

«Контроль использования ресурсов ОС Linux»

Группа 2-МВ-4

Ярошевская Д.А.

1. Вывести список всех процессов системы.

Для вывода списка процессов используется команда **ps** с ключами:

a – процессы всех пользователей, **u** – отображение информации о владельце,

X – процессы, которые не связаны с терминалом.

```

dari@Ubuntu:~$ ps aux
USER          PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root            1  0.9  0.5 168144 11596 ?        Ss   18:44   0:00 /sbin/init sp
root            2  0.0  0.0      0      0 ?        S    18:44   0:00 [kthreadd]
root            3  0.0  0.0      0      0 ?        I<   18:44   0:00 [rcu_gp]
root            4  0.0  0.0      0      0 ?        I<   18:44   0:00 [rcu_par_gp]
root            5  0.0  0.0      0      0 ?        I<   18:44   0:00 [slub_flushwq
root            6  0.0  0.0      0      0 ?        I<   18:44   0:00 [netns]
root            7  0.0  0.0      0      0 ?        I    18:44   0:00 [kworker/0:0-
root            8  0.0  0.0      0      0 ?        I<   18:44   0:00 [kworker/0:0H
root            9  0.0  0.0      0      0 ?        I    18:44   0:00 [kworker/u2:0
root           10  0.0  0.0      0      0 ?        I<   18:44   0:00 [mm_percpu_wq
root           11  0.0  0.0      0      0 ?        S    18:44   0:00 [rcu_tasks_ru
root           12  0.0  0.0      0      0 ?        S    18:44   0:00 [rcu_tasks_tr
root           13  0.1  0.0      0      0 ?        S    18:44   0:00 [ksoftirqd/0]
root           14  0.2  0.0      0      0 ?        I    18:44   0:00 [rcu_sched]
root           15  0.0  0.0      0      0 ?        S    18:44   0:00 [migration/0]

```

2. Вывести дерево процессов.

Для просмотра дерева процессов используем команду **ps tree**.

```

dari@Ubuntu:~$ pstree
systemd├─ModemManager─2*[{ModemManager}]
      ├─NetworkManager─2*[{NetworkManager}]
      ├─accounts-daemon─2*[{accounts-daemon}]
      ├─acpid
      ├─avahi-daemon─avahi-daemon
      ├─colord─2*[{colord}]
      ├─cron
      ├─cups-browsed─2*[{cups-browsed}]
      ├─cupsd
      ├─dbus-daemon
      └─gdm3├─gdm-session-wor├─gdm-x-session├─Xorg─{Xorg}
          │               │               │   └─gnome-session-b├─ssh-agent
          │               │               │                   │ 2*[{gnome-ses+
          │               │               └─2*[{gdm-x-session}]
          │               └─2*[{gdm-session-wor}]
          └─2*[{gdm3}]
      └─gnome-keyring-d─3*[{gnome-keyring-d}]
      └─2*[kerneloops]
      └─networkd-dispat
      └─polkitd─2*[{polkitd}]
      └─sshd─2*[{sshd}]

```

3. С помощью команды **top** получить список 5 процессов, потребляющих наибольшее количество процессорного времени.

Команда **top** представляет динамически обновляемые сведения о процессах.

```
dari@Ubuntu:~$ top

top - 19:05:53 up 21 min,  1 user,  load average: 0,02, 0,05, 0,03
Tasks: 168 total,   1 running, 167 sleeping,   0 stopped,   0 zombie
%Cpu(s):  3,1 us,  0,7 sy,  0,0 ni, 95,9 id,  0,3 wa,  0,0 hi,  0,0 si,  0,0 st
MiB Mem :  1964,1 total,   582,3 free,   732,5 used,   649,3 buff/cache
MiB Swap:  1401,6 total,  1401,6 free,    0,0 used.  1073,5 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM    TIME+  COMMAND
 1407 dari      20   0 3448384 355812 127056 S   2,3   17,7   0:15.90 gnome-shell
 1247 dari      20   0 259860  74700  47364 S   1,3    3,7   0:05.64 Xorg
 1377 dari      20   0    7248   4284   3820 S   0,7    0,2   0:00.02 dbus-daemon
 1730 dari      20   0 814992  51224  38724 S   0,7    2,5   0:02.15 gnome-term+
   25 root       20   0      0      0      0 S   0,3    0,0   0:00.12 kcompactd0
 1926 dari      20   0   12172   3656   3152 R   0,3    0,2   0:00.96 top
    1 root       20   0 168144  11596   8488 S   0,0    0,6   0:08.31 systemd
    2 root       20   0      0      0      0 S   0,0    0,0   0:00.00 kthreadd
```

Нажимаем **n** и вводим количество процессов – 5.

```
Maximum tasks = 0, change to (0 is unlimited) 5

  PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM    TIME+  COMMAND
 1407 dari      20   0 3440188 355780 127092 S   0,8   17,7   0:16.55 gnome-shell
    1 root       20   0 168144  11596   8488 S   0,0    0,6   0:00.91 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_gp
```

Для перехода в меню нажимаем **f**, выбираем **TIME+**, затем нажимаем **s** для сохранения и **q** для выхода.

Получаем необходимый список:

```
  PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM    TIME+  COMMAND
 1407 dari      20   0 3440188 355780 127092 S   0,8   17,7   0:16.81 gnome-shell
    1 root       20   0 168144  11596   8488 S   0,0    0,6   0:08.35 systemd
 1247 dari      20   0 259860  74700  47364 S   0,8    3,7   0:06.54 Xorg
 1730 dari      20   0 814992  51224  38724 S   0,4    2,5   0:02.46 gnome-term+
 266 root       20   0  24772   7376   4036 S   0,0    0,4   0:02.35 systemd-ud+
```

4. Найти 2 процесса, имеющих более двух потоков. Использовать состояние процесса.

Устанавливаем приложение htop (улучшенный top) с помощью команды `sudo apt install htop`.

```
dari@Ubuntu:~$ sudo apt install htop
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  htop
0 upgraded, 1 newly installed, 0 to remove and 237 not upgraded.
Need to get 80,5 kB of archives.
After this operation, 225 kB of additional disk space will be used.
Get:1 http://ru.archive.ubuntu.com/ubuntu focal/main amd64 htop amd64 2.2.0-2build1 [80,5 kB]
Fetched 80,5 kB in 0s (331 kB/s)
Selecting previously unselected package htop.
(Reading database ... 179185 files and directories currently installed.)
Preparing to unpack .../htop_2.2.0-2build1_amd64.deb ...
Unpacking htop (2.2.0-2build1) ...
Setting up htop (2.2.0-2build1) ...
Processing triggers for desktop-file-utils (0.24-1ubuntu3) ...
Processing triggers for mime-support (3.64ubuntu1) ...
Processing triggers for gnome-menus (3.36.0-1ubuntu1) ...
Processing triggers for man-db (2.9.1-1) ...
```

Запускаем.

```
dari@Ubuntu:~$ htop
```

Нажимаем **f6** и стрелками выбираем **PERCENT_CPU**.

```

CPU[||| 1.6%] Tasks: 98, 213 thr; 1 running
Mem[||||| 749M/1.92G] Load average: 0.05 0.20 0.34
Swp[|| 47.9M/1.37G] Uptime: 00:57:08

Sort by PID USER PRI NI VIRT RES SHR S CPU% MEM% TIME+ C
PID 1407 dari 20 0 3638M 314M 89224 S 0.4 16.0 0:44.00 /
USER 1417 dari 20 0 3638M 314M 89224 S 0.0 16.0 0:00.10 /
PRIORITY 1421 dari 20 0 3638M 314M 89224 S 0.0 16.0 0:00.47 /
NICE 1422 dari 20 0 3638M 314M 89224 S 0.0 16.0 0:00.00 /
M_SIZE 1423 dari 39 19 3638M 314M 89224 S 0.0 16.0 0:00.00 /
M_RESIDENT 1424 dari 20 0 3638M 314M 89224 S 0.0 16.0 0:00.06 /
M_SHARE 1425 dari 20 0 3638M 314M 89224 S 0.0 16.0 0:00.06 /
STATE 2082 dari 20 0 3638M 314M 89224 S 0.0 16.0 0:00.00 /
PERCENT_CPU 1247 dari 20 0 253M 50904 26404 S 0.4 2.5 0:16.71 /
PERCENT_MEM 1284 dari 20 0 253M 50904 26404 S 0.0 2.5 0:02.51 /
TIME 23424 dari 20 0 795M 49100 36556 S 0.4 2.4 0:01.81 /
Command 23425 dari 20 0 795M 49100 36556 S 0.0 2.4 0:00.00 /
23426 dari 20 0 795M 49100 36556 S 0.0 2.4 0:00.00 /
23427 dari 20 0 795M 49100 36556 S 0.0 2.4 0:00.00 /
23431 dari 20 0 795M 49100 36556 S 0.0 2.4 0:00.00 /
1665 dari 20 0 774M 39652 23724 S 0.0 2.0 0:00.00 /
1666 dari 20 0 774M 39652 23724 S 0.0 2.0 0:00.00 /
EnterSort EscCancel
```

Нажимаем Enter.

CPU[2.0%]				Tasks: 98, 213 thr; 1 running							
Mem[747M/1.92G]				Load average: 0.00 0.06 0.23							
Swp[47.9M/1.37G]				Uptime: 01:03:14							
PID	USER	PRI	NI	VIRT	RES	SHR	S	CPU%	MEM%	TIME+	Command
44676	dari	20	0	10948	4104	3408	R	1.3	0.2	0:04.95	htop
1407	dari	20	0	3638M	314M	89224	S	0.7	16.0	0:45.40	/usr/bin/gnome-s
23424	dari	20	0	795M	49100	36556	S	0.7	2.4	0:02.46	/usr/libexec/gno
1247	dari	20	0	253M	50904	26404	S	0.0	2.5	0:17.66	/usr/lib/xorg/Xo
1233	dari	20	0	309M	7820	7392	S	0.0	0.4	0:00.22	/usr/libexec/gvf
1838	dari	20	0	486M	34508	24836	S	0.0	1.7	0:00.43	update-notifier
1284	dari	20	0	253M	50904	26404	S	0.0	2.5	0:02.73	/usr/lib/xorg/Xo
9297	root	20	0	1733M	20960	9192	S	0.0	1.0	0:00.55	/usr/lib/snapd/s
606	root	20	0	17476	7764	6624	S	0.0	0.4	0:00.28	/lib/systemd/sys
1417	dari	20	0	3638M	314M	89224	S	0.0	16.0	0:00.11	/usr/bin/gnome-s
1541	dari	20	0	307M	9132	8152	S	0.0	0.5	0:00.10	/usr/libexec/gsd

Нажимаем f10 для выхода, вводим команду ps с ключами **-o** – формат вывода, **thcount** – количество используемых процессом процессоров, **id** процесса (берем из списка).

```
dari@Ubuntu:~$ ps -o thcount 1407
THCNT
9
dari@Ubuntu:~$ ps -o thcount 23424
THCNT
5
```

5. Используя команду top, изменить приоритеты 2 процессов.

Вводим **top**, нажимаем Enter, затем **r**, чтобы использовать переназначение приоритета. Вводим PID процесса (у меня 44720), жмем Enter и задаем значение NICE (напр. 10).

Renice PID 44720 to value 10											
PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
44720	dari	20	0	12172	3676	3180	R	0,3	0,2	0:00.86	top
1	root	20	0	169584	12284	7708	S	0,0	0,6	10:13.35	systemd
2	root	20	0	0	0	0	S	0,0	0,0	0:00.00	kthreadd

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1407	dari	35	15	3745212	346884	91640	S	0,8	17,2	0:51.11	gnome-shell
559	root	20	0	2548	700	636	S	0,4	0,0	0:00.10	acpid
617	root	20	0	0	0	0	I	0,4	0,0	0:02.69	kworker/0:3-events
44720	dari	30	10	12172	3676	3180	R	0,4	0,2	0:00.90	top
1	root	20	0	169584	12284	7708	S	0,0	0,6	0:03.73	systemd

Приоритет виден в колонках PR и NI.

6. Получить список открытых файлов пользователя aa.

С помощью команды `lsdf` получаем список открытых файлов, ключ `-u` показывает список файлов заданного пользователя.

```
dari@Ubuntu:~$ lsdf -u dari
COMMAND  PID USER  FD      TYPE          DEVICE  SIZE/OFF      NODE NAME
systemd  1184 dari   cwd      DIR           8,5      4096           2 /
systemd  1184 dari   rtd      DIR           8,5      4096           2 /
systemd  1184 dari   txt      REG           8,5    1620224    1709628 /usr/lib/systemd/systemd
systemd  1184 dari   mem      REG           8,5    1369384    1710697 /usr/lib/x86_64-linux-gnu/li
bm-2.31.so
systemd  1184 dari   mem      REG           8,5     178528    1704573 /usr/lib/x86_64-linux-gnu/li
budev.so.1.6.17
systemd  1184 dari   mem      REG           8,5    1575112    1711229 /usr/lib/x86_64-linux-gnu/li
bunistring.so.2.1.0
systemd  1184 dari   mem      REG           8,5     137584    1710427 /usr/lib/x86_64-linux-gnu/li
bpng-0.5505.so.0.28.0
```

7. Получить текущее состояние системной памяти.

С помощью `free` выводим объемы оперативной и swap-памяти.

```
dari@Ubuntu:~$ free
              total        used        free      shared  buff/cache   available
Mem:           2011232       783072       112764        10460       1115396       1046028
Swap:          1435264         49096       1386168
```

8. Получить справку об использовании дискового пространства.

С помощью команды `df` выводим данные об объеме доступного дискового пространства в Кб, ключ `-h` улучшает восприятие результатов.

```
dari@Ubuntu:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
udev            937M   0    937M   0% /dev
tmpfs           197M  1,4M  196M   1% /run
/dev/sda5       29G   11G   17G  39% /
tmpfs           983M   0    983M   0% /dev/shm
tmpfs           5,0M  4,0K  5,0M   1% /run/lock
tmpfs           983M   0    983M   0% /sys/fs/cgroup
/dev/loop0      128K  128K   0 100% /snap/bare/5
/dev/loop1       64M   64M   0 100% /snap/core20/2682
/dev/loop2       64M   64M   0 100% /snap/core20/1828
/dev/loop3       74M   74M   0 100% /snap/core22/2139
/dev/loop7       92M   92M   0 100% /snap/gtk-common-themes/1535
/dev/loop4      517M  517M   0 100% /snap/gnome-42-2204/226
/dev/loop6      347M  347M   0 100% /snap/gnome-3-38-2004/119
/dev/loop5      350M  350M   0 100% /snap/gnome-3-38-2004/143
/dev/loop8       13M   13M   0 100% /snap/snap-store/1216
/dev/loop9       46M   46M   0 100% /snap/snap-store/638
/dev/loop10      51M   51M   0 100% /snap/snapd/25577
/dev/loop11      50M   50M   0 100% /snap/snapd/18357
/dev/sda1       511M  4,0K  511M   1% /boot/efi
tmpfs           197M  32K   197M   1% /run/user/1000
```


9. Вывести информацию о каком-либо процессе, используя содержимое каталога /proc.

Используя `ls -l /proc/10` для вывода информации о процессе с PID 10.

```
dari@Ubuntu:~$ ls -l /proc/10
ls: cannot read symbolic link '/proc/10/cwd': Permission denied
ls: cannot read symbolic link '/proc/10/root': Permission denied
ls: cannot read symbolic link '/proc/10/exe': Permission denied
total 0
-r--r--r-- 1 root root 0 ноя 8 21:01 arch_status
dr-xr-xr-x 2 root root 0 ноя 8 21:01 attr
-rw-r--r-- 1 root root 0 ноя 8 21:01 autogroup
-r----- 1 root root 0 ноя 8 21:01 auxv
-r--r--r-- 1 root root 0 ноя 8 21:01 cgroup
--w----- 1 root root 0 ноя 8 21:01 clear_refs
-r--r--r-- 1 root root 0 ноя 8 18:44 cmdline
-rw-r--r-- 1 root root 0 ноя 8 21:01 comm
-rw-r--r-- 1 root root 0 ноя 8 21:01 coredump_filter
-r--r--r-- 1 root root 0 ноя 8 21:01 cpu_resctrl_groups
-r--r--r-- 1 root root 0 ноя 8 21:01 cpuset
lrwxrwxrwx 1 root root 0 ноя 8 21:01 cwd
lrwxrwxrwx 1 root root 0 ноя 8 21:01 environ
```

10. Вывести информацию о процессоре ПК, используя содержимое каталога /proc.

Просматриваем с помощью `cat` файл `cpuinfo`, в котором содержится информация о процессоре.

```
dari@Ubuntu:~$ cat /proc/cpuinfo
processor       : 0
vendor_id     : GenuineIntel
cpu family    : 6
model         : 140
model name    : 11th Gen Intel(R) Core(TM) i5-1135G7 @ 2.40GHz
stepping      : 1
microcode     : 0x52
cpu MHz       : 2419.200
cache size    : 8192 KB
physical id   : 0
siblings      : 1
core id       : 0
cpu cores     : 1
apicid        : 0
initial apicid : 0
fpu           : yes
fpu_exception : yes
```

11. Вывести список модулей, используемых в настоящий момент ядром ОС.

Аналогично просматриваем файл **modules**.

```
dari@Ubuntu:~$ cat /proc/modules
nls_iso8859_1 16384 1 - Live 0x0000000000000000
binfmt_misc 24576 1 - Live 0x0000000000000000
snd_intel8x0 49152 2 - Live 0x0000000000000000
snd_ac97_codec 155648 1 snd_intel8x0, Live 0x0000000000000000
ac97_bus 16384 1 snd_ac97_codec, Live 0x0000000000000000
vmwgfx 364544 2 - Live 0x0000000000000000
snd_pcm 135168 2 snd_intel8x0,snd_ac97_codec, Live 0x0000000000000000
intel_rapl_msr 20480 0 - Live 0x0000000000000000
intel_rapl_common 40960 1 intel_rapl_msr, Live 0x0000000000000000
snd_seq_midi 20480 0 - Live 0x0000000000000000
intel_powerclamp 24576 0 - Live 0x0000000000000000
snd_seq_midi_event 16384 1 snd_seq_midi, Live 0x0000000000000000
crct10dif_pclmul 16384 1 - Live 0x0000000000000000
ghash_clmulni_intel 16384 0 - Live 0x0000000000000000
sha256_ssse3 32768 0 - Live 0x0000000000000000
sha1_ssse3 32768 0 - Live 0x0000000000000000
l... i... 16150 1 - Live 0x0000000000000000
```