

Липецкий государственный технический университет

Кафедра прикладной математики

Отчет по лабораторной работе №4 «Управление процессами ОС Ubuntu»

Студент

подпись, дата

Стукановский А.О.
фамилия, инициалы

Группа

ПМ-18

Руководитель
доц., к.п.н. кафедры АСУ
ученая степень, ученое звание

подпись, дата

Кургасов В. В.
фамилия, инициалы

Липецк 2020 г.

Содержание

Цель работы	3
Задание кафедры	3
Выполнение работы	4
Вывести общую информацию о системе	4
Вывести информацию о текущем интерпритаторе команд	4
Вывести информацию о текущем пользователе	4
Вывести информацию о текущем каталоге	4
Вывести информацию об оперативной памяти и области подкачки	4
Вывести информацию о дисковой памяти	4
Выполнить команды получения информации о процессах	5
Получить индентификатор текущего процесса(PID)	5
Получить индентификатор родительского процесса(PPID)	5
Получить индентификатор процесса инициализации системы	5
Получить информацию о выполняющихся процессах текущего пользователя в текущем интерпритаторе команд	5
Отобразить все процессы	6
Выполнить команды управления процессами	8
Получить информацию о выполняющихся процессах текущего пользователя в текущем интерпритаторе	8
Определить текущее значение nice	8
Запустить интерпритатор bash с пониженным приоритетом	8
Определить PID запущенного интерпритатора	8
Установить приоритет запущенного интерпритатора равным 5	8
Получить информацию о процессах bash	9
Контрольные вопросы	9
Вывод	11
Список литературы	12

Цель работы

Ознакомиться на практике с управлением процессами в ОС Ubuntu.

Задание кафедры

- 1) Запустить программу виртуализации Oracle VM VirtualBox.
- 2) Запустить виртуальную машину.
- 3) Открыть окно интерпритатора команд.
- 4) Вывести общую информацию о системе.
 - а) Вывести информацию о текущем интерпритаторе команд
 - б) Вывести информацию о текущем пользователе
 - в) Вывести информацию о текущем каталоге
 - г) Вывести информацию об оперативной памяти и области подкачки
 - д) Вывести информацию о дисковой памяти
- 5) Выполнить команды получения информации о процессах.
 - а) Получить идентификатор текущего процесса(PID)
 - б) Получить идентификатор родительского процесса(PPID)
 - в) Получить идентификатор процесса инициализации системы
 - г) Получить информацию о выполняющихся процессах текущего пользователя в текущем интерпритаторе команд
 - д) Отобразить все процессы
- 6) Выполнить команды управления процессами.
 - а) Получить информацию о выполняющихся процессах текущего пользователя в текущем интерпритаторе
 - б) Определить текущее значение nice
 - в) Запустить интерпритатор bash с пониженным приоритетом
`nice -n 10 bash`
 - г) Определить PID запущенного интерпритатора
 - д) Установить приоритет запущенного интерпритатора равным 5
`renice -n 5 <PID процесса>`
 - е) Получить информацию о процессах bash
`ps lax | grep bash`

Выполнение работы

Вывести общую информацию о системе

Вывести информацию о текущем интерпритаторе команд

Для получения информации о текущем интерпритаторе команд вызовем команду `echo $SHELL` (Рисунок 1).

```
artem@ubuntu-server:~$ echo $SHELL
/bin/bash
artem@ubuntu-server:~$
```

Рисунок 1.

Вывести информацию о текущем пользователе

Для получения информации о текущем пользователе вызовем команду `whoami` (Рисунок 2).

```
artem@ubuntu-server:~$ whoami
artem
artem@ubuntu-server:~$ _
```

Рисунок 2.

Вывести информацию о текущем каталоге

Для получения информации о текущей директории вызовем команду `pwd` (Рисунок 3).

```
artem@ubuntu-server:~$ pwd
/home/artem
artem@ubuntu-server:~$ _
```

Рисунок 3.

Вывести информацию об оперативной памяти и области подкачки

Для получения информации об оперативной памяти и области подкачки вызовем команду `free` (Рисунок 4).

```
artem@ubuntu-server:~$ free
              total        used         free       shared    buff/cache   available
Mem:           4030896       158676       3355556          1040         516664       3640524
Swap:              0              0              0
```

Рисунок 4.

Вывести информацию о дисковой памяти

Для получения информации о дисковой памяти вызовем команду `df` (Рисунок 5).

```

artem@ubuntuserver:~$ df
Filesystem            1K-blocks    Used Available Use% Mounted on
udev                  1971748        0   1971748    0% /dev
tmpfs                  403092     1040   402052    1% /run
/dev/mapper/ubuntu--vg-ubuntu--lv 7155192 3722424 3049588   55% /
tmpfs                 2015448        0   2015448    0% /dev/shm
tmpfs                  5120         0     5120    0% /run/lock
tmpfs                 2015448        0   2015448    0% /sys/fs/cgroup
/dev/loop0             100096    100096         0 100% /snap/core/10185
/dev/loop1             100096    100096         0 100% /snap/core/10126
/dev/loop2             56704     56704         0 100% /snap/core18/188
/dev/loop3             56704     56704         0 100% /snap/core18/193
/dev/sda2              999320   296884   633624   32% /boot
/dev/loop5              72320     72320         0 100% /snap/lxd/16922
/dev/loop7             31744     31744         0 100% /snap/snapd/9607
/dev/loop8              4352      4352         0 100% /snap/tree/18
/dev/loop6             31744     31744         0 100% /snap/snapd/9721
tmpfs                  403088        0   403088    0% /run/user/1000
/dev/loop9             69376     69376         0 100% /snap/lxd/18150
artem@ubuntuserver:~$ _

```

Рисунок 5.

Выполнить команды получения информации о процессах

Получить индентификатор текущего процесса(PID)

Для получения индентификатора текущего процесса вызовем команду echo \$\$ (Рисунок 6).

```

artem@ubuntuserver:~$ echo $$
943
artem@ubuntuserver:~$

```

Рисунок 6.

Получить индентификатор родительского процесса(PPID)

Для получения индентификатора родительского процесса вызовем команду echo \$PPID (Рисунок 7).

```

artem@ubuntuserver:~$ echo $PPID
657
artem@ubuntuserver:~$

```

Рисунок 7.

Получить индентификатор процесса инициализации системы

Для получения индентификатора процесса инициализации системы вызовем команду pidof init (Рисунок 8).

```

artem@ubuntuserver:~$ pidof init
1
artem@ubuntuserver:~$

```

Рисунок 8.

Получить информацию о выполняющихся процессах текущего пользователя в текущем интерпритаторе команд

Для получения информации о выполняющихся процессах текущего пользователя в текущем интерпритаторе команд вызовем команду ps. (Рисунок 9).

```

artem@ubuntu:~$ ps
  PID TTY          TIME CMD
   943 tty1        00:00:00 bash
  1391 tty1        00:00:00 ps
artem@ubuntu:~$ _

```

Рисунок 9.

Отобразить все процессы

Для отображения всех процессов воспользуемся командой `ps -e | less` (Рисунок 10). Полный список всех процессов приведён на рисунках с 11 по 13.

```

artem@ubuntu:~$ ps -e | less_

```

Рисунок 10.

```

  PID TTY          TIME CMD
    1 ?            00:00:03 systemd
    2 ?            00:00:00 kthreadd
    3 ?            00:00:00 rcu_gp
    4 ?            00:00:00 rcu_par_gp
    6 ?            00:00:00 kworker/0:0H-kblockd
    8 ?            00:00:00 kworker/u2:0-events_unbound
    9 ?            00:00:00 mm_percpu_wq
   10 ?            00:00:00 ksoftirqd/0
   11 ?            00:00:00 rcu_sched
   12 ?            00:00:00 migration/0
   13 ?            00:00:00 idle_inject/0
   14 ?            00:00:00 cpuhp/0
   15 ?            00:00:00 kdevtmpfs
   16 ?            00:00:00 netns
   17 ?            00:00:00 rcu_tasks_kthre
   18 ?            00:00:00 kauditd
   19 ?            00:00:00 khungtaskd
   20 ?            00:00:00 oom_reaper
   21 ?            00:00:00 writeback
   22 ?            00:00:00 kcompactd0
   23 ?            00:00:00 ksmd
   24 ?            00:00:00 khugepaged
   70 ?            00:00:00 kintegrityd
   71 ?            00:00:00 kblockd
   72 ?            00:00:00 blkcg_punt_bio
   73 ?            00:00:00 tpm_dev_wq
   74 ?            00:00:00 ata_sff
   75 ?            00:00:00 md
   76 ?            00:00:00 edac-poller
   77 ?            00:00:00 devfreq_wq
   78 ?            00:00:00 watchdogd
   81 ?            00:00:00 kworker/0:2-events
   83 ?            00:00:00 kswapd0
   84 ?            00:00:00 ecryptfs-kthrea
   86 ?            00:00:00 kthrotld

```

Рисунок 11.

```

87 ?      00:00:00 acpi_thermal_pm
88 ?      00:00:00 scsi_eh_0
89 ?      00:00:00 scsi_tmfs_0
90 ?      00:00:00 scsi_eh_1
91 ?      00:00:00 scsi_tmfs_1
93 ?      00:00:00 vfio-irqfd-clea
95 ?      00:00:00 ipv6_addrconf
104 ?     00:00:00 kstrp
107 ?     00:00:00 kworker/u3:0
120 ?     00:00:00 charger_manager
121 ?     00:00:00 kworker/0:1H-kblockd
164 ?     00:00:00 scsi_eh_2
165 ?     00:00:00 scsi_tmfs_2
175 ?     00:00:00 cryptd
196 ?     00:00:00 irq/18-vmwgfx
199 ?     00:00:00 ttm_swap
202 ?     00:00:00 kdmflush
226 ?     00:00:00 raid5wq
266 ?     00:00:00 jbd2/dm-0-8
267 ?     00:00:00 ext4-rsv-conver
335 ?     00:00:00 systemd-journal
365 ?     00:00:00 systemd-udevd
393 ?     00:00:00 iprt-VBoxHQueue
504 ?     00:00:00 kaluad
505 ?     00:00:00 kmpath_rdacd
506 ?     00:00:00 kmpathd
507 ?     00:00:00 kmpath_handlerd
508 ?     00:00:00 multipathd
518 ?     00:00:00 loop0
520 ?     00:00:00 loop1
521 ?     00:00:00 loop2
523 ?     00:00:00 loop3
527 ?     00:00:00 loop5
530 ?     00:00:00 loop6
531 ?     00:00:00 loop7
532 ?     00:00:00 jbd2/sda2-8

```

Рисунок 12.

```

532 ?     00:00:00 jbd2/sda2-8
533 ?     00:00:00 ext4-rsv-conver
535 ?     00:00:00 loop8
548 ?     00:00:00 systemd-timesyn
593 ?     00:00:00 systemd-network
595 ?     00:00:00 systemd-resolve
609 ?     00:00:00 accounts-daemon
612 ?     00:00:00 cron
613 ?     00:00:00 dbus-daemon
621 ?     00:00:00 networkd-dispat
629 ?     00:00:00 rsyslogd
631 ?     00:00:03 snapd
632 ?     00:00:00 systemd-logind
641 ?     00:00:00 atd
656 ?     00:00:00 kworker/0:5-events
657 tty1   00:00:00 login
682 ?     00:00:00 unattended-upgr
733 ?     00:00:00 polkitd
930 ?     00:00:00 systemd
932 ?     00:00:00 (sd-pam)
943 tty1   00:00:00 bash
1041 ?    00:00:00 loop9
1367 ?    00:00:00 kworker/u2:2-events_power_efficient
1389 ?    00:00:00 kworker/u2:1-events_power_efficient
1414 ?    00:00:00 kworker/0:0
1426 tty1  00:00:00 ps
1427 tty1  00:00:00 less
(END)

```

Рисунок 13.

Выполнить команды управления процессами

Получить информацию о выполняющихся процессах текущего пользователя в текущем интерпритаторе

Для получения информации о выполняющихся процессах текущего пользователя в текущем интерпритаторе вызовем команду `ps` (Рисунок 14).

```
artem@ubuntu:~$ ps -f
UID          PID    PPID  C STIME TTY          TIME CMD
artem         943      657  0 13:34 tty1        00:00:00 -bash
artem        1430      943  0 13:54 tty1        00:00:00 ps -f
artem@ubuntu:~$ _
```

Рисунок 14.

Определить текущее значение `nice`

Для определения текущего приоритета выполним команду `nice` (Рисунок 15).

```
artem@ubuntu:~$ nice
0
artem@ubuntu:~$ _
```

Рисунок 15.

Запустить интерпритатор `bash` с пониженным приоритетом

Для запуска `bash` с пониженным (на 10) приоритетом используем команду `nice -n 10 bash` (Рисунок 16).

```
artem@ubuntu:~$ nice -n 10 bash
artem@ubuntu:~$ ps -l
F S      UID          PID    PPID  C PRI  NI ADDR SZ WCHAN  TTY          TIME CMD
4 S      1000          943      657  0  80   0 - 1768 do_wai  tty1        00:00:00 bash
0 S      1000         1432      943  0  90  10 - 1760 do_wai  tty1        00:00:00 bash
0 R      1000         1439      1432  0  90  10 - 1888 -      tty1        00:00:00 ps
artem@ubuntu:~$ _
```

Рисунок 16.

Определить `PID` запущенного интерпритатора

Приоритет запущенного нами `bash` можно узнать, используя команду `ps -l`, посмотрев в столбик `NI`. (Рисунок 16).

Установить приоритет запущенного интерпритатора равным 5

Для установки приоритета уже работающему процессу используем команду `renice`. Однако как видно из рисунка 17, у нас недостаточно прав, чтобы повысить приоритет, поэтому используем команду через `sudo`.


```

artem@ubuntuserver:~$ renice -n 5 1432
renice: failed to set priority for 1432 (process ID): Permission denied
artem@ubuntuserver:~$ sudo renice -n 5 1432
[sudo] password for artem:
1432 (process ID) old priority 10, new priority 5
artem@ubuntuserver:~$ ps -l
F S   UID     PID   PPID  C PRI  NI ADDR SZ WCHAN  TTY          TIME CMD
4 S   1000     943    657   0  80   0 - 1768 do_wai tty1        00:00:00 bash
0 S   1000    1432    943   0  85   5 - 1760 do_wai tty1        00:00:00 bash
0 R   1000    1449    1432   0  85   5 - 1888 -      tty1        00:00:00 ps
artem@ubuntuserver:~$ _

```

Рисунок 17.

Получить информацию о процессах bash

Для получения информации о процессах bash выведем их всех через команду `ps lax | grep bash` (Рисунок 18).

```

artem@ubuntuserver:~$ ps lax | grep bash
4 1000  943  657 20 0 7072 4988 do_wai S      tty1        0:00 -bash
0 1000  1432  943 25 5 7040 4980 do_wai SN    tty1        0:00 bash
0 1000  1451  1432 25 5 5192 736 -      RN+  tty1        0:00 grep --color=auto bash
artem@ubuntuserver:~$

```

Рисунок 18.

Контрольные вопросы

1) Перечислите состояние задачи в ОС Ubuntu.

- **running** - Задача, которой выделили процессор.
- **sleeping** - Заблокированная задача
- **stopped** - остановленная задача
- **zombie** - задача остановлена, но ещё не удалена из системы
- **dead** - задача в состоянии смерти
- **active** и **expired** - состояния, которые используются в планировании выполнения процесса

2) Как создаются задачи в ОС Ubuntu

Задачи создаются путём вызова системной функции `clone`.

Любые обращения к `fork` и `vfork` преобразуются в системные вызовы `clone` во время компиляции. Функция `fork` создаёт дочернюю задачу, виртуальная память для которой выделяется по принципу копирования при записи. Когда дочерний или родительский процесс пытается выполнить запись в страницу памяти, записывающая программа создаёт собственную копию страницы в памяти.

3) Назовите классы потоков ОС Ubuntu.

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

4) Как используется приоритет планирования при запуске задачи?

У каждого потока есть приоритет планирования. Значение по умолчанию равно 20, но оно может быть изменено при помощи системного вызова `nice(value)`, вычитающего значение `value` из 20. Поскольку `value` должно находиться в диапазоне от -20 до +19, приоритеты всегда попадают в промежутки от 1 до 40.

5) Как можно изменить приоритет для выполняющейся задачи?

Использовать команду `renice` с заданным нужным приоритетом, однако, если необходимо повысить приоритет, то необходимы права суперпользователя.

Вывод

В ходе лабораторной работы я на практике ознакомился с управлением процессами в ОС Ubuntu.

Список литературы

- [1] Львовский, С.М. Набор и верстка в системе \LaTeX [Текст] / С.М. Львовский. М.: МЦНМО, 2006. — 448 с.
- [2] SEDICOMM. 30 полезных команд «ps» для мониторинга процессов Linux: <https://blog.sedicomm.com/2018/05/28/30-poleznyh-komand-ps-dlya-monitoringa-protsessov-linux/> (дата обращения: 29.10.2020). - Текст: электронный.