

Лабораторная работа №6

Дисциплина: Основы администрирования операционных систем

Жибицкая Евгения Дмитриевна

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Ответы на контрольные вопросы	17
4	Выводы	20
	Список литературы	21

Список иллюстраций

2.1	Запуск в фоновом и режиме и на переднем плане	6
2.2	jobs и перевод на передний план	7
2.3	Запуск процесса в другом терминале	7
2.4	top	8
2.5	Процессы dd	9
2.6	ps fax grep -B5 dd	10
2.7	Прекращение процессов	11
2.8	Задание 1	12
2.9	Задание 2. 1-9	13
2.10	Задание 2. 10	14
2.11	Задание 2. 11-13	15
2.12	Задание 2. 14-17	16
3.1	Пример	17
3.2	Пример	17
3.3	Пример	18
3.4	Пример	19

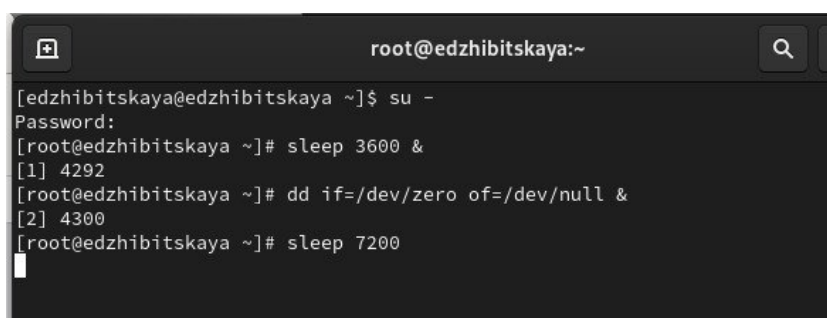
Список таблиц

1 Цель работы

Продолжение изучения ОС Linux. Изучение и получение навыков по управлению процессами ОС.

2 Выполнение лабораторной работы

Ознакомимся с теоретической частью и перейдем к управлению заданиями. Получим полномочия администратора и введем команды `sleep 3600 & dd if=/dev/zero of=/dev/null & sleep 7200`. Последняя команда была запущена не в фоновом режиме, поэтому для выхода понадобится сочетание клавиш `ctr Z` (рис. 2.1).



```
root@edzhibitskaya:~  
[edzhibitskaya@edzhibitskaya ~]$ su -  
Password:  
[root@edzhibitskaya ~]# sleep 3600 &  
[1] 4292  
[root@edzhibitskaya ~]# dd if=/dev/zero of=/dev/null &  
[2] 4300  
[root@edzhibitskaya ~]# sleep 7200
```

Рис. 2.1: Запуск в фоновом и режиме и на переднем плане

Командой `jobs` посмотрим на состояние заданий. Затем, используя `bg 3`, продолжим выполнение 3 задания. Переместим командой `fg 1` задание 1 на передний план, отменим его и проверим с помощью `jobs`. Выполним аналогичные действия для заданий 2 и 3 (рис. 2.2).

```
root@edzhibitskaya:~  
[root@edzhibitskaya ~]# dd if=/dev/zero of=/dev/null &  
[2] 4300  
[root@edzhibitskaya ~]# sleep 7200  
^Z  
[3]+  Stopped                  sleep 7200  
[root@edzhibitskaya ~]# jobs  
[1]  Running                  sleep 3600 &  
[2]-  Running                  dd if=/dev/zero of=/dev/null &  
[3]+  Stopped                  sleep 7200  
[root@edzhibitskaya ~]# bg 3  
[3]+  sleep 7200 &  
[root@edzhibitskaya ~]# jobs  
[1]  Running                  sleep 3600 &  
[2]-  Running                  dd if=/dev/zero of=/dev/null &  
[3]+  Running                  sleep 7200 &  
[root@edzhibitskaya ~]# fg 1  
sleep 3600  
^C  
[root@edzhibitskaya ~]# jobs  
[2]-  Running                  dd if=/dev/zero of=/dev/null &  
[3]+  Running                  sleep 7200 &  
[root@edzhibitskaya ~]# fg 2  
dd if=/dev/zero of=/dev/null  
^C142110467+0 records in  
142110467+0 records out  
72760559104 bytes (73 GB, 68 GiB) copied, 208.869 s, 348 MB/s  
  
[root@edzhibitskaya ~]# jobs  
[3]+  Running                  sleep 7200 &  
[root@edzhibitskaya ~]# fg 3  
sleep 7200  
^C  
[root@edzhibitskaya ~]# jobs  
[root@edzhibitskaya ~]#
```

Рис. 2.2: jobs и перевод на передний план

Под учетной записью пользователя введем команду `dd if=/dev/zero of=/dev/null &`, а затем выйдем и закроем терминал(рис. 2.3).

```
root@edzhibitskaya:~  x  edzhibitskaya@edzhibitskaya:~  x  
[edzhibitskaya@edzhibitskaya ~]$ dd if=/dev/zero of=/dev/null &  
[1] 4360  
[edzhibitskaya@edzhibitskaya ~]$ exit
```

Рис. 2.3: Запуск процесса в другом терминале

Используя команду `top`, увидим, что задание все равно запущено. Введем `k` и его номер, тем самым прекратив выполнение (рис. 2.4).

```

root@edzhbitskaya:~ x edzhbitskaya@edzhbitskaya:~ — ... x
top - 22:13:23 up 40 min, 2 users, load average: 1.24, 0.90, 0.56
Tasks: 202 total, 3 running, 199 sleeping, 0 stopped, 0 zombie
%Cpu0 : 27.3 us, 63.6 sy, 0.0 ni, 0.0 id, 0.0 wa, 9.1 hi, 0.0 si, 0.0 st
%Cpu1 : 3.1 us, 3.1 sy, 0.0 ni, 92.3 id, 0.0 wa, 1.5 hi, 0.0 si, 0.0 st
MiB Mem : 1967.5 total, 689.6 free, 1084.1 used, 375.6 buff/cache
MiB Swap: 2096.0 total, 1652.9 free, 443.1 used, 883.4 avail Mem
PID to signal/kill [default pid = 49] 4360

```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
49	root	0	-20	0	0	0	I	1.5	0.0	0:00.82	kworker+
4188	edzhibi+	20	0	762492	49264	36644	S	1.5	2.4	0:02.78	gnome-t+
4402	edzhibi+	20	0	225896	3968	3200	R	1.5	0.2	0:00.32	top
1	root	20	0	173996	9828	6496	S	0.0	0.5	0:02.07	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kthreadd
3	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_gp
4	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_par+
5	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	slub_fl+
6	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	netns
8	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker+
10	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	mm_perc+
12	root	20	0	0	0	0	I	0.0	0.0	0:00.00	rcu_tas+
13	root	20	0	0	0	0	I	0.0	0.0	0:00.00	rcu_tas+
14	root	20	0	0	0	0	I	0.0	0.0	0:00.00	rcu_tas+
15	root	20	0	0	0	0	S	0.0	0.0	0:00.17	ksoftir+
16	root	20	0	0	0	0	S	0.0	0.0	0:00.01	pr/tty0
17	root	20	0	0	0	0	R	0.0	0.0	0:00.73	rcu_pre+
18	root	rt	0	0	0	0	S	0.0	0.0	0:02.87	migrati+
19	root	-51	0	0	0	0	S	0.0	0.0	0:00.00	idle_in+
21	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/0
22	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/1
23	root	-51	0	0	0	0	S	0.0	0.0	0:00.00	idle_in+
24	root	rt	0	0	0	0	S	0.0	0.0	0:02.90	migrati+
25	root	20	0	0	0	0	S	0.0	0.0	0:00.12	ksoftir+
27	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker+
29	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kdevtmp+

Рис. 2.4: top

Перейдем к управлению процессами. Трижды введем команду `dd if=/dev/zero of=/dev/null &`. После найдем ксе строки, содержащие `dd` - `ps aux | grep dd`(рис. 2.5).


```

[root@edzhbitskaya ~]# dd if=/dev/zero of=/dev/null &
[2] 4497
[root@edzhbitskaya ~]# dd if=/dev/zero of=/dev/null &
[3] 4500
[root@edzhbitskaya ~]# ps aux | grep dd
root      2  0.0  0.0      0  0 ?        S   21:33   0:00 [kthreadd]
root     67  0.0  0.0      0  0 ?        I<  21:33   0:00 [ipv6_addrcon
f]
dbus      723  0.0  0.1  10108 3884 ?        S   21:33   0:00 dbus-broker -
-log 4 --controller 9 --machine-id 4be477a317dd4d2f9d4e164ed15963e2 --max-bytes
536870912 --max-fds 4096 --max-matches 131072 --audit
edzhibi+  2736  0.0  0.1  6720 2560 ?        S   21:56   0:00 dbus-broker -
-log 4 --controller 9 --machine-id 4be477a317dd4d2f9d4e164ed15963e2 --max-bytes
1000000000000000 --max-fds 2500000000000000 --max-matches 5000000000
edzhibi+  2862  0.0  0.0  4988 1792 ?        S   21:56   0:00 dbus-broker -
-log 4 --controller 9 --machine-id 4be477a317dd4d2f9d4e164ed15963e2 --max-bytes
1000000000000000 --max-fds 6400000 --max-matches 5000000000
edzhibi+  2974  0.0  0.5 880196 10344 ?        Ssl 21:56   0:00 /usr/libexec/
evolution-addressbook-factory
edzhibi+  3773  0.0  1.3 249372 27008 ?        Sl   21:57   0:00 /usr/lib64/fi
refox/firefox -contentproc -parentBuildID 20240418173117 -prefsLen 32334 -prefMa
pSize 236567 -appDir /usr/lib64/firefox/browser {865b6532-0f4e-49e5-9406-b967473
2bdfc} 3440 rdd
edzhibi+  4360 95.5  0.0 220988 1664 ?        R   22:10   5:21 dd if=/dev/ze
ro of=/dev/null
root     4489 53.1  0.0 220988 1664 pts/3    R   22:15   0:37 dd if=/dev/ze
ro of=/dev/null
root     4497 37.3  0.0 220988 1664 pts/3    R   22:15   0:18 dd if=/dev/ze
ro of=/dev/null
root     4500 31.7  0.0 220988 1664 pts/3    R   22:15   0:08 dd if=/dev/ze
ro of=/dev/null
root     4502  0.0  0.1 221796 2176 pts/3    S+  22:16   0:00 grep --color=
auto dd
[root@edzhbitskaya ~]#

```

Рис. 2.5: Процессы dd

Изменим приоритет одного из процессов на 5, затем введем команду, показывающую иерархию отношений между процессами (также параметр -B5 показывает соответствующие запросу строки, включая пять строк до этого) - `ps fax | grep -B5 dd` (рис. 2.6).

```
root@edzhbitskay... x edzhbitskaya@ed... x root@edzhbitskay... x
[root@edzhbitskaya ~]# renice -n 5 4500
4500 (process ID) old priority 0, new priority 5
> [root@edzhbitskaya ~]# ps fax | grep -B5 dd
  PID TTY          STAT TIME  COMMAND
  --  --  --
    2 ?            S    0:00 [kthreadd]
  --  --  --
   61 ?            I<    0:00 \_ [acpi_thermal_pm]
   62 ?            I<    0:00 \_ [kmpath_rdacd]
   63 ?            I<    0:00 \_ [kaluad]
   64 ?            I    0:02 \_ [kworker/u4:3-events_unbound]
   66 ?            I<    0:00 \_ [mld]
   67 ?            I<    0:00 \_ [ipv6_addrconf]
  --  --  --
  600 ?           Ss    0:00 /usr/lib/systemd/systemd-journald
  615 ?           Ss    0:00 /usr/lib/systemd/systemd-udev
  699 ?           S<sl  0:00 /sbin/auditd
  701 ?           S<    0:00 \_ /usr/sbin/sedispatch
  722 ?           Ss    0:00 /usr/bin/dbus-broker-launch --scope system --aud
  723 ?           S    0:00 \_ dbus-broker --log 4 --controller 9 --machine-
4be477a317dd4d2f9d4e164ed15963e2 --max-bytes 536870912 --max-fds 4096 --max-p
ches 131072 --audit
```

Рис. 2.6: ps fax | grep -B5 dd

Найдем номер корневой оболочки и остановим все запущенные процессы(рис. 2.7).

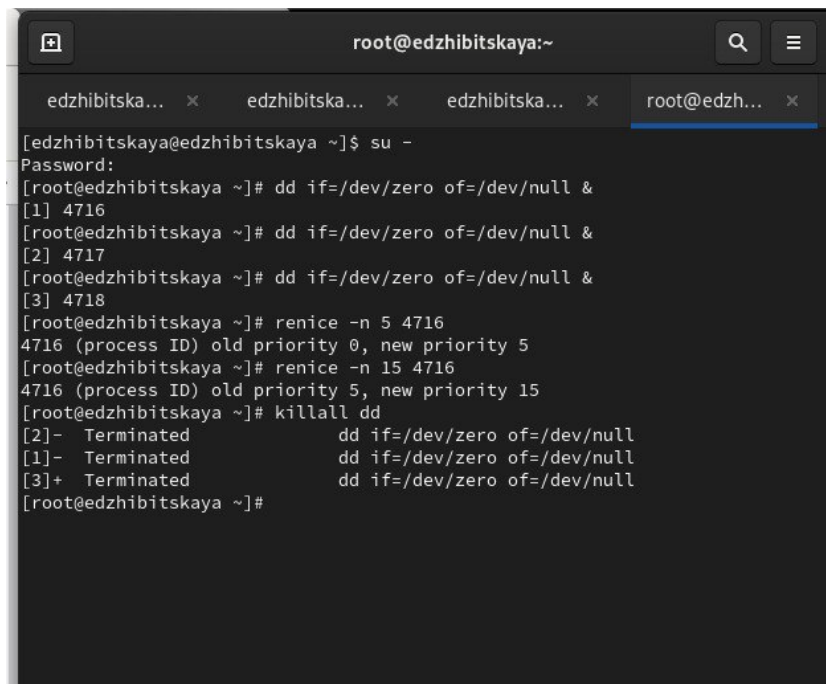
```

edzhibitskaya@e... x edzhibitskaya@ed... x edzhibitskaya@ed... x
3773 ?      Sl      0:00 |      \_ /usr/lib64/firefox/firefox -contentproc
-parentBuildID 20240418173117 -prefsLen 32334 -prefMapSize 236567 -appDir /usr/
lib64/firefox/browser {865b6532-0f4e-49e5-9406-b9674732bdfc} 3440 rdd
3793 ?      Sl      0:35 |      \_ /usr/lib64/firefox/firefox -contentproc
-childID 7 -isForBrowser -prefsLen 29040 -prefMapSize 236567 -jsInitLen 240916
-parentBuildID 20240418173117 -appDir /usr/lib64/firefox/browser {20d066e1-42e2-
4f19-8330-2184afaea446} 3440 tab
2837 ?      Ssl     0:00 \_ /usr/libexec/gvfsd
2845 ?      Sl      0:00 \_ /usr/libexec/gvfsd-fuse /run/user/1000/gvfs -f
2855 ?      Ssl     0:00 \_ /usr/libexec/at-spi-bus-launcher
2861 ?      S       0:00 |      \_ /usr/bin/dbus-broker-launch --config-file=/
usr/share/defaults/at-spi2/accessibility.conf --scope user
2862 ?      S       0:00 |      \_ dbus-broker --log 4 --controller 9 --ma
chine-id 4be477a317dd4d2f9d4e164ed15963e2 --max-bytes 1000000000000000 --max-fds
64000000 --max-matches 50000000000
--
2930 ?      Ssl     0:00 \_ /usr/libexec/evolution-calendar-factory
2933 ?      Ssl     0:00 \_ /usr/libexec/goa-identity-service
2951 ?      Ssl     0:00 \_ /usr/libexec/gvfs-udisks2-volume-monitor
2962 ?      Ssl     0:00 \_ /usr/libexec/gvfs-goa-volume-monitor
2968 ?      Ssl     0:00 \_ /usr/libexec/dconf-service
2974 ?      Ssl     0:00 \_ /usr/libexec/evolution-adddressbook-factory
--
4264 pts/0    S+      0:00 |      |      \_ -bash
4367 pts/2    Ss+     0:00 |      \_ _bash
4424 pts/3    Ss      0:00 |      \_ _bash
4450 pts/3    S       0:00 |      \_ su -
4461 pts/3    S       0:00 |      \_ -bash
4489 pts/3    R       1:35 |      \_ dd if=/dev/zero of=/dev/null
4497 pts/3    R       1:15 |      \_ dd if=/dev/zero of=/dev/null
4500 pts/3    RN      0:55 |      \_ dd if=/dev/zero of=/dev/null
4520 pts/3    R+      0:00 |      \_ ps fax
4521 pts/3    S+      0:00 |      \_ grep --color=auto -B5 dd
4360 ?      R       7:55 \_ dd if=/dev/zero of=/dev/null
[root@edzhibitskaya ~]# kill -9 4264
[root@edzhibitskaya ~]# kill -9 4360
[root@edzhibitskaya ~]# kill -9 4461
Killed
[edzhibitskaya@edzhibitskaya ~]$

```

Рис. 2.7: Прекращение процессов

Приступим к выполнению самостоятельной работы. Выполним задание 1. Для этого запустим команду `dd if=/dev/zero of=/dev/null` в фоновом режиме трижды. Командой `renice` изменим приоритет процесса сначала на 5, затем на 15. Завершив все процессы командой `killall dd`(рис. 2.8).



```
root@edzhibitskaya:~  
edzhibitska... x edzhibitska... x edzhibitska... x root@edzh... x  
[edzhibitskaya@edzhibitskaya ~]$ su -  
Password:  
[root@edzhibitskaya ~]# dd if=/dev/zero of=/dev/null &  
[1] 4716  
[root@edzhibitskaya ~]# dd if=/dev/zero of=/dev/null &  
[2] 4717  
[root@edzhibitskaya ~]# dd if=/dev/zero of=/dev/null &  
[3] 4718  
[root@edzhibitskaya ~]# renice -n 5 4716  
4716 (process ID) old priority 0, new priority 5  
[root@edzhibitskaya ~]# renice -n 15 4716  
4716 (process ID) old priority 5, new priority 15  
[root@edzhibitskaya ~]# killall dd  
[2]- Terminated dd if=/dev/zero of=/dev/null  
[1]- Terminated dd if=/dev/zero of=/dev/null  
[3]+ Terminated dd if=/dev/zero of=/dev/null  
[root@edzhibitskaya ~]#
```

Рис. 2.8: Задание 1

Выполним задание 2. Запустим программу `yues` в фоновом режиме с подавлением потока вывода, а также на переднем плане с подавлением потока вывода (отличие в `&`). Приостановим их выполнение. Командой `jobs` проверим состояние. Переведем процесс, который выполняется в фоновом режиме, на передний план (`fg 1`) и остановим его. Переведем еще один процесс в фоновый режим (`bg 3`), проверим состояние процессов. Далее запустим процесс в фоновом режиме таким образом, чтобы он продолжил свою работу даже после отключения от терминала (с помощью `nohup`) (рис. 2.9). Закроем окно и заново запустим консоль. Убедимся, что процесс продолжил свою работу. Получим информацию о запущенных в операционной системе процессах с помощью утилиты `top` (рис. 2.10).

```
root@edzhbitskaya:~  
edzhbitskaya@ed... x edzhbitskaya@ed... x root@edzhbitskay... x  
[edzhbitskaya@edzhbitskaya ~]$ su -  
Password:  
[root@edzhbitskaya ~]# yes > /dev/null &  
[1] 6305  
[root@edzhbitskaya ~]# yes > /dev/null  
^Z  
[2]+  Stopped                  yes > /dev/null  
[root@edzhbitskaya ~]# jobs  
[1]-  Running                  yes > /dev/null &  
[2]+  Stopped                  yes > /dev/null  
[root@edzhbitskaya ~]# fg 1  
yes > /dev/null  
^C  
[root@edzhbitskaya ~]# yes > /dev/null  
^Z  
[3]+  Stopped                  yes > /dev/null  
[root@edzhbitskaya ~]# bg 3  
[3]+ yes > /dev/null &  
[root@edzhbitskaya ~]# jobs  
[2]+  Stopped                  yes > /dev/null  
[3]-  Running                  yes > /dev/null &  
[root@edzhbitskaya ~]# nohup yes > /dev/null &  
[4] 6576  
[root@edzhbitskaya ~]# nohup: ignoring input and redirecting stderr to stdout
```

Рис. 2.9: Задание 2. 1-9

```

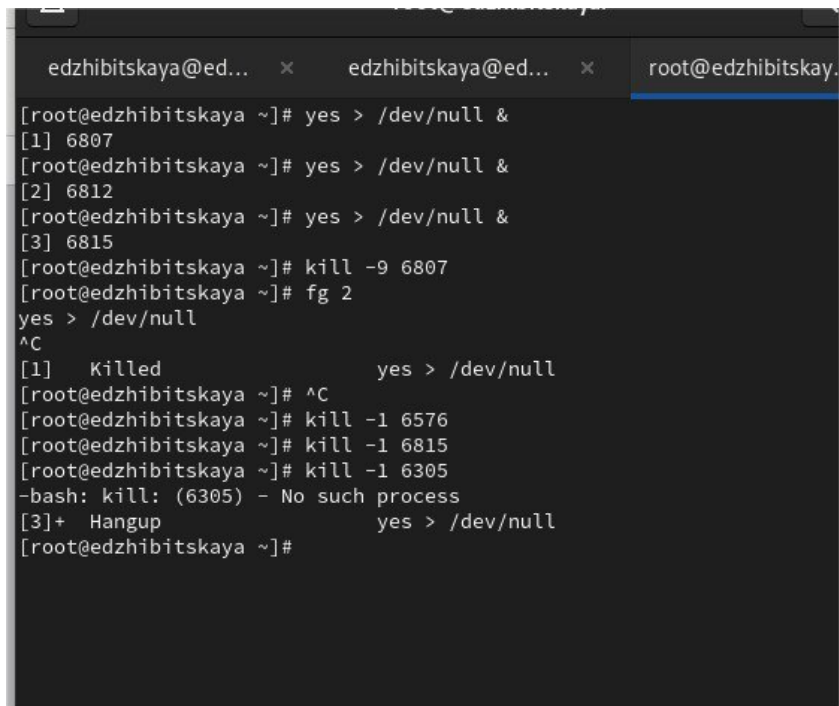
root@edzhbitskaya:~
edzhbitskaya@ed... x edzhbitskaya@ed... x root@edzhbitskay... x
top - 22:59:57 up 1:26, 2 users, load average: 3.31, 2.41, 2.12
Tasks: 213 total, 4 running, 209 sleeping, 0 stopped, 0 zombie
%Cpu(s): 24.4 us, 72.9 sy, 0.0 ni, 0.0 id, 0.0 wa, 2.7 hi, 0.0 si, 0.0 st
MiB Mem : 1967.5 total, 232.3 free, 1481.6 used, 451.7 buff/cache
MiB Swap: 2096.0 total, 1651.1 free, 444.9 used, 485.9 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR  S  %CPU  %MEM    TIME+  COMMAND
 6178 root        20   0   220948    1664    1664  R   88.1    0.1   6:45.78  yes
 6341 root        20   0   220948    1664    1664  R   47.7    0.1   3:58.52  yes
 6576 root        20   0   220948    1664    1664  R   47.4    0.1   0:36.82  yes
 3677 edzhibi+    20   0   10.9g   266380  92052  S    7.3   13.2   1:13.22  Isolate+
 4826 root        20   0         0         0         0  I    2.3    0.0   0:01.84  kworker+
 6724 root        20   0         0         0         0  I    1.3    0.0   0:00.13  kworker+
 2806 edzhibi+    20   0   4123968 246156  70664  S    1.0   12.2   4:43.78  gnome-s+
 4188 edzhibi+    20   0   764000  51580  37040  S    0.7    2.6   0:12.35  gnome-t+
 3440 edzhibi+    20   0   3549312 407404  132816 S    0.3   20.2   3:29.09  firefox
 6731 root        20   0   226012    3968    3200  R    0.3    0.2   0:00.03  top
    1 root        20   0   173996    9828    6496  S    0.0    0.5   0:02.24  systemd
    2 root        20   0         0         0         0  S    0.0    0.0   0:00.00  kthreadd
    3 root         0 -20         0         0         0  I    0.0    0.0   0:00.00  rcu_gp
    4 root         0 -20         0         0         0  I    0.0    0.0   0:00.00  rcu_par+
    5 root         0 -20         0         0         0  I    0.0    0.0   0:00.00  slub_fl+
    6 root         0 -20         0         0         0  I    0.0    0.0   0:00.00  netns
    8 root         0 -20         0         0         0  I    0.0    0.0   0:00.00  kworker+
   10 root         0 -20         0         0         0  I    0.0    0.0   0:00.00  mm_perc+
   12 root        20   0         0         0         0  I    0.0    0.0   0:00.00  rcu_tas+
   13 root        20   0         0         0         0  I    0.0    0.0   0:00.00  rcu_tas+
   14 root        20   0         0         0         0  I    0.0    0.0   0:00.00  rcu_tas+
   15 root        20   0         0         0         0  S    0.0    0.0   0:00.42  ksoftir+
   16 root        20   0         0         0         0  S    0.0    0.0   0:00.02  pr/tty0
   17 root        20   0         0         0         0  I    0.0    0.0   0:01.18  rcu_pre+
   18 root        rt    0         0         0         0  S    0.0    0.0   0:02.88  migrati+
   19 root       -51   0         0         0         0  S    0.0    0.0   0:00.00  idle_in+
   21 root        20   0         0         0         0  S    0.0    0.0   0:00.00  cpuhp/0
   22 root        20   0         0         0         0  S    0.0    0.0   0:00.00  cpuhp/1
   23 root       -51   0         0         0         0  S    0.0    0.0   0:00.00  idle_in+
   24 root        rt    0         0         0         0  S    0.0    0.0   0:02.92  migrati+
   25 root        20   0         0         0         0  S    0.0    0.0   0:00.15  ksoftir+
   27 root         0 -20         0         0         0  I    0.0    0.0   0:00.00  kworker+

```

Рис. 2.10: Задание 2. 10

Запустим еще 3 программы в фоновом режиме с подавлением потока вывода. Завершим первый процесс командой kill -9(используя PID), а второй - используя его идентификатор. Попробуем также послать сигнал 1 (SIGHUP) процессу, запущенному с помощью nohup, и оставшемуся процессу.(рис. 2.11).

A terminal window with three tabs. The active tab is 'root@edzhibitskaya'. The terminal shows a series of commands and their outputs. The user runs 'yes > /dev/null &' three times, which outputs '[1] 6807', '[2] 6812', and '[3] 6815'. Then, the user runs 'kill -9 6807' and 'fg 2'. The output shows the first process is killed and the second process is brought to the foreground. The user then runs 'kill -1 6576', 'kill -1 6815', and 'kill -1 6305'. The output shows that the process 6305 does not exist. Finally, the user runs 'yes > /dev/null' and the terminal shows a hangup signal.

```
edzhibitskaya@ed... x edzhibitskaya@ed... x root@edzhibitskay.
[root@edzhibitskaya ~]# yes > /dev/null &
[1] 6807
[root@edzhibitskaya ~]# yes > /dev/null &
[2] 6812
[root@edzhibitskaya ~]# yes > /dev/null &
[3] 6815
[root@edzhibitskaya ~]# kill -9 6807
[root@edzhibitskaya ~]# fg 2
yes > /dev/null
^C
[1] Killed yes > /dev/null
[root@edzhibitskaya ~]# ^C
[root@edzhibitskaya ~]# kill -1 6576
[root@edzhibitskaya ~]# kill -1 6815
[root@edzhibitskaya ~]# kill -1 6305
-bash: kill: (6305) - No such process
[3]+ Hangup yes > /dev/null
[root@edzhibitskaya ~]#
```

Рис. 2.11: Задание 2. 11-13

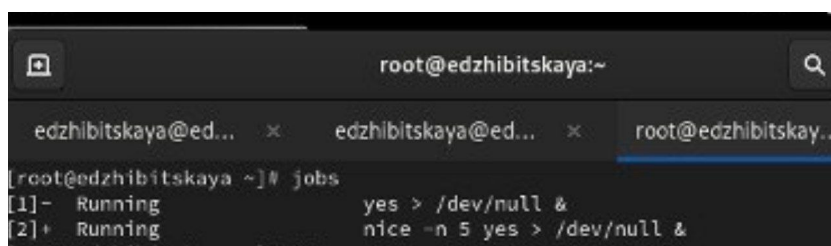
Запустим 2 аналогичные программы. Командой `killall` завершим их работу одновременно. Создадим еще 1 программу, затем командой `nice` запустим программу `yes` с приоритетом большим на 5. Сравним приоритеты(воспользовавшись фильтром), затем утилитой `renice` сравняем их значения и завершим нашу работу(рис. 2.12).

```
edzhibitskaya@ed... x edzhibitskaya@ed... x root@edzhibitskay... x
> [root@edzhibitskaya ~]# yes > /dev/null &
[1] 7350
> [root@edzhibitskaya ~]# yes > /dev/null &
[2] 7358
[root@edzhibitskaya ~]# killall yes
[1]- Terminated yes > /dev/null
[2]+ Terminated yes > /dev/null
[root@edzhibitskaya ~]# yes > /dev/null &
[1] 7398
[root@edzhibitskaya ~]# nice -n 5 yes > /dev/null &
[2] 7700
[root@edzhibitskaya ~]# ps -l | grep yes
0 R 0 7398 6696 96 80 0 - 55237 - pts/4 00:05:20 yes
0 R 0 7700 6696 83 85 5 - 55237 - pts/4 00:00:40 yes
[root@edzhibitskaya ~]# renice -n 5 7398
7398 (process ID) old priority 0, new priority 5
[root@edzhibitskaya ~]# ps -l | grep yes
0 R 0 7398 6696 96 85 5 - 55237 - pts/4 00:06:13 yes
0 R 0 7700 6696 87 85 5 - 55237 - pts/4 00:01:30 yes
[root@edzhibitskaya ~]#
```

Рис. 2.12: Задание 2. 14-17

3 Ответы на контрольные вопросы


1. Чтобы получить обзор всех текущих заданий оболочки, используем команду `jobs`(рис. 3.1).



```
root@edzhbitskaya:~  
edzhbitskaya@ed... x edzhbitskaya@ed... x root@edzhbitskay...  
[root@edzhbitskaya ~]# jobs  
[1]-  Running                  yes > /dev/null &  
[2]+  Running                  nice -n 5 yes > /dev/null &
```

Рис. 3.1: Пример

2. Чтобы остановить текущее задание оболочки и продолжить его выполнение в фоновом режиме, используем: `Ctrl + Z bg (number)`
3. Комбинация клавиш для отмены текущего задания оболочки: `Ctrl + C`(рис. 3.2).



```
[root@edzhbitskaya ~]# yes > /dev/null  
^C  
[root@edzhbitskaya ~]#
```

Рис. 3.2: Пример

4. Чтобы отменить одно из начатых заданий, если у вас нет доступа к оболочке, можно использовать команду `kill` вместе с идентификатором (PID) процесса. Например: `kill (process_id)`(рис. 3.3).

```
edzhibitskaya@ed... x edzhibitskaya@ed... x root@edzhibitskay... x
> [root@edzhibitskaya ~]# yes > /dev/null &
[1] 7350
>> [root@edzhibitskaya ~]# yes > /dev/null &
[2] 7358
[root@edzhibitskaya ~]# killall yes
[1]- Terminated yes > /dev/null
[2]+ Terminated yes > /dev/null
[root@edzhibitskaya ~]# yes > /dev/null &
[1] 7398
[root@edzhibitskaya ~]# nice -n 5 yes > /dev/null &
[2] 7700
[root@edzhibitskaya ~]# ps -l | grep yes
0 R 0 7398 6696 96 80 0 - 55237 - pts/4 00:05:20 yes
0 R 0 7700 6696 83 85 5 - 55237 - pts/4 00:00:40 yes
[root@edzhibitskaya ~]# renice -n 5 7398
7398 (process ID) old priority 0, new priority 5
[root@edzhibitskaya ~]# ps -l | grep yes
0 R 0 7398 6696 96 85 5 - 55237 - pts/4 00:06:13 yes
0 R 0 7700 6696 87 85 5 - 55237 - pts/4 00:01:30 yes
[root@edzhibitskaya ~]#
```

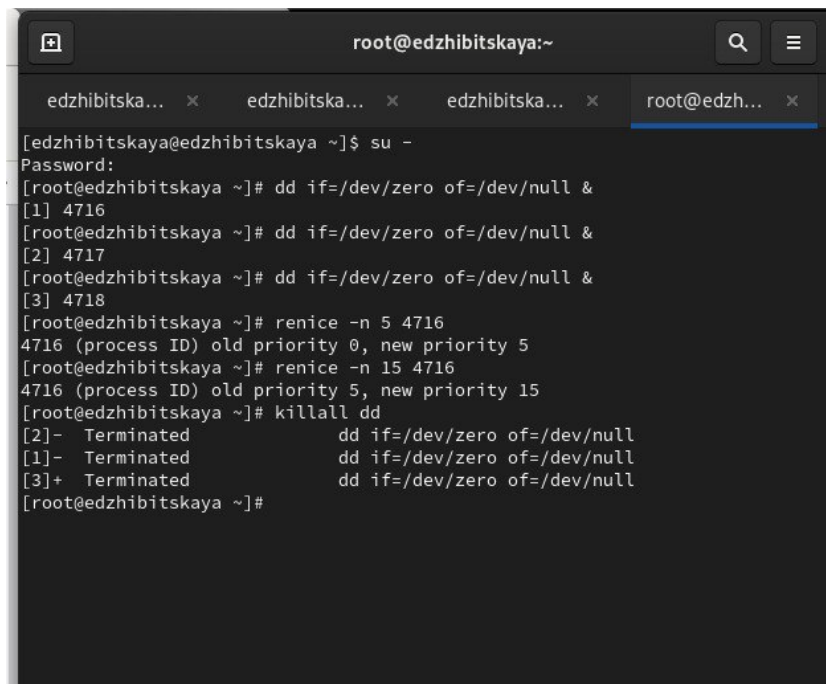
Рис. 3.3: Пример

5. Команда, используемая для отображения отношений между родительскими и дочерними процессами: `ps fax`

6. Чтобы изменить приоритет процесса с идентификатором 1234 на более высокий, используем команду: `renice -n -5 -p 1234`

7. Чтобы остановить все запущенные процессы `dd`, можно использовать: `pkill dd`

`killall dd` (рис. 3.4).



```
root@edzhibitskaya:~  
edzhibitska... x edzhibitska... x edzhibitska... x root@edzh... x  
[edzhibitskaya@edzhibitskaya ~]$ su -  
Password:  
[root@edzhibitskaya ~]# dd if=/dev/zero of=/dev/null &  
[1] 4716  
[root@edzhibitskaya ~]# dd if=/dev/zero of=/dev/null &  
[2] 4717  
[root@edzhibitskaya ~]# dd if=/dev/zero of=/dev/null &  
[3] 4718  
[root@edzhibitskaya ~]# renice -n 5 4716  
4716 (process ID) old priority 0, new priority 5  
[root@edzhibitskaya ~]# renice -n 15 4716  
4716 (process ID) old priority 5, new priority 15  
[root@edzhibitskaya ~]# killall dd  
[2]- Terminated dd if=/dev/zero of=/dev/null  
[1]- Terminated dd if=/dev/zero of=/dev/null  
[3]+ Terminated dd if=/dev/zero of=/dev/null  
[root@edzhibitskaya ~]#
```

Рис. 3.4: Пример

8. Чтобы остановить команду с именем `mysommand`, используйте: `pkill mysommand`
9. В командной утилите `top`, чтобы убить процесс, используем клавишу `k`, затем PID процесса, который нужно завершить, и сигнал.
10. Чтобы запустить команду с достаточно высоким приоритетом, не рискуя исчерпать ресурсы для других процессов, используем: `nice -n 10 command`

4 Выводы

В ходе работы было произведено знакомство с принципами управления процессами, получены навыки и проделаны различные действия по запуску и остановке процессов, заданий.

Список литературы

ТУИС