

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «КубГУ»)

Факультет компьютерных технологий и прикладной математики
Кафедра информационных технологий

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 6
«ПРОЦЕССЫ. РАБОТА С ПРОЦЕССАМИ»
по дисциплине
«ОПЕРАЦИОННЫЕ СИСТЕМЫ»

Выполнил студент группы МО-32/2 _____ С. Е. Пятунин

Направление подготовки 02.03.03 Математическое обеспечение и администрирование информационных систем
Курс 3

Отчет принял _____ А. А. Полупанов

Краснодар
2025 г.

Процессы в Linux

Для просмотра таблицы процессов в Linux предназначена утилита *ps*.

Один из наиболее часто используемых ключей **aux**:

```
s0188249@dc-1:~$ ps aux | head -20
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root        1  0.0  0.4 145952  8936 ?        Ss   08:21  0:00 /sbin/init
root        2  0.0  0.0     0     0 ?        S    08:21  0:00 [kthreadd]
root        3  0.0  0.0     0     0 ?        I<  08:21  0:00 [rcu_gp]
root        4  0.0  0.0     0     0 ?        I<  08:21  0:00 [rcu_par_gp]
root        5  0.0  0.0     0     0 ?        I<  08:21  0:00 [slub_flushwq]
root        6  0.0  0.0     0     0 ?        I<  08:21  0:00 [netns]
root        8  0.0  0.0     0     0 ?        I<  08:21  0:00 [kworker/0:0H-ev]
root       10  0.0  0.0     0     0 ?        I<  08:21  0:00 [mm_percpu_wq]
root       11  0.0  0.0     0     0 ?        S    08:21  0:00 [rcu_tasks_rude_]
root       12  0.0  0.0     0     0 ?        S    08:21  0:00 [rcu_tasks_trace]
root       13  0.0  0.0     0     0 ?        S    08:21  0:00 [ksoftirqd/0]
root       14  0.0  0.0     0     0 ?        I    08:21  0:00 [rcu_sched]
root       15  0.0  0.0     0     0 ?        S    08:21  0:00 [migration/0]
root       16  0.0  0.0     0     0 ?        S    08:21  0:00 [idle_inject/0]
root       18  0.0  0.0     0     0 ?        S    08:21  0:00 [cpuhp/0]
root       19  0.0  0.0     0     0 ?        S    08:21  0:00 [cpuhp/1]
root       20  0.0  0.0     0     0 ?        S    08:21  0:00 [idle_inject/1]
root       21  0.0  0.0     0     0 ?        S    08:21  0:00 [migration/1]
root       22  0.0  0.0     0     0 ?        S    08:21  0:00 [ksoftirqd/1]
s0188249@dc-1:~$
```

Рисунок 1 – Просмотр процессов

Вывод процессов с идентификатором родителя:

```
s0188249@dc-1:~$ ps -ef | head -20
UID      PID  PPID  C STIME TTY      TIME CMD
root        1      0  0 08:21 ?        00:00:00 /sbin/init
root        2      0  0 08:21 ?        00:00:00 [kthreadd]
root        3      2  0 08:21 ?        00:00:00 [rcu_gp]
root        4      2  0 08:21 ?        00:00:00 [rcu_par_gp]
root        5      2  0 08:21 ?        00:00:00 [slub_flushwq]
root        6      2  0 08:21 ?        00:00:00 [netns]
root        8      2  0 08:21 ?        00:00:00 [kworker/0:0H-ev]
root       10      2  0 08:21 ?        00:00:00 [mm_percpu_wq]
root       11      2  0 08:21 ?        00:00:00 [rcu_tasks_rude_]
root       12      2  0 08:21 ?        00:00:00 [rcu_tasks_trace]
root       13      2  0 08:21 ?        00:00:00 [ksoftirqd/0]
root       14      2  0 08:21 ?        00:00:00 [rcu_sched]
root       15      2  0 08:21 ?        00:00:00 [migration/0]
root       16      2  0 08:21 ?        00:00:00 [idle_inject/0]
root       18      2  0 08:21 ?        00:00:00 [cpuhp/0]
root       19      2  0 08:21 ?        00:00:00 [cpuhp/1]
root       20      2  0 08:21 ?        00:00:00 [idle_inject/1]
root       21      2  0 08:21 ?        00:00:00 [migration/1]
root       22      2  0 08:21 ?        00:00:00 [ksoftirqd/1]
s0188249@dc-1:~$
```

Рисунок 2 – Вывод с ключом -ef

Используя значение PPID, можно легко найти все процессы, запущенные из текущей оболочки. Можно воспользоваться специальной утилитой pgrep и системной переменной \$\$, в которой содержится идентификатор текущего процесса:

```
s0188249@dc-1:~$ ps -f -p $$  
UID          PID  PPID  C STIME TTY          TIME CMD  
s0188249  19860 19857  0 09:43 pts/0    00:00:00 /bin/bash  
s0188249@dc-1:~$
```

Рисунок 3 – Все процессы, запущенные из текущей оболочки

С помощью утилиты pstree можно вывести список всех потомков процесса с PID=0, которые были порождены ядром системы:

```
s0188249@dc-1:~$ pstree -p 0 | head -30  
?()---kthreadd(2)---acpi_thermal_pm(97)  
|           |-ata_sff(84)  
|           |-blkcg_punt_bio(82)  
|           |-card0-crtc0(176)  
|           |-card0-crtc1(177)  
|           |-card0-crtc2(178)  
|           |-card0-crtc3(179)  
|           |-card0-crtc4(180)  
|           |-card0-crtc5(181)  
|           |-card0-crtc6(182)  
|           |-card0-crtc7(183)  
|           |-charger_manager(132)  
|           |-cpuhp/0(18)  
|           |-cpuhp/1(19)  
|           |-cryptd(413)  
|           |-devfreq_wq(87)  
|           |-ecryptfs-kthrea(94)  
|           |-edac-poller(86)  
|           |-ext4-rsv-conver(225)  
|           |-idle_inject/0(16)  
|           |-idle_inject/1(20)  
|           |-inet_frag_wq(26)  
|           |-ipv6_addrconf(109)  
|           |-irq/18-vmwgfx(175)  
|           |-jbd2/sda1-8(224)  
|           |-kauditfd(27)  
|           |-kblockd(81)  
|           |-kcompactd0(31)  
|           |-kdevtmpfs(25)  
|           |-khugepaged(33)  
s0188249@dc-1:~$
```

Рисунок 4 – Вывод утилиты pstree

Адресное пространство

Обычно процессы могут аллоцировать весь доступный объем памяти, и ничего настраивать дополнительно не требуется.

```
s0188249@dc-1:~$ ulimit -a
core file size          (blocks, -c) 0
data seg size           (kbytes, -d) unlimited
scheduling priority     (-e) 0
file size               (blocks, -f) unlimited
pending signals          (-i) 7527
max locked memory       (kbytes, -l) 64
max memory size         (kbytes, -m) unlimited
open files              (-n) 1024
pipe size               (512 bytes, -p) 8
POSIX message queues    (bytes, -q) 819200
real-time priority      (-r) 0
stack size              (kbytes, -s) 8192
cpu time                (seconds, -t) unlimited
max user processes       (-u) 7527
virtual memory           (kbytes, -v) unlimited
file locks              (-x) unlimited
s0188249@dc-1:~$
```

Рисунок 5 – Значение, установленное для «virtual memory»

Сигналы для процессов в Linux

Запускаем калькулятор в фоновом режиме:

```
s0188249@dc-1:~$ kcalc &
[1] 19878
```

Рисунок 6 – Запуск процесса в фоновом режиме

Завершаем процесс с выбранным PID 22416 из прошлой команды:

```
s0188249@dc-1:~$ kill -SIGTERM 19878
```

Рисунок 7 – Завершение процесса

Процессы Linux поддерживают 64 сигнала, список которых можно посмотреть с помощью ключа -L (-l, --list) команды kill:

```
s0188249@dc-1:~$ kill -l
 1) SIGHUP      2) SIGINT      3) SIGQUIT      4) SIGILL      5) SIGTRAP
 6) SIGABRT     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    29) SIGIO       30) SIGPWR
31) SIGSYS      34) SIGRTMIN    35) SIGRTMIN+1  36) SIGRTMIN+2  37) SIGRTMIN+3
38) SIGRTMIN+4  39) SIGRTMIN+5  40) SIGRTMIN+6  41) SIGRTMIN+7  42) SIGRTMIN+8
43) SIGRTMIN+9  44) SIGRTMIN+10 45) SIGRTMIN+11 46) SIGRTMIN+12 47) SIGRTMIN+13
48) SIGRTMIN+14 49) SIGRTMIN+15 50) SIGRTMAX-14 51) SIGRTMAX-13 52) SIGRTMAX-12
53) SIGRTMAX-11 54) SIGRTMAX-10 55) SIGRTMAX-9  56) SIGRTMAX-8  57) SIGRTMAX-7
58) SIGRTMAX-6  59) SIGRTMAX-5  60) SIGRTMAX-4  61) SIGRTMAX-3  62) SIGRTMAX-2
63) SIGRTMAX-1  64) SIGRTMAX
s0188249@dc-1:~$
```

Рисунок 8 – Сигналы, поддерживаемые процессами Linux

Для того чтобы приложение игнорировало сигнал -1, его можно запустить с помощью команды nohup. Если закрыть терминал, такие процессы «осиротеют» и будут узочерены процессом init (systemd).

```
root@dc-1:~# kcalc &
[1] 19898
```

Рисунок 9 – Запуск калькулятора в фоновом режиме

```
s0188249@dc-1:~$ kill -SIGHUP 19898
```

Рисунок 10 – Завершение работы приложения

```
s0188249@dc-1:~$ nohup kcalc &
[1] 19902
s0188249@dc-1:~$ nohup: ввод игнорируется, вывод добавляется в 'nohup.out'
```

Рисунок 11 – Запуск команды через nohup

Завершить процесс с помощью сигнала HUP больше не удается, он игнорируется приложением:

```
s0188249@dc-1:~$ kill -SIGHUP 19902
```

Рисунок 12 – Попытка завершить процесс

Планировщик задач в Linux и управление приоритетами процессов

Группы процессов FIFO, RR и Other соответствуют политикам планирования SCHED_FIFO , SCHED_RR и SCHED_OTHER (всего таких политик 6). Посмотреть список политик планирования можно командой chrt -m.

```
s0188249@dc-1:~$ chrt -m
SCHED_OTHER min/max priority : 0/0
SCHED_FIFO min/max priority : 1/99
SCHED_RR min/max priority : 1/99
SCHED_BATCH min/max priority : 0/0
SCHED_IDLE min/max priority : 0/0
SCHED_DEADLINE min/max priority : 0/0
s0188249@dc-1:~$ █
```

Рисунок 13 – Просмотр списка политик планирования

Просмотр фоновых заданий выполняется командой jobs.

```
s0188249@dc-1:~$ sleep 3000 &
[1] 19906
```

Рисунок 14 – Запуск 1-го процесса

```
s0188249@dc-1:~$ sleep 3000 &
[2] 19907
s0188249@dc-1:~$ sleep 3000 &
[3] 19908
```

Рисунок 15 – Запуск 2-го и 3-го процессов

```
s0188249@dc-1:~$ jobs
[1]  Запущен           sleep 3000 &
[2]- Запущен           sleep 3000 &
[3]+ Запущен           sleep 3000 &
s0188249@dc-1:~$ █
```

Рисунок 16 – Список заданий

Извлечение информации о процессах

Разберем, что хранится в каталогах /proc/PID/, где PID – числовой идентификатор процесса.

```
s0188249@dc-1:~$ sudo ls /proc/1
arch_status  clear_refs      cpuset    fdinfo    map_files
attr          cmdline        cwd       gid_map   maps
autogroup     comm          environ   io        mem
auxv          coredump_filter exe      limits    mountinfo
cgroup        cpu_resctrl_groups fd      loginuid mounts
s0188249@dc-1:~$ █
```

Рисунок 17 – Содержимое /proc/PID/

```
s0188249@dc-1:~$ cat /proc/1/cmdline && echo
/sbin/init
s0188249@dc-1:~$ █
```

Рисунок 18 – Стока запуска процесса

```
s0188249@dc-1:~$ sudo ls -l --color=always /proc/1/exe
lrwxrwxrwx 1 root root 0 дек 16 08:21 /proc/1/exe -> /lib/systemd/systemd
s0188249@dc-1:~$ █
```

Рисунок 19 – Символическая ссылка, ведущая к полному пути до исполняемого файла

```
s0188249@dc-1:~$ sudo ls -l --color=always /proc/1/cwd
lrwxrwxrwx 1 root root 0 дек 16 09:58 /proc/1/cwd -> /
s0188249@dc-1:~$ █
```

Рисунок 20 – Текущий рабочий каталог процесса

```
s0188249@dc-1:~$ sudo cat /proc/1/environ && echo
SHLVL=1 HOME=/init=/sbin/init TERM=linuxB00T_IMAGE=/boot/vmlinuz-5.15.0-70-generic drop_caps=PA
s0188249@dc-1:~$ █
```

Рисунок 21 – Окружение процесса, создающее контекст его выполнения

```
s0188249@dc-1:~$ sudo ls -l /proc/1/fd --color=always | head -30
итого 0
lrwx----- 1 root root 64 дек 16 08:21 0 -> /dev/null
lrwx----- 1 root root 64 дек 16 08:21 1 -> /dev/null
lr-x----- 1 root root 64 дек 16 10:02 10 -> /proc/1/mountinfo
lr-x----- 1 root root 64 дек 16 10:02 11 -> anon_inode:inotify
lr-x----- 1 root root 64 дек 16 10:02 117 -> pipe:[17592]
lr-x----- 1 root root 64 дек 16 10:02 12 -> /proc/swaps
lrwx----- 1 root root 64 дек 16 10:02 13 -> socket:[22922]
lr-x----- 1 root root 64 дек 16 10:02 14 -> anon_inode:inotify
lrwx----- 1 root root 64 дек 16 10:02 15 -> socket:[22931]
lrwx----- 1 root root 64 дек 16 10:02 16 -> socket:[22932]
lrwx----- 1 root root 64 дек 16 10:02 17 -> socket:[17569]
lrwx----- 1 root root 64 дек 16 10:02 18 -> anon_inode:[timerfd]
lrwx----- 1 root root 64 дек 16 10:02 19 -> socket:[23724]
lrwx----- 1 root root 64 дек 16 08:21 2 -> /dev/null
lrwx----- 1 root root 64 дек 16 10:02 20 -> socket:[23725]
lr-x----- 1 root root 64 дек 16 10:02 21 -> anon_inode:inotify
lr-x----- 1 root root 64 дек 16 10:02 22 -> anon_inode:inotify
lr-x----- 1 root root 64 дек 16 10:02 23 -> anon_inode:inotify
lrwx----- 1 root root 64 дек 16 10:02 24 -> socket:[23731]
lrwx----- 1 root root 64 дек 16 10:02 25 -> socket:[17565]
lrwx----- 1 root root 64 дек 16 10:02 26 -> socket:[17566]
lrwx----- 1 root root 64 дек 16 10:02 27 -> socket:[17567]
lrwx----- 1 root root 64 дек 16 10:02 28 -> socket:[17568]
lrwx----- 1 root root 64 дек 16 10:02 29 -> socket:[17581]
l-wx----- 1 root root 64 дек 16 08:21 3 -> /dev/kmsg
lrwx----- 1 root root 64 дек 16 10:02 30 -> socket:[17582]
lrwx----- 1 root root 64 дек 16 10:02 31 -> socket:[17580]
lrwx----- 1 root root 64 дек 16 10:02 32 -> socket:[17577]
lr-x----- 1 root root 64 дек 16 10:02 33 -> /dev/autofs
s0188249@dc-1:~$ █
```

Рисунок 22 – Дескрипторы открытых файлов

```
s0188249@dc-1:~$ sudo cat /proc/1/io
rchar: 253407658
wchar: 251083623
syscr: 263253
syscw: 117977
read_bytes: 153065472
write_bytes: 13721600
cancelled_write_bytes: 1961984
s0188249@dc-1:~$
```

Рисунок 23 – Сведения об объемах данных, прочитанных и записанных процессом в хранилище информации

| Limit | Soft Limit | Hard Limit | Units |
|-----------------------|------------|------------|-----------|
| Max cpu time | unlimited | unlimited | seconds |
| Max file size | unlimited | unlimited | bytes |
| Max data size | unlimited | unlimited | bytes |
| Max stack size | 8388608 | unlimited | bytes |
| Max core file size | 0 | unlimited | bytes |
| Max resident set | unlimited | unlimited | bytes |
| Max processes | 7527 | 7527 | processes |
| Max open files | 1048576 | 1048576 | files |
| Max locked memory | 65536 | 65536 | bytes |
| Max address space | unlimited | unlimited | bytes |
| Max file locks | unlimited | unlimited | locks |
| Max pending signals | 7527 | 7527 | signals |
| Max msgqueue size | 819200 | 819200 | bytes |
| Max nice priority | 0 | 0 | |
| Max realtime priority | 0 | 0 | |
| Max realtime timeout | unlimited | unlimited | us |

Рисунок 24 – Ограничения процесса, установленные конфигурационным файлом

```
s0188249@dc-1:~$ sudo cat /proc/1/maps | head
5e0a4e4a1000-5e0a4e58f000 r-xp 00000000 08:01 1048998
5e0a4e58f000-5e0a4e5b4000 r--p 000ed000 08:01 1048998
5e0a4e5b4000-5e0a4e5b5000 rw-p 00112000 08:01 1048998
5e0a4e77a000-5e0a4e926000 rw-p 00000000 00:00 0
70bc24000000-70bc24029000 rw-p 00000000 00:00 0
70bc24029000-70bc28000000 ---p 00000000 00:00 0
70bc29435000-70bc29436000 ---p 00000000 00:00 0
70bc29436000-70bc29c36000 rw-p 00000000 00:00 0
70bc29c36000-70bc29c37000 ---p 00000000 00:00 0
70bc29c37000-70bc2a437000 rw-p 00000000 00:00 0
s0188249@dc-1:~$
```

Рисунок 25 – Физические адреса страниц памяти, используемые в данный момент

```
s0188249@dc-1:~$ sudo cat /proc/1/sched
systemd (1, #threads: 1)
-----
se.exec_start : 6436367.596786
se.vruntime : 392.315456
se.sum_exec_runtime : 830.263360
se.nr_migrations : 159
nr_switches : 3122
nr_voluntary_switches : 1225
nr_involuntary_switches : 1897
se.load.weight : 1048576
se.avg.load_sum : 244
se.avg.runnable_sum : 249856
se.avg.util_sum : 249856
se.avg.load_avg : 0
se.avg.runnable_avg : 0
se.avg.util_avg : 0
se.avg.last_update_time : 6436367596544
se.avg.util_est.ewma : 18
se.avg.util_est.enqueued : 0
uclamp.min : 0
uclamp.max : 1024
effective_uclamp.min : 0
effective_uclamp.max : 1024
policy : 0
prio : 120
clock-delta : 80
mm->numa_scan_seq : 0
numa_pages_migrated : 0
numa_preferred_nid : -1
total_numa_faults : 0
current_node=0, numa_group_id=0
numa_faults node=0 task_private=0 task_shared=0 group_private=0 group_shared=0
s0188249@dc-1:~$
```

Рисунок 26 – Текущие значения переменных планировщика процессов

```
s0188249@dc-1:~$ sudo cat /proc/1/stat
1 (systemd) S 0 1 1 0 -1 4194560 7461 1751514 126 9078 20 62 1653 719 20 0 1 0 7 149454848
0 671173123 4096 1260 1 0 0 17 0 0 0 0 0 103398357139072 103398357287240 103398359146496
s0188249@dc-1:~$
```

Рисунок 27 – Основные сведения о процессе в машиночитаемом формате

```
s0188249@dc-1:~$ sudo cat /proc/1/status | head -30
Name:    systemd
Umask:   0000
State:   S (sleeping)
Tgid:    1
Ngid:    0
Pid:     1
PPid:   0
TracerPid:      0
Uid:     0      0      0      0
Gid:     0      0      0      0
FDSize: 128
Groups:
NStgid: 1
NSpid:  1
NSpgid: 1
NSsid:  1
VmPeak:  210832 kB
VmSize:  145952 kB
VmLck:    0 kB
VmPin:    0 kB
VmHWM:   9196 kB
VmRSS:   8936 kB
RssAnon:      2608 kB
RssFile:      6328 kB
RssShmem:     0 kB
VmData:   18576 kB
VmStk:    132 kB
VmExe:    952 kB
VmLib:   10692 kB
VmPTE:   164 kB
```

Рисунок 28 – Основные сведения о процессе в человекочитаемом формате

```
s0188249@dc-1:~$ cat /proc/1/statm
36488 2234 1582 238 0 4677 0
s0188249@dc-1:~$ █
```

Рисунок 29 – Статистика по использованию памяти

Содержимое /proc

```
s0188249@dc-1:~$ ls /proc
 1    108   13   179    19855   2     25   310   498   614   779   86   911   94
10    109   132   18    19857   20    26   32   499   619   8    87   914   959
100   11    14   180    19860  20014  262   33   5    620   80   88   915   96
101   112   15   181    19880  20017  27    4    505   622   81   883   917   97
102   113   16   182    19891  20102  28    413   519   624   82   895   922   971
103   1158  174  183    19892  21    29   477   530   706   83   90    923   975
1044  12    175  185    19906  22    297   479   569   724   84   905   924   98
1049  121   176  19    19907  224    3    480   580   732   85   907   927   99
105   126   177  19515  19908  225    30   481   583   772   857   908   93    acpi
1060  127   178  19516  19915  24    31   496   6    773   858   909   939   asound
s0188249@dc-1:~$
```

Рисунок 30 – Содержимое каталога /proc

```
s0188249@dc-1:~$ cat /proc/cmdline
BOOT_IMAGE=/boot/vmlinuz-5.15.0-70-generic root=UUID=4c91d2c3-b4bd-4cb4-9ac9-50590f54ea93 ro quiet net.ifnames=0
s0188249@dc-1:~$
```

Рисунок 31 – Список параметров, которые были переданы ядру при загрузке

```
s0188249@dc-1:~$ cat /proc/cpuinfo | head -30
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 msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush
rep_good nopl nonstop_tsc cpuid extd_apicid tsc_known_freq pnpi pclmulqdq ssse3 fma cx16 sse4_1 ss
p_legacy cr8_legacy abm sse4a misalignsse 3dnowprefetch ssbd vmmcall fsgsbase bmi1 avx2 bmi2 rdse
bugs             : fxsave_leak sysret_ss_attrs null_seg spectre_v1 spectre_v2 retbleed
bogomips        : 4192.12
TLB size         : 2560 4K pages
clflush size     : 64
cache_alignment  : 64
address sizes    : 43 bits physical, 48 bits virtual
power management:

processor      : 1
vendor_id      : AuthenticAMD
s0188249@dc-1:~$
```

Рисунок 32 – Сведения о всех установленных процессорах

```
s0188249@dc-1:~$ cat /proc/diskstats
 7      0 loop0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 7      1 loop1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 7      2 loop2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 7      3 loop3 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 7      4 loop4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 7      5 loop5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 7      6 loop6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 7      7 loop7 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 8      0 sda 14413 6705 832521 162706 5381 10307 197288 182501 0 226140 491716 0 0 0 0 1958 146508
 8      1 sda1 14269 6705 823450 161357 5381 10307 197288 182501 0 226024 343859 0 0 0 0 0 0
 8      2 sda2 2 0 4 12 0 0 0 0 0 20 12 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 8      5 sda5 50 0 4864 716 0 0 0 0 560 716 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
11      0 sr0 8 0 3 1 0 0 0 0 0 0 20 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
s0188249@dc-1:~$
```

Рисунок 33 – Статистика операций со всеми дисками

Файл /proc/meminfo – отображение информации о состоянии памяти.

Предоставляет больше параметров, чем утилита free.

```
s0188249@dc-1:~$ free
              total        used        free      shared  buff/cache   available
Mem:       2023596      295148     1389144          0      13660      339304     1568324
Swap:      2095100          0     2095100
s0188249@dc-1:~$
```

Рисунок 34 – Информация, предоставленная утилитой free

```
s0188249@dc-1:~$ cat /proc/meminfo | head -30
MemTotal:           2023596 kB
MemFree:            1389144 kB
MemAvailable:       1568360 kB
Buffers:             34604 kB
Cached:              274512 kB
SwapCached:          0 kB
Active:              130784 kB
Inactive:            355256 kB
Active(anon):        1872 kB
Inactive(anon):      188712 kB
Active(file):        128912 kB
Inactive(file):      166544 kB
Unevictable:          0 kB
Mlocked:              0 kB
SwapTotal:            2095100 kB
SwapFree:             2095100 kB
Dirty:                  12 kB
Writeback:              0 kB
AnonPages:            176892 kB
Mapped:               140016 kB
Shmem:                 13660 kB
KReclaimable:         30224 kB
Slab:                  59956 kB
SReclaimable:         30224 kB
SUnreclaim:            29732 kB
KernelStack:            3352 kB
PageTables:            15708 kB
NFS_Unstable:          0 kB
Bounce:                  0 kB
WritebackTmp:          0 kB
s0188249@dc-1:~$ █
```

Рисунок 35 – Информация, предоставленная файлом /proc/meminfo

```
s0188249@dc-1:~$ cat /proc/devices | head -30
Character devices:
 1 mem
 4 /dev/vc/0
 4 tty
 4 ttys
 5 /dev/tty
 5 /dev/console
 5 /dev/ptmx
 5 ttyprintk
 6 lp
 7 vcs
10 misc
13 input
21 sg
29 fb
89 i2c
99 ppdev
108 ppp
116 alsa
128 ptm
136 pts
180 usb
189 usb_device
204 ttymAX
226 drm
237 aux
238 hidraw
239 cec
240 lirc
241 vfio
s0188249@dc-1:~$ █
```

Рисунок 36 – Перечень устройств в системе

```
s0188249@dc-1:~$ cat /proc/filesystems
nodev    sysfs
nodev    tmpfs
nodev    bdev
nodev    proc
nodev    cgroup
nodev    cgroup2
nodev    cpuset
nodev    devtmpfs
nodev    configfs
nodev    debugfs
nodev    tracefs
nodev    securityfs
nodev    sockfs
nodev    bpf
nodev    pipefs
nodev    ramfs
nodev    hugetlbfs
nodev    devpts
          ext3
          ext2
          ext4
          squashfs
          vfat
nodev    ecryptfs
          fuseblk
nodev    fuse
nodev    fusectl
nodev    mqueue
nodev    pstore
nodev    autofs
s0188249@dc-1:~$
```

Рисунок 37 – Перечень файловых систем, поддерживаемых ядром ОС

```
s0188249@dc-1:~$ cat /proc/mounts | head -30
sysfs /sys sysfs rw,nosuid,nodev,noexec,relatime 0 0
proc /proc proc rw,nosuid,nodev,noexec,relatime 0 0
udev /dev devtmpfs rw,nosuid,relatime,size=963592k,nr_inodes=240898,mode=755,inode64 0 0
devpts /dev/pts devpts rw,nosuid,noexec,relatime,gid=5,mode=620,ptmxmode=000 0 0
tmpfs /run tmpfs rw,nosuid,noexec,relatime,size=202360k,mode=755,inode64 0 0
/dev/sda1 / ext4 rw,relatime,errors=remount-ro 0 0
securityfs /sys/kernel/security securityfs rw,nosuid,nodev,noexec,relatime 0 0
tmpfs /dev/shm tmpfs rw,nosuid,nodev,inode64 0 0
tmpfs /run/lock tmpfs rw,nosuid,nodev,noexec,relatime,size=5120k,inode64 0 0
tmpfs /sys/fs/cgroup tmpfs ro,nosuid,nodev,noexec,mode=755,inode64 0 0
cgroup /sys/fs/cgroup/systemd cgroup rw,nosuid,nodev,noexec,relatime,xattr,release_agent=/lib/
pstore /sys/fs/pstore pstore rw,nosuid,nodev,noexec,relatime 0 0
cgroup /sys/fs/cgroup/cpu cpuacct cgroup rw,nosuid,nodev,noexec,relatime,cpu,cpuacct 0 0
cgroup /sys/fs/cgroup/memory cgroup rw,nosuid,nodev,noexec,relatime,memory 0 0
cgroup /sys/fs/cgroup/hugetlb cgroup rw,nosuid,nodev,noexec,relatime,hugetlb 0 0
cgroup /sys/fs/cgroup/blkio cgroup rw,nosuid,nodev,noexec,relatime,blkio 0 0
cgroup /sys/fs/cgroup/rdma cgroup rw,nosuid,nodev,noexec,relatime,rdma 0 0
cgroup /sys/fs/cgroup/pids cgroup rw,nosuid,nodev,noexec,relatime,pids 0 0
cgroup /sys/fs/cgroup/perf_event cgroup rw,nosuid,nodev,noexec,relatime,perf_event 0 0
cgroup /sys/fs/cgroup/net_cls,net_prio cgroup rw,nosuid,nodev,noexec,relatime,net_cls,net_prio
cgroup /sys/fs/cgroup/devices cgroup rw,nosuid,nodev,noexec,relatime,devices 0 0
cgroup /sys/fs/cgroup/cpuset cgroup rw,nosuid,nodev,noexec,relatime,cpuset 0 0
cgroup /sys/fs/cgroup/misc cgroup rw,nosuid,nodev,noexec,relatime,misc 0 0
cgroup /sys/fs/cgroup/freezer cgroup rw,nosuid,nodev,noexec,relatime,freezer 0 0
systemd-1 /proc/sys/fs/binfmt_misc autofs rw,relatime,fd=36,pgrp=1,timeout=0,minproto=5,maxproto=4,mqueue=mqueue rw,relatime 0 0
hugetlbfss /dev/hugepages hugetlbfss rw,relatime,pagesize=2M 0 0
debugfs /sys/kernel/debug debugfs rw,relatime 0 0
fusectl /sys/fs/fuse/connections fusectl rw,relatime 0 0
configfs /sys/kernel/config configfs rw,relatime 0 0
s0188249@dc-1:~$
```

Рисунок 38 – Перечень смонтированных файловых систем

```
s0188249@dc-1:~$ cat /proc/modules | head -30
bluetooth 684032 2 - Live 0x0000000000000000
ecdh_generic 16384 1 bluetooth, Live 0x0000000000000000
ecc 40960 1 ecdh_generic, Live 0x0000000000000000
snd_intel8x0 53248 2 - Live 0x0000000000000000
snd_ac97_codec 163840 1 snd_intel8x0, Live 0x0000000000000000
intel_rapl_msr 20480 0 - Live 0x0000000000000000
ac97_bus 16384 1 snd_ac97_codec, Live 0x0000000000000000
intel_rapl_common 32768 1 intel_rapl_msr, Live 0x0000000000000000
snd_pcm 139264 2 snd_intel8x0,snd_ac97_codec, Live 0x0000000000000000
snd_timer 40960 1 snd_pcm, Live 0x0000000000000000
snd 106496 8 snd_intel8x0,snd_ac97_codec,snd_pcm,snd_timer, Live 0x0000000000000000
crc10dif_pcldmul 16384 1 - Live 0x0000000000000000
crc32_pcldmul 16384 0 - Live 0x0000000000000000
ghash_clmulni_intel 16384 0 - Live 0x0000000000000000
soundcore 16384 1 snd, Live 0x0000000000000000
joydev 32768 0 - Live 0x0000000000000000
aesni_intel 376832 0 - Live 0x0000000000000000
crypto_simd 16384 1 aesni_intel, Live 0x0000000000000000
input_leds 16384 0 - Live 0x0000000000000000
cryptd 24576 2 ghash_clmulni_intel,crypto_simd, Live 0x0000000000000000
serio_raw 20480 0 - Live 0x0000000000000000
vboxguest 45056 0 - Live 0x0000000000000000
mac_hid 16384 0 - Live 0x0000000000000000
vhci_hcd 53248 0 - Live 0x0000000000000000
usbip_host 45056 0 - Live 0x0000000000000000
usbip_core 36864 2 vhci_hcd,usbip_host, Live 0x0000000000000000
parport_pc 53248 0 - Live 0x0000000000000000
ppdev 24576 0 - Live 0x0000000000000000
lp 36864 0 - Live 0x0000000000000000
parport 69632 3 parport_pc,ppdev,lp, Live 0x0000000000000000
s0188249@dc-1:~$ █
```

Рисунок 39 – Список подгруженных модулей ядра

```
s0188249@dc-1:~$ cat /proc/swaps
Filename                                Type  Size   Used  Priority
/dev/sda5                                partition 2095100      0       -2
s0188249@dc-1:~$ █
```

Рисунок 40 – Список разделов подкачки

```
s0188249@dc-1:~$ cat /proc/version
Linux version 5.15.0-70-generic (builder@build) (gcc (Debian 6.3.0-18+deb9u1) 6.3.0 2017
57:55 UTC 2023
s0188249@dc-1:~$ █
```

Рисунок 41 – Версия ядра ОС

Каталог `/sys/kernel/` содержит набор файлов, которые позволяют нам оперативно без перезагрузки изменять параметры ядра ОС:

```
s0188249@dc-1:~$ ls /proc/sys/kernel | head -30
acct
acpi_video_flags
auto_msgmni
bootloader_type
bootloader_version
bpf_stats_enabled
cad_pid
cap_last_cap
core_pattern
core_pipe_limit
core_uses_pid
ctrl-alt-del
dmesg_restrict
domainname
firmware_config
ftrace_dump_on_oops
ftrace_enabled
hardlockup_all_cpu_backtrace
hardlockup_panic
hostname
hotplug
hung_task_all_cpu_backtrace
hung_task_check_count
hung_task_check_interval_secs
hung_task_panic
hung_task_timeout_secs
hung_task_warnings
io_delay_type
kexec_load_disabled
keys
s0188249@dc-1:~$ █
```

Рисунок 42 – Содержимое каталога /sys/kernel/

Управление процессами

Для управления процессами в Linux существует набор утилит. Рассмотрим работу с основными из них: консольными утилитами (ps, top и htop, kill):

```
s0188249@dc-1:~$ ps
  PID TTY          TIME CMD
 1108 pts/0    00:00:00 bash
 1142 pts/0    00:00:00 ps
s0188249@dc-1:~$ █
```

Рисунок 43 – Просмотр процессов через утилиту ps

```
s0188249@dc-1:~$ ps aux --sort=%mem | tail -n 3
s0188249 1105 0.3 2.8 577176 58584 ? Ssl 16:47 0:01 fly-term
s0188249 948 0.0 3.0 698436 61992 ? Ssl 16:47 0:00 nm-applet
fly-dm 617 0.5 4.3 399480 87188 tty7 Ssl+ 16:47 0:01 /usr/lib/xorg/Xorg
t0 -auth /var/run/xauth/A:0-bMaikb
s0188249@dc-1:~$ █
```

Рисунок 44 – Сортировка процессов

```
s0188249@dc-1:~$ ps -eo euser,ruser,suser,fuser,f,comm,label | head -30
EUSER RUSER SUSER FUSER F COMMAND LABEL
root root root root 4 systemd -
root root root root 1 kthreadd -
root root root root 1 rcu_gp -
root root root root 1 rcu_par_gp -
root root root root 1 slub_flushwq -
root root root root 1 netns -
root root root root 1 kworker/0:0H-ev -
root root root root 1 mm_percpu_wq -
root root root root 1 rcu_tasks_rude_ -
root root root root 1 rcu_tasks_trace -
root root root root 1 ksoftirqd/0 -
root root root root 1 rcu_sched -
root root root root 1 migration/0 -
root root root root 1 idle_inject/0 -
root root root root 1 kworker/0:1-eve -
root root root root 1 cpuhp/0 -
root root root root 1 cpuhp/1 -
root root root root 1 idle_inject/1 -
root root root root 1 migration/1 -
root root root root 1 ksoftirqd/1 -
root root root root 1 kworker/1:0H-kb -
root root root root 5 kdevtmpfs -
root root root root 1 inet_frag_wq -
root root root root 1 kauditfd -
root root root root 1 khungtaskd -
root root root root 1 oom_reaper -
root root root root 1 writeback -
root root root root 1 kcompactd0 -
root root root root 1 ksmd -
s0188249@dc-1:~$ █
```

Рисунок 45 – Информация об атрибутах EUID, RUID, SUID.

```
s0188249@dc-1:~$ ps -elf | head -30
F S UID      PID  PPID  C PRI  NI ADDR SZ WCHAN  STIME TTY          TIME CMD
4 S root      1    0  80  0 - 36492 -        16:46 ?          00:00:00 /sbin/init
1 S root      2    0  80  0 -     0 -        16:46 ?          00:00:00 [kthreadd]
1 I root      3    2  60 -20 -     0 -        16:46 ?          00:00:00 [rcu_gp]
1 I root      4    2  60 -20 -     0 -        16:46 ?          00:00:00 [rcu_par_gp]
1 I root      5    2  60 -20 -     0 -        16:46 ?          00:00:00 [slub_flushwq]
1 I root      6    2  60 -20 -     0 -        16:46 ?          00:00:00 [netns]
1 I root      8    2  60 -20 -     0 -        16:46 ?          00:00:00 [kworker/0:0H-ev]
1 I root     10   2  60 -20 -     0 -        16:46 ?          00:00:00 [mm_percpu_wq]
1 S root     11   2  80  0 -     0 -        16:46 ?          00:00:00 [rcu_tasks_rude_]
1 S root     12   2  80  0 -     0 -        16:46 ?          00:00:00 [rcu_tasks_trace]
1 S root     13   2  80  0 -     0 -        16:46 ?          00:00:00 [ksoftirqd/0]
1 I root     14   2  80  0 -     0 -        16:46 ?          00:00:00 [rcu_sched]
1 S root     15   2  40 - -     0 -        16:46 ?          00:00:00 [migration/0]
1 S root     16   2  9  - -     0 -        16:46 ?          00:00:00 [idle_inject/0]
1 I root     17   2  80  0 -     0 -        16:46 ?          00:00:00 [kworker/0:1-eve]
1 S root     18   2  80  0 -     0 -        16:46 ?          00:00:00 [cpuhp/0]
1 S root     19   2  80  0 -     0 -        16:46 ?          00:00:00 [cpuhp/1]
1 S root     20   2  9  - -     0 -        16:46 ?          00:00:00 [idle_inject/1]
1 S root     21   2  40 - -     0 -        16:46 ?          00:00:00 [migration/1]
1 S root     22   2  80  0 -     0 -        16:46 ?          00:00:00 [ksoftirqd/1]
1 I root     24   2  60 -20 -     0 -        16:46 ?          00:00:00 [kworker/1:0H-kb]
5 S root     25   2  80  0 -     0 -        16:46 ?          00:00:00 [kdevtmpfs]
1 I root     26   2  60 -20 -     0 -        16:46 ?          00:00:00 [inet_frag_wq]
1 S root     27   2  80  0 -     0 -        16:46 ?          00:00:00 [kauditfd]
1 S root     28   2  80  0 -     0 -        16:46 ?          00:00:00 [khungtaskd]
1 S root     29   2  80  0 -     0 -        16:46 ?          00:00:00 [oom_reaper]
1 I root     30   2  60 -20 -     0 -        16:46 ?          00:00:00 [writeback]
1 S root     31   2  80  0 -     0 -        16:46 ?          00:00:00 [kcompactd0]
1 S root     32   2  85  5 -     0 -        16:46 ?          00:00:00 [ksmd]
s0188249@dc-1:~$
```

Рисунок 46 – Просмотр потоков с помощью команды ps -eLf

```
s0188249@dc-1:~$ ps axmu | head -20
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root      1  0.1  0.4 145968  8920 ?          - 16:46  0:00 /sbin/init
root      -  0.1   -   - - -           Ss 16:46  0:00 -
root      2  0.0  0.0     0    0 ?          - 16:46  0:00 [kthreadd]
root      -  0.0   -   - - -           S 16:46  0:00 -
root      3  0.0  0.0     0    0 ?          - 16:46  0:00 [rcu_gp]
root      -  0.0   -   - - -           I< 16:46  0:00 -
root      4  0.0  0.0     0    0 ?          - 16:46  0:00 [rcu_par_gp]
root      -  0.0   -   - - -           I< 16:46  0:00 -
root      5  0.0  0.0     0    0 ?          - 16:46  0:00 [slub_flushwq]
root      -  0.0   -   - - -           I< 16:46  0:00 -
root      6  0.0  0.0     0    0 ?          - 16:46  0:00 [netns]
root      -  0.0   -   - - -           I< 16:46  0:00 -
root      8  0.0  0.0     0    0 ?          - 16:46  0:00 [kworker/0:0H-ev]
root      -  0.0   -   - - -           I< 16:46  0:00 -
root     10  0.0  0.0     0    0 ?          - 16:46  0:00 [mm_percpu_wq]
root      -  0.0   -   - - -           I< 16:46  0:00 -
root     11  0.0  0.0     0    0 ?          - 16:46  0:00 [rcu_tasks_rude_]
root      -  0.0   -   - - -           S 16:46  0:00 -
root     12  0.0  0.0     0    0 ?          - 16:46  0:00 [rcu_tasks_trace]
s0188249@dc-1:~$
```

Рисунок 47 – Просмотр потоков с помощью команды ps axmu

```
s0188249@dc-1:~$ pstree | head -30
systemd---NetworkManager---dhclient---{dhclient}
|           |           |-{gdbus}
|           |           `-{gmain}
|-acpid
-agetty
-astra-orientati---{QDBusConnection}
`-{XCBEventListene}
-at-spi2-registr---{gdbus}
`-{gmain}
-avahi-daemon---avahi-daemon
-cron
-cupsd
-2*[dbus-daemon]
-dbus-launch
-fly-dm---Xorg---{InputThread}
|           |-2*[{Xorg}]
|           |-{llvmpipe-0}
|           `-{llvmpipe-1}
`-fly-dm---fly-wm---at-spi-bus-laun---dbus-daemon
|           |           |-{dconf worker}
|           |           |-{gdbus}
|           |           `-{gmain}
|           |-fly-brightness---{QDBusConnection}
|           `-{QXcbEventReader}
|           |-fly-cups-watch---{QDBusConnection}
|           |-fly-reflex-serv---{QDBusConnection}
|           `-{QXcbEventReader}
|           |-fly-search-pane---{QDBusConnection}
|           `-{QXcbEventReader}
`-fly-term---bash---head
s0188249@dc-1:~$
```

Рисунок 48 – Дерево процессов

```
s0188249@dc-1:~$ pstree -p | head -30
systemd(1)---NetworkManager(518)---dhclient(631)---{dhclient}(640)
|           |           |-{gdbus}(558)
|           |           `-{gmain}(556)
|-acpid(515)
-agetty(583)
-astra-orientati(764)---{QDBusConnection}(765)
`-{XCBEventListene}(814)
-at-spi2-registr(1065)---{gdbus}(1069)
`-{gmain}(1068)
-avahi-daemon(472)---avahi-daemon(477)
-cron(473)
-cupsd(1196)
-dbus-daemon(475)
-dbus-daemon(897)
-dbus-launch(896)
-fly-dm(589)---Xorg(617)---{InputThread}(745)
|           |-{Xorg}(671)
|           |-{Xorg}(672)
|           |-{llvmpipe-0}(669)
|           `-{llvmpipe-1}(670)
`-fly-dm(746)---fly-wm(818)---at-spi-bus-laun(957)---dbus-daemon(995)
|           |           |-{dconf worker}(991)
|           |           |-{gdbus}(994)
|           |           `-{gmain}(992)
|           |-fly-brightness(963)---{QDBusConnection}(985)
|           `-{QXcbEventReader}(981)
|           |-fly-cups-watch(949)---{QDBusConnection}(972)
|           |-fly-reflex-serv(947)---{QDBusConnection}(999)
|           `-{QXcbEventReader}(997)
`-fly-search-pane(954)---{QDBusConnection}(989)
s0188249@dc-1:~$
```

Рисунок 49 – Вывод PID процессов

```
s0188249@dc-1:~$ ps -1
  PID TTY      STAT   TIME COMMAND
    1 ?        Ss      0:00 /sbin/init
s0188249@dc-1:~$ █
```

Рисунок 50 – Фильтрация вывода команды ps

```
s0188249@dc-1:~$ ps -C bash
  PID TTY      TIME CMD
 1108 pts/0    00:00:00 bash
s0188249@dc-1:~$ █
```

Рисунок 51 – Вывод информации по процессам

```
s0188249@dc-1:~$ ps -C bash -o pid
  PID
 1108
s0188249@dc-1:~$ ps -C bash -o pid=
 1108
s0188249@dc-1:~$ █
```

Рисунок 52 – Вывод только колонки с PID найденных процессов и только PID
без названия колонки

```
s0188249@dc-1:~$ ps aux | grep tty
root      583  0.0  0.0 14532 1852  tty1      Ss+ 16:47  0:00 /sbin/agetty
fly-dm    617  0.2  4.3 399480 87188  tty7      Ssl+ 16:47  0:02 /usr/lib/xo
t0 -auth /var/run/xauth/A:0-bMaikb
s0188249 1259  0.0  0.0 12788   972  pts/0      S+   17:01  0:00 grep tty
s0188249@dc-1:~$ █
```

Рисунок 53 – Дополнительная фильтрация с помощью утилиты grep