

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №7.

КОНТРОЛЬ ИСПОЛЬЗОВАНИЯ РЕСУРСОВ ОС LINUX.

ЦЕЛЬ РАБОТЫ – практическое знакомство с командами, используемыми для контроля использования ресурсов и виртуальной файловой системой /proc.

МЕТОДИКА ВЫПОЛНЕНИЯ

1. Вывести список всех процессоров системы.

```
vboxuser@tusabusa:~/Desktop$ ps aux
```

Для просмотра списка всех процессов системы обращаемся к утилите **ps**. Ключ **a** добавляет процессы всех пользователей, **u** выводит информацию о владельце, а ключ **x** отвечает за добавление процессов, у которых нет привязанного терминала.

Готовый результат:

```
vboxuser@tusabusa:~/Desktop$ ps aux
```

USER	PID	%CPU	%MEM	VSZ	RSS	TTY	STAT	START	TIME	COMMAND
root	1	1.4	0.2	23088	13860	?	Ss	17:02	0:03	/sbin/init sp
root	2	0.0	0.0	0	0	?	S	17:02	0:00	[kthreadd]
root	3	0.0	0.0	0	0	?	S	17:02	0:00	[pool_workque
root	4	0.0	0.0	0	0	?	I<	17:02	0:00	[kworker/R-rc
root	5	0.0	0.0	0	0	?	I<	17:02	0:00	[kworker/R-sy
root	6	0.0	0.0	0	0	?	I<	17:02	0:00	[kworker/R-kv
root	7	0.0	0.0	0	0	?	I<	17:02	0:00	[kworker/R-sl
root	8	0.0	0.0	0	0	?	I<	17:02	0:00	[kworker/R-ne
root	9	0.0	0.0	0	0	?	I	17:02	0:00	[kworker/0:0-
root	10	0.0	0.0	0	0	?	I	17:02	0:00	[kworker/0:1-
root	11	0.0	0.0	0	0	?	I<	17:02	0:00	[kworker/0:0H
root	12	0.2	0.0	0	0	?	I	17:02	0:00	[kworker/u8:0
root	13	0.0	0.0	0	0	?	I<	17:02	0:00	[kworker/R-mm
root	14	0.0	0.0	0	0	?	I	17:02	0:00	[rcu_tasks_kt
root	15	0.0	0.0	0	0	?	I	17:02	0:00	[rcu_tasks_ru
root	16	0.0	0.0	0	0	?	I	17:02	0:00	[rcu_tasks_tr
root	17	0.0	0.0	0	0	?	S	17:02	0:00	[ksoftirqd/0]
root	18	0.3	0.0	0	0	?	I	17:02	0:00	[rcu_preempt]
root	19	0.0	0.0	0	0	?	S	17:02	0:00	[rcu_exp_par_

2. Вывести дерево процессов.

```
vboxuser@tusabusa:~/Desktop$ pstree
```

Данная команда показывает процессы в виде дерева.

Готовый результат:

```
vboxuser@tusabusa:~/Desktop$ pstree
systemd├─ModemManager─3*[{ModemManager}]
      │NetworkManager─3*[{NetworkManager}]
      │accounts-daemon─3*[{accounts-daemon}]
      │avahi-daemon─avahi-daemon
      │colord─3*[{colord}]
      │cron
      │cups-browsed─3*[{cups-browsed}]
      │cupsd─3*[dbus]
      │dbus-daemon
      │gdm3├─gdm-session-wor├─gdm-wayland-ses├─gnome-session-b─3*[{gnome-+
      │││3*[{gdm3}]││3*[{gdm-session-wor}]│3*[{gdm-wayland-ses}]
      │└─gnome-remote-de─3*[{gnome-remote-de}]
      │2*[kerneloops]
      │polkitd─3*[{polkitd}]
      │power-profiles─3*[{power-profiles-}]
      │rsyslogd─3*[{rsyslogd}]
      └─rtkit-daemon─2*[{rtkit-daemon}]
```

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

```
vboxuser@tusabusa:~/Desktop$ top
```

Запускаем программу мониторинга процессов командой **top**. С помощью комбинации **Shift + T** сортируем строки по столбцу **TIME+** (накопленное процессорное время). Первые 5 процессов потребляют наибольшее количество процессорного времени.

Готовый результат:

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1884	vboxuser	20	0	3902132	351372	142160	S	4.7	5.9	0:17.44	gnome-s+
2181	vboxuser	20	0	420952	28508	17368	S	0.0	0.5	0:04.43	ibus-ex+
1	root	20	0	23112	14092	9484	S	0.0	0.2	0:03.24	systemd
817	root	20	0	1848692	37744	24748	S	0.0	0.6	0:02.62	snappd
2621	vboxuser	20	0	553592	52620	41992	S	1.0	0.9	0:01.59	gnome-t+
2690	vboxuser	20	0	14536	5824	3648	R	0.3	0.1	0:01.44	top
1644	vboxuser	20	0	21100	12500	9428	S	0.0	0.2	0:01.20	systemd
2550	vboxuser	20	0	2812524	67004	51676	S	0.0	1.1	0:01.16	gjs
813	polkitd	20	0	384648	11000	7792	S	0.0	0.2	0:01.12	polkitd
794	message+	20	0	12096	6944	4384	S	0.0	0.1	0:01.06	dbus-da+
1674	vboxuser	20	0	10704	6420	4628	S	0.0	0.1	0:00.85	dbus-da+
244	root	19	-1	50908	17048	15640	S	0.0	0.3	0:00.74	systemd+
18	root	20	0	0	0	0	I	0.0	0.0	0:00.72	rcu_pre+
314	root	20	0	30580	8484	4900	S	0.0	0.1	0:00.43	systemd+
64	root	20	0	0	0	0	I	0.0	0.0	0:00.42	kworker+
2428	vboxuser	20	0	429616	30556	21960	S	0.0	0.5	0:00.42	snappd+
493	root	20	0	0	0	0	I	0.3	0.0	0:00.40	kworker+

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

Используем интерактивную программу **htop**. Вызываем программу командой **htop**.

```
vboxuser@tusabusa:~/Desktop$ htop
```

В открывшемся окне переходим в сортировку(F6) и выбираем сортировку по столбцу CPU%(percent_cpu).

System Statistics:

- Tasks: 111, 329 thr, 77 kthr; 1 running
- Load average: 0.05 0.04 0.05
- Uptime: 01:36:53
- Mem: 821M/5.69G
- Swp: 0K/0K

Sort by	PID	USER	PRI	NI	VIRT	RES	SHR	S	CPU%	MEM%	TIME+
PID	3877	vboxuser	20	0	11180	4760	3608	R	3.3	0.1	0:02.9
USER	1918	vboxuser	20	0	3835M	373M	139M	S	2.6	6.4	0:22.3
PRIORITY	1884	vboxuser	20	0	3835M	373M	139M	S	2.0	6.4	0:54.3
NICE	1919	vboxuser	20	0	3835M	373M	139M	S	2.0	6.4	0:22.9
M_VIRT	3149	vboxuser	20	0	543M	54896	42500	S	0.7	0.9	0:05.9
M_RESIDENT	1	root	20	0	23112	14092	9484	S	0.0	0.2	0:03.6
M_SHARE	244	root	19	-1	50908	17432	15896	S	0.0	0.3	0:00.8
STATE	314	root	20	0	30580	8484	4900	S	0.0	0.1	0:00.4
PERCENT_CPU	366	systemd-oo	20	0	17560	7572	6676	S	0.0	0.1	0:03.0
PERCENT_MEM	371	systemd-re	20	0	21580	13068	10764	S	0.0	0.2	0:00.3
TIME	373	systemd-ti	20	0	91048	7800	6904	S	0.0	0.1	0:00.1
Command	453	systemd-ti	20	0	91048	7800	6904	S	0.0	0.1	0:00.0
	793	avahi	20	0	8668	4188	3932	S	0.0	0.1	0:00.1
	794	messagebus	20	0	12192	7200	4384	S	0.0	0.1	0:01.6
	797	gnome-remo	20	0	356M	16340	13908	S	0.0	0.3	0:00.1

После сортировки нажимаем **enter** и в выведенном списке процессов запоминаем PID двух процессов(необходимо для последующей проверки количества потоков), затем выходим из программы(F10).

System Statistics:

- Tasks: 111, 329 thr, 77 kthr; 1 running
- Load average: 0.03 0.08 0.07
- Uptime: 01:46:36
- Mem: 834M/5.69G
- Swp: 0K/0K

PID	USER	PRI	NI	VIRT	RES	SHR	S	CPU%	MEM%	TIME+	Command
3980	vboxuser	20	0	11200	4608	3456	R	3.3	0.1	0:00.84	htop
366	systemd-oo	20	0	17560	7572	6676	S	0.7	0.1	0:03.26	/usr/lib/syst
1918	vboxuser	20	0	3835M	373M	139M	S	0.7	6.4	0:24.74	/usr/bin/gnom
1	root	20	0	23112	14092	9484	S	0.0	0.2	0:03.66	/sbin/init sp
244	root	19	-1	50908	17432	15896	S	0.0	0.3	0:00.89	/usr/lib/syst
314	root	20	0	30580	8484	4900	S	0.0	0.1	0:00.44	/usr/lib/syst
371	systemd-re	20	0	21580	13068	10764	S	0.0	0.2	0:00.30	/usr/lib/syst
373	systemd-ti	20	0	91048	7800	6904	S	0.0	0.1	0:00.14	/usr/lib/syst
453	systemd-ti	20	0	91048	7800	6904	S	0.0	0.1	0:00.00	/usr/lib/syst
793	avahi	20	0	8668	4188	3932	S	0.0	0.1	0:00.11	avahi-daemon:
794	messagebus	20	0	12192	7200	4384	S	0.0	0.1	0:01.67	@dbus-daemon
797	gnome-remo	20	0	356M	16340	13908	S	0.0	0.3	0:00.10	/usr/libexec/
813	polkitd	20	0	381M	12212	8108	S	0.0	0.2	0:00.98	/usr/lib/polkit
814	root	20	0	306M	7420	6652	S	0.0	0.1	0:00.10	/usr/libexec/
817	root	20	0	1805M	36864	24748	S	0.0	0.6	0:01.72	/usr/lib/snap

Далее проверяем выбранные процессы на количество потоков. В терминале вводим команды:

```
vboxuser@tusabusa:~/Desktop$ ps -o thcount 1884
THCNT
18
```

```
vboxuser@tusabusa:~/Desktop$ ps -o thcount 3149
THCNT
6
```

Команда выводит информацию о процессе и показывает количество потоков. В конце команды указан **PID** процесса, который мы взяли из htop.

Проверка показала, что выбранные процессы многопоточные (имеют более двух потоков).

5. Используя команду **top**, изменить приоритеты 2 процессов.

Запускаем программу **top** от имени администратора, для этого вводим команду **sudo top**(необходимо для изменения приоритета). Затем нажимаем **R**(renice) и меняем приоритет у таких процессов, где PID=4 и PID=3.

```
top - 20:40:31 up 35 min, 1 user, load average: 0.14, 0.17, 0.22
Tasks: 190 total, 1 running, 189 sleeping, 0 stopped, 0 zombie
%Cpu(s): 1.7 us, 2.2 sy, 0.0 ni, 96.0 id, 0.0 wa, 0.0 hi, 0.2 si, 0.0 st
MiB Mem : 5825.6 total, 3995.5 free, 1052.2 used, 1038.1 buff/cache
MiB Swap: 0.0 total, 0.0 free, 0.0 used, 4773.4 avail Mem

PID to renice [default pid = 1918]
PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
1918 vboxuser 20 0 3919308 382436 143256 S 6.2 6.4 1:10.43 gnome-s+
2738 vboxuser 20 0 553700 52792 41916 S 2.0 0.9 0:07.39 gnome-t+
17 root 20 0 0 0 0 S 0.3 0.0 0:00.18 ksoftir+
34 root 20 0 0 0 0 I 0.3 0.0 0:01.37 kworker+
846 root 20 0 1922424 38520 25256 S 0.3 0.6 0:07.65 snapd
889 root 20 0 336580 19392 16320 S 0.3 0.3 0:00.42 Network+
2953 root 20 0 14536 5836 3660 R 0.3 0.1 0:02.30 top
3071 root 20 0 0 0 0 I 0.3 0.0 0:00.78 kworker+
1 root 20 0 23108 14120 9384 S 0.0 0.2 0:04.04 systemd
2 root 20 0 0 0 0 S 0.0 0.0 0:00.00 kthreadd
3 root 20 0 0 0 0 S 0.0 0.0 0:00.00 pool_wo+
4 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 kworker+
5 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 kworker+
6 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 kworker+
7 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 kworker+
8 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 kworker+
10 root 20 0 0 0 0 I 0.0 0.0 0:01.19 kworker+
```

Готовый результат:

```
top - 20:43:09 up 38 min, 1 user, load average: 0.27, 0.20, 0.22
Tasks: 190 total, 1 running, 189 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.8 us, 0.3 sy, 0.0 ni, 98.7 id, 0.0 wa, 0.0 hi, 0.2 si, 0.0 st
MiB Mem : 5825.6 total, 3995.5 free, 1052.1 used, 1038.1 buff/cache
MiB Swap: 0.0 total, 0.0 free, 0.0 used. 4773.4 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1918	vboxuser	20	0	3923468	382516	143256	S	2.0	6.4	1:19.42	gnome-s+
2738	vboxuser	20	0	553832	52792	41916	S	0.3	0.9	0:08.73	gnome-t+
2953	root	20	0	14536	5836	3660	R	0.3	0.1	0:02.47	top
1	root	20	0	23108	14120	9384	S	0.0	0.2	0:04.04	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kthreadd
3	root	20	0	0	0	0	S	0.0	0.0	0:00.00	pool_wo+
4	root	30	10	0	0	0	I	0.0	0.0	0:00.00	kworker+
5	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker+
6	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker+
7	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker+
8	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker+
10	root	20	0	0	0	0	I	0.0	0.0	0:01.27	kworker+
13	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker+
14	root	20	0	0	0	0	I	0.0	0.0	0:00.00	rcu_tas+
15	root	20	0	0	0	0	I	0.0	0.0	0:00.00	rcu_tas+
16	root	20	0	0	0	0	I	0.0	0.0	0:00.00	rcu_tas+
17	root	20	0	0	0	0	S	0.0	0.0	0:00.18	ksoftir+

6. Получить список открытых файлов пользователя aa.

Для получения списка открытых файлов конкретного пользователя используется команда **lsuf -u имя_пользователя**

```
vboxuser@tusabusa:~/Desktop$ lsuf -u vboxuser
```

Готовый результат:

```
dbus-daem 1708 vboxuser mem REG 8,2 910592 1578747 /usr/lib/x86_64-linux-gnu/libsystemd.so.0.3
8.0
dbus-daem 1708 vboxuser mem REG 8,2 317752 1578255 /usr/lib/x86_64-linux-gnu/libdbus-1.so.3.32
.4
dbus-daem 1708 vboxuser mem REG 8,2 236616 1569182 /usr/lib/x86_64-linux-gnu/ld-linux-x86-64.s
o.2
dbus-daem 1708 vboxuser 0u CHR 1,3 0t0 5 /dev/null
dbus-daem 1708 vboxuser 1u unix 0x0000000000000000 0t0 13909 type=STREAM (CONNECTED)
dbus-daem 1708 vboxuser 2u unix 0x0000000000000000 0t0 13909 type=STREAM (CONNECTED)
dbus-daem 1708 vboxuser 3u unix 0x0000000000000000 0t0 13731 /run/user/1000/bus type=STREAM (LISTEN)
dbus-daem 1708 vboxuser 4u a_inode 0,16 0 44 [eventpoll:3,6,7,9,10,11,12,13,14,15,16,17,
18,19,20,21,24,25,73,74,75,76,77,78,79,80,81,82,83,84,85,86...]
dbus-daem 1708 vboxuser 5u unix 0x0000000000000000 0t0 13924 type=DGRAM (CONNECTED)
dbus-daem 1708 vboxuser 6r a_inode 0,16 0 44 inotify
dbus-daem 1708 vboxuser 7u unix 0x0000000000000000 0t0 13925 type=STREAM (CONNECTED)
dbus-daem 1708 vboxuser 8u unix 0x0000000000000000 0t0 13926 type=STREAM (CONNECTED)
dbus-daem 1708 vboxuser 9u unix 0x0000000000000000 0t0 13095 /run/user/1000/bus type=STREAM (CONNECTED)
dbus-daem 1708 vboxuser 10u unix 0x0000000000000000 0t0 13102 /run/user/1000/bus type=STREAM (CONNECTED)
dbus-daem 1708 vboxuser 11u unix 0x0000000000000000 0t0 13931 /run/user/1000/bus type=STREAM (CONNECTED)
dbus-daem 1708 vboxuser 12u unix 0x0000000000000000 0t0 13124 /run/user/1000/bus type=STREAM (CONNECTED)
dbus-daem 1708 vboxuser 13u unix 0x0000000000000000 0t0 13130 /run/user/1000/bus type=STREAM (CONNECTED)
dbus-daem 1708 vboxuser 14u unix 0x0000000000000000 0t0 13218 /run/user/1000/bus type=STREAM (CONNECTED)
dbus-daem 1708 vboxuser 15u unix 0x0000000000000000 0t0 14107 /run/user/1000/bus type=STREAM (CONNECTED)
dbus-daem 1708 vboxuser 16u unix 0x0000000000000000 0t0 14109 /run/user/1000/bus type=STREAM (CONNECTED)
dbus-daem 1708 vboxuser 17u unix 0x0000000000000000 0t0 13146 /run/user/1000/bus type=STREAM (CONNECTED)
dbus-daem 1708 vboxuser 18u unix 0x0000000000000000 0t0 14378 /run/user/1000/bus type=STREAM (CONNECTED)
```


7. Получить текущее состояние системной памяти

С помощью команды **free** выводим на экран текущее состояние системной памяти.

Готовый результат:

```
vboxuser@tusabusa:~/Desktop$ free
```

	total	used	free	shared	buff/cache	available
Mem:	5965380	1077672	4070016	36632	1085984	4887708
Swap:	0	0	0			

8. Получить справку об использовании дискового пространства.

Команда **df** с ключом **-h** выводит нужную справку.

Готовый результат:

```
vboxuser@tusabusa:~/Desktop$ df -h
```

Filesystem	Size	Used	Avail	Use%	Mounted on
tmpfs	583M	1.5M	582M	1%	/run
/dev/sda2	27G	5.8G	20G	23%	/
tmpfs	2.9G	0	2.9G	0%	/dev/shm
tmpfs	5.0M	8.0K	5.0M	1%	/run/lock
tmpfs	583M	116K	583M	1%	/run/user/1000

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

Для вывода информации о процессе **Pid=24**, который является содержимым каталога **/proc** используется команда **ls** с ключом **-l**

Готовый результат:

```
vboxuser@tusabusa:~/Desktop$ ls -l /proc/24
```

```
ls: cannot read symbolic link '/proc/24/cwd': Permission denied
ls: cannot read symbolic link '/proc/24/root': Permission denied
ls: cannot read symbolic link '/proc/24/exe': Permission denied
total 0
```

-r--r--r--	1	root	root	0	Nov 10 21:19	arch_status
dr-xr-xr-x	2	root	root	0	Nov 10 21:19	attr
-rw-r--r--	1	root	root	0	Nov 10 21:19	autogroup
-r-----	1	root	root	0	Nov 10 21:19	auxv
-r--r--r--	1	root	root	0	Nov 10 20:05	cgroup
--w-----	1	root	root	0	Nov 10 21:19	clear_refs
-r--r--r--	1	root	root	0	Nov 10 20:05	cmdline
-rw-r--r--	1	root	root	0	Nov 10 21:19	comm
-rw-r--r--	1	root	root	0	Nov 10 21:19	coredump_filter
-r--r--r--	1	root	root	0	Nov 10 21:19	cpu_resctrl_groups
-r--r--r--	1	root	root	0	Nov 10 21:19	cpuset
lrwxrwxrwx	1	root	root	0	Nov 10 20:07	cwd
-r-----	1	root	root	0	Nov 10 21:19	environ

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

Для вывода информации о процессоре используется команда **cat**, которая показывает содержимое файла из каталога `/proc`.

Готовый результат:

```
vboxuser@tusabusa:~/Desktop$ cat /proc/cpuinfo
processor       : 0
vendor_id      : AuthenticAMD
cpu family     : 23
model          : 24
model name     : AMD Ryzen 5 3500U with Radeon Vega Mobile Gfx
stepping       : 1
microcode      : 0x8108102
cpu MHz        : 2096.062
cache size     : 512 KB
physical id    : 0
siblings       : 2
core id        : 0
cpu cores      : 2
apicid         : 0
initial apicid : 0
fpu            : yes
fpu_exception  : yes
cpuid level    : 13
wp             : yes
flags          : fpu_vme_de_pse_tsc_mce_mca_cy8_rip_sse_mtrr
```

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

Для вывода списка модулей используется команда **cat**, которая показывает содержимое из каталога `/proc`.

Готовый результат:

```
vboxuser@tusabusa:~/Desktop$ cat /proc/modules
snd_seq_dummy 12288 0 - Live 0x0000000000000000
snd_hrtimer 12288 1 - Live 0x0000000000000000
qrtr 53248 2 - Live 0x0000000000000000
snd_intel8x0 53248 1 - Live 0x0000000000000000
snd_ac97_codec 196608 1 snd_intel8x0, Live 0x0000000000000000
ac97_bus 12288 1 snd_ac97_codec, Live 0x0000000000000000
intel_rapl_msr 20480 0 - Live 0x0000000000000000
intel_rapl_common 53248 1 intel_rapl_msr, Live 0x0000000000000000
snd_pcm 192512 2 snd_intel8x0,snd_ac97_codec, Live 0x0000000000000000
snd_seq_midi 24576 0 - Live 0x0000000000000000
snd_seq_midi_event 16384 1 snd_seq_midi, Live 0x0000000000000000
snd_rawmidi 57344 1 snd_seq_midi, Live 0x0000000000000000
snd_seq 122880 9 snd_seq_dummy,snd_seq_midi,snd_seq_midi_event, Live 0x0000000000000000
joydev 32768 0 - Live 0x0000000000000000
vmwgfx 466944 1 - Live 0x0000000000000000
polyval_clmulni 12288 0 - Live 0x0000000000000000
```

