 0:00.02 /usr/sbin/rsyslogd -n -iNONE
419 root 20 0 22052 6952 6076 S 0.0 0.3 0:00.11 /lib/systemd/systemd-logind
421 root 20 0 14620 5080 4472 S 0.0 0.3 0:00.12 /sbin/wpa_supplicant -u -s -O /run/wpa_supplicant
423 root 20 0 99888 9956 4532 S 0.0 0.5 0:00.00 /sbin/dhclient -4 -v -i -pf /run/dhclient-4
424 root 20 0 99888 9956 4532 S 0.0 0.5 0:00.00 /sbin/dhclient -4 -v -i -pf /run/dhclient-4
425 root 20 0 99888 9956 4532 S 0.0 0.5 0:00.00 /sbin/dhclient -4 -v -i -pf /run/dhclient-4
428 root 20 0 215M 6096 3368 S 0.0 0.3 0:00.00 /usr/sbin/rsyslogd -n -iNONE
429 root 20 0 215M 6096 3368 S 0.0 0.3 0:00.00 /usr/sbin/rsyslogd -n -iNONE
431 root 20 0 215M 6096 3368 S 0.0 0.3 0:00.00 /usr/sbin/rsyslogd -n -iNONE
454 root 20 0 7324 4124 3444 S 0.0 0.2 0:00.05 /bin/login -p --
466 root 20 0 13356 7164 6268 S 0.0 0.4 0:00.00 sshd: /usr/sbin/sshd -D [listener] 4 of 128 max
482 mitina_mv 20 0 15204 8200 7012 S 0.0 0.4 0:00.05 /lib/systemd/systemd-logind
483 mitina_mv 20 0 98M 2588 24 S 0.0 0.1 0:00.00 (sd-pam)
488 mitina_mv 20 0 7912 4600 3288 S 0.0 0.2 0:00.01 -bash
491 root 20 0 10864 5052 4444 S 0.0 0.2 0:00.03 sudo -i
492 root 20 0 8196 4900 3464 S 0.0 0.2 0:00.02 -bash
501 root 20 0 10700 5044 4472 T 0.0 0.2 0:00.00 sudo -s /usr/bin/fish
840 root 20 0 10036 4892 4288 S 0.0 0.2 0:00.02 su - root
841 root 20 0 7928 4556 3236 S 0.0 0.2 0:00.00 -bash
846 root 20 0 10044 4416 3848 S 0.0 0.2 0:00.00 su - MariaPI201
854 root 20 0 10036 4872 4264 S 0.0 0.2 0:00.03 su - root
855 root 20 0 7928 4616 3292 S 0.0 0.2 0:00.01 -bash
859 root 20 0 10044 4364 3792 S 0.0 0.2 0:00.00 su - MariaPI201
864 root 20 0 10068 4444 3856 S 0.0 0.2 0:00.03 su - mitina_mv
874 root 20 0 10864 5200 4588 T 0.0 0.3 0:00.03 sudo -s /usr/bin/fish
3260 root 20 0 7304 4324 3664 S 0.0 0.2 0:00.03 /bin/login -p --
Enter Show Esc Cancel

```

Рисунок 24. для утилиты htop клавиша u позволяет показать процессы пользователя.

```

CPU[|||||100.0%] Tasks: 44, 7 thr: 1 running
Mem[|||||110M/1.94G] Load average: 2.00 2.00 2.11
Swp[|||||0K/976M] Uptime: 05:46:47

PID USER PRI NI VIRT RES SHR S CPU% MEM% TIME+ Command
875 root 20 0 557M 10168 6016 R 49.7 0.5 57:49.15 /usr/bin/fish
502 root 20 0 557M 8632 6048 R 49.0 0.4 57:48.21 /usr/bin/fish
1 root 20 0 98288 10044 7656 S 0.0 0.5 0:01.19 /sbin/init
188 root 20 0 81224 9036 8048 S 0.0 0.4 0:00.22 /lib/systemd/systemd-journald
207 root 20 0 21592 5100 3980 S 0.0 0.3 0:00.09 /lib/systemd/systemd-udevd
411 root 20 0 6748 2872 2608 S 0.0 0.1 0:00.03 /usr/sbin/cron -f
412 root 20 0 99888 9956 4532 S 0.0 0.5 0:00.01 /sbin/dhclient -4 -v -i -pf /run/dhclient-4
417 root 20 0 215M 6096 3368 S 0.0 0.3 0:00.02 /usr/sbin/rsyslogd -n -iNONE
419 root 20 0 22052 6952 6076 S 0.0 0.3 0:00.11 /lib/systemd/systemd-logind
421 root 20 0 14620 5080 4472 S 0.0 0.3 0:00.12 /sbin/wpa_supplicant -u -s -O /run/wpa_supplicant
423 root 20 0 99888 9956 4532 S 0.0 0.5 0:00.00 /sbin/dhclient -4 -v -i -pf /run/dhclient-4
424 root 20 0 99888 9956 4532 S 0.0 0.5 0:00.00 /sbin/dhclient -4 -v -i -pf /run/dhclient-4
425 root 20 0 99888 9956 4532 S 0.0 0.5 0:00.00 /sbin/dhclient -4 -v -i -pf /run/dhclient-4
428 root 20 0 215M 6096 3368 S 0.0 0.3 0:00.00 /usr/sbin/rsyslogd -n -iNONE
429 root 20 0 215M 6096 3368 S 0.0 0.3 0:00.00 /usr/sbin/rsyslogd -n -iNONE
431 root 20 0 215M 6096 3368 S 0.0 0.3 0:00.00 /usr/sbin/rsyslogd -n -iNONE
454 root 20 0 7324 4124 3444 S 0.0 0.2 0:00.05 /bin/login -p --
466 root 20 0 13356 7164 6268 S 0.0 0.4 0:00.00 sshd: /usr/sbin/sshd -D [listener] 4 of 128 max
491 root 20 0 10864 5052 4444 S 0.0 0.2 0:00.03 sudo -i
492 root 20 0 8196 4900 3464 S 0.0 0.2 0:00.02 -bash
501 root 20 0 10700 5044 4472 T 0.0 0.2 0:00.00 sudo -s /usr/bin/fish
840 root 20 0 10036 4892 4288 S 0.0 0.2 0:00.02 su - root
841 root 20 0 7928 4556 3236 S 0.0 0.2 0:00.00 -bash
846 root 20 0 10044 4416 3848 S 0.0 0.2 0:00.00 su - MariaPI201
854 root 20 0 10036 4872 4264 S 0.0 0.2 0:00.03 su - root
855 root 20 0 7928 4616 3292 S 0.0 0.2 0:00.01 -bash
859 root 20 0 10044 4364 3792 S 0.0 0.2 0:00.00 su - MariaPI201
864 root 20 0 10068 4444 3856 S 0.0 0.2 0:00.03 su - mitina_mv
874 root 20 0 10864 5200 4588 T 0.0 0.3 0:00.03 sudo -s /usr/bin/fish
3260 root 20 0 7304 4324 3664 S 0.0 0.2 0:00.03 /bin/login -p --
F1 Help F2 Setup F3 Search F4 Filter F5 Tree F6 Sort By F7 Nice F8 Nice + F9 Kill F10 Quit

```

Рисунок 25. процессы пользователя root

По данным данной утилиты мы видим: **mem** – использованная и доступная память, **cpu** – использование процессора (в процентах), **swp** – уровень занятости файла подкачки. Еще мы видим количество выполняемых задач (**tasks**), количество выполняемых процессов, среднюю нагрузку на систему (**Load average**) и (**Uptime**) время работы системы.

**PID** – идентификатор процесса. **USER** – владелец процесса.

**PRI** – текущий приоритет .

**NI** – величина изменения приоритета относительно значения **PRI**.

**VIRT** – общий объем виртуальной памяти, используемой процессом.

**RES** – количество резидентной (не перемещаемой в **swp**) памяти в КБ.

**SHR** – количество разделяемой (**shared**) памяти программы в КБ.

**S** – состояние процесса. **TIME+** – время работы процесса.

**CPU%** – использование процессора в процентном отношении.

**MEM%** – использование процессом памяти в процентном отношении.

**Command** – указывает на команду, которой был запущен процесс.



## Контрольные вопросы

1. Перечислите состояния задачи в ОС Ubuntu.

В ОС Ubuntu задачи могут находиться в 4 состояниях: Running – выполняемый, Sleeping – в ожидании, T – приостановлен, Zombie – зомби.

2. Как создаются задачи в ОС Ubuntu?

Задачи в Linux создаются следующим путем: процесс /bin/bash клонирует себя системным вызовом fork(), при этом создается клон /bin/bash с новым PID (идентификатор процесса) и PPID - равный PID родителя. Клон выполняет системный вызов exec с указанием на исполняемый файл и заменяет свой код - кодом исполняемого файла (родительский процесс при этом ждет завершения потомка - wait)

3. Назовите классы потоков ОС Ubuntu

- Потоки реального времени, обслуживаемые по алгоритму FIFO.
- Потоки реального времени, обслуживаемые в порядке циклической очереди.10
- Потоки разделения времени.

4. Как используется приоритет планирования при запуске задачи

При планировании задач приоритет используется при вычислении «добродетели» процесса. Цель алгоритма планирования состоит в том, чтобы обеспечить грубое пропорциональное соответствие качества обслуживания приоритету, то есть чем выше приоритет, тем меньше должно быть время отклика и тем большая доля процессорного времени достанется процессу

5. Как можно изменить приоритет для выполняющейся задачи?

С помощью команды **renice -n [значение nice] -p [PID процесса]**.

## Вывод

В ходе работы я научилась работать с процессами и их состояниями, узнала о демонах и других типах процессов.