

## Практическое задание по уроку №4 «Загрузка ОС и процессы»

1. Потоки ввода/вывода. Создать файл, используя команду echo. Используя команду cat, прочитать содержимое каталога etc, ошибки перенаправить в отдельный файл.

Решение

- Создаем директорию test, переходим в нее. Внутри с помощью команды echo создаем файл errors.txt, куда будем перенаправлять ошибки.

```
root@ubuntu-server: /test
drwxr-xr-x  6 root root    4096 Sep 19 17:52 home
lrwxrwxrwx  1 root root      7 Feb  1 2021 lib -> usr/lib
lrwxrwxrwx  1 root root      9 Feb  1 2021 lib32 -> usr/lib32
lrwxrwxrwx  1 root root      9 Feb  1 2021 lib64 -> usr/lib64
lrwxrwxrwx  1 root root     10 Feb  1 2021 libx32 -> usr/libx32
drwx----- 2 root root   16384 Sep 12 06:18 lost+found
drwxr-xr-x  2 root root    4096 Feb  1 2021 media
drwxr-xr-x  2 root root    4096 Feb  1 2021 mnt
drwxr-xr-x  2 root root    4096 Feb  1 2021 opt
dr-xr-xr-x 160 root root      0 Sep 22 15:33 proc
drwx----- 9 root root    4096 Sep 20 15:34 root
drwxr-xr-x 25 root root    780 Sep 22 15:34 run
lrwxrwxrwx  1 root root      8 Feb  1 2021/sbin -> usr/sbin
drwxr-xr-x  7 root root    4096 Sep 16 18:22 snap
drwxr-xr-x  2 root root    4096 Feb  1 2021 srv
-rw-----  1 root root 1300234240 Sep 12 06:28 swap.img
dr-xr-xr-x 13 root root      0 Sep 22 15:32 sys
drwxr-xr-x  2 root root    4096 Sep 22 15:40 test
drwxrwxrwt 11 root root    4096 Sep 22 15:34 tmp
drwxr-xr-x 14 root root    4096 Feb  1 2021 usr
drwxr-xr-x 13 root root    4096 Feb  1 2021 var
root@ubuntu-server: /# cd /test/
root@ubuntu-server: /test# echo "Errors" > errors.txt
root@ubuntu-server: /test#
```

- Для записи ошибок воспользуемся стандартным потоком 2 (STDERR) и с помощью команды cat /etc/\* 2>/test/errors.txt направим информацию в файл errors.txt (2 – поток ошибок, \* - прочитать все файлы внутри каталога)

```
root@ubuntu-server: /
cat: /etc/security: Is a directory
cat: /etc/selinux: Is a directory
cat: /etc/skel: Is a directory
cat: /etc/sos: Is a directory
cat: /etc/ssh: Is a directory
cat: /etc/ssl: Is a directory
cat: /etc/sudoers.d: Is a directory
cat: /etc/sysctl.d: Is a directory
cat: /etc/systemd: Is a directory
cat: /etc/terminfo: Is a directory
cat: /etc/thermald: Is a directory
cat: /etc/tmpfiles.d: Is a directory
cat: /etc/ubuntu-advantage: Is a directory
cat: /etc/udev: Is a directory
cat: /etc/ufw: Is a directory
cat: /etc/update-manager: Is a directory
cat: /etc/update-motd.d: Is a directory
cat: /etc/update-notifier: Is a directory
cat: /etc/UPower: Is a directory
cat: /etc/vim: Is a directory
cat: /etc/vmware-tools: Is a directory
cat: /etc/X11: Is a directory
cat: /etc/xdg: Is a directory
root@ubuntu-server: /# cat /etc/* 2>/test/errors.txt
```

- После выполнения команды с помощью редактора vim можем посмотреть содержание файла errors.txt



2. Конвейре (pipeline). Использовать команду `cut` на вывод длинного списка каталога, чтобы отобразить только права доступа к файлам. Затем отправить в конвейере этот вывод на `sort` и `uniq`, чтобы отфильтровать все повторяющиеся строки.

Решение

- С помощью команды `ls -la` выводим список файлов, располагающихся в текущей директории с информацией по правам доступа в том числе.

```
root@ubuntu-server: /
cat: /etc/update-notifier: Is a directory
cat: /etc/UPower: Is a directory
cat: /etc/vim: Is a directory
cat: /etc/vmware-tools: Is a directory
cat: /etc/X11: Is a directory
cat: /etc/xdg: Is a directory
root@ubuntu-server: /# ls -la
total 1269844
drwxr-xr-x 21 root root      4096 Sep 22 15:40 .
drwxr-xr-x 21 root root      4096 Sep 22 15:40 ..
lrwxrwxrwx 1 root root         7 Feb  1  2021 bin -> usr/bin
drwxr-xr-x 4 root root      4096 Sep 12 06:35 boot
drwxr-xr-x 2 root root      4096 Sep 12 06:20 cdrom
drwxr-xr-x 19 root root     4080 Sep 22 15:33 dev
drwxr-xr-x 97 root root      4096 Sep 21 15:09 etc
drwxr-xr-x 6 root root      4096 Sep 19 17:52 home
lrwxrwxrwx 1 root root         7 Feb  1  2021 lib -> usr/lib
lrwxrwxrwx 1 root root         9 Feb  1  2021 lib32 -> usr/lib32
lrwxrwxrwx 1 root root         9 Feb  1  2021 lib64 -> usr/lib64
lrwxrwxrwx 1 root root        10 Feb  1  2021 libx32 -> usr/libx32
drwx----- 2 root root    16384 Sep 12 06:18 lost+found
drwxr-xr-x 2 root root      4096 Feb  1  2021 media
drwxr-xr-x 2 root root      4096 Feb  1  2021 mnt
drwxr-xr-x 2 root root      4096 Feb  1  2021 opt
dr-xr-xr-x 162 root root         0 Sep 22 15:33 proc
```

- Для удобства работы с данными добавим всю информацию в отдельный файл с помощью перенаправления (`>`). Используя редактор `vim` просмотрим содержимое файла. Как видно, информация записалась в полном объеме.

```
root@ubuntu-server: /
drwxr-xr-x 25 root root      800 Sep 22 16:11 run
lrwxrwxrwx 1 root root         8 Feb  1  2021 sbin -> usr/sbin
drwxr-xr-x 7 root root      4096 Sep 16 18:22 snap
drwxr-xr-x 2 root root      4096 Feb  1  2021 srv
-rw----- 1 root root 1300234240 Sep 12 06:28 swap.img
dr-xr-xr-x 13 root root         0 Sep 22 15:32 sys
drwxr-xr-x 2 root root      4096 Sep 22 15:54 test
drwxrwxrwt 11 root root      4096 Sep 22 16:11 tmp
drwxr-xr-x 14 root root      4096 Feb  1  2021 usr
drwxr-xr-x 13 root root      4096 Feb  1  2021 var
root@ubuntu-server: /# ls -la > /test/pipeline_test.txt
root@ubuntu-server: /# vim /test/pipeline_test.txt
total 1269844
drwxr-xr-x 21 root root      4096 Sep 22 15:40 .
drwxr-xr-x 21 root root      4096 Sep 22 15:40 ..
lrwxrwxrwx 1 root root         7 Feb  1  2021 bin -> usr/bin
drwxr-xr-x 4 root root      4096 Sep 12 06:35 boot
drwxr-xr-x 2 root root      4096 Sep 12 06:20 cdrom
drwxr-xr-x 19 root root     4080 Sep 22 15:33 dev
drwxr-xr-x 97 root root      4096 Sep 21 15:09 etc
drwxr-xr-x 6 root root      4096 Sep 19 17:52 home
lrwxrwxrwx 1 root root         7 Feb  1  2021 lib -> usr/lib
lrwxrwxrwx 1 root root         9 Feb  1  2021 lib32 -> usr/lib32
lrwxrwxrwx 1 root root         9 Feb  1  2021 lib64 -> usr/lib64
lrwxrwxrwx 1 root root        10 Feb  1  2021 libx32 -> usr/libx32
```

- Произведем сортировку информации: с помощью команды `cut` оставим только первые 10 символов, которые относятся к правам доступа для файла (атрибут `-b` – bytes служит для задания диапазона (1-10 прописываем вручную)). Отсортированный список записываем в файл `pipeline_test2.txt`. Используя редактор `vim` производим просмотр содержимого файла (осталась только информация по правам доступа).

```

root@ubuntu-server: /
drwxr-xr-x  2 root root      4096 Feb  1  2021 srv
-rw-----  1 root root 1300234240 Sep 12 06:28 swap.img
dr-xr-xr-x 13 root root          0 Sep 22 15:32 sys
drwxr-xr-x  2 root root      4096 Sep 22 15:54 test
drwxrwxrwt 11 root root      4096 Sep 22 16:11 tmp
drwxr-xr-x 14 root root      4096 Feb  1  2021 usr
drwxr-xr-x 13 root root      4096 Feb  1  2021 var
root@ubuntu-server:/# ls -la > /test/pipeline_test.txt
root@ubuntu-server:/# vim /test/pipeline_test.txt
root@ubuntu-server:/# cut -b 1-10 /test/pipeline_test.txt > /test/pipeline_test2
.txt
root@ubuntu-server:/# vim /test/pipeline_test2.txt
total 1269
drwxr-xr-x
drwxr-xr-x
lrwxrwxrwx
drwxr-xr-x
drwxr-xr-x
drwxr-xr-x
drwxr-xr-x
drwxr-xr-x
lrwxrwxrwx
lrwxrwxrwx
lrwxrwxrwx
lrwxrwxrwx

```

- Используем следующую команду для отправки в конвейере на `sort` (параллельно отсортированная информация записана в файл `pipeline_test3.txt`). Используя редактор `vim` убеждаемся, что информация была отсортирована.

```

root@ubuntu-server: /
drwxr-xr-x  2 root root      4096 Sep 22 15:54 test
drwxrwxrwt 11 root root      4096 Sep 22 16:11 tmp
drwxr-xr-x 14 root root      4096 Feb  1  2021 usr
drwxr-xr-x 13 root root      4096 Feb  1  2021 var
root@ubuntu-server:/# ls -la > /test/pipeline_test.txt
root@ubuntu-server:/# vim /test/pipeline_test.txt
root@ubuntu-server:/# cut -b 1-10 /test/pipeline_test.txt > /test/pipeline_test2
.txt
root@ubuntu-server:/# vim /test/pipeline_test2.txt
root@ubuntu-server:/# cut -b 1-10 /test/pipeline_test.txt | sort > /test/pipeline_test3.txt
root@ubuntu-server:/# vim /test/pipeline_test3.txt
drwx-----
drwx-----
drwxrwxrwt
drwxr-xr-x
drwxr-xr-x
drwxr-xr-x
drwxr-xr-x
drwxr-xr-x
drwxr-xr-x
drwxr-xr-x
drwxr-xr-x
drwxr-xr-x

```

- Используя команду `uniq` отфильтруем повторяющиеся строки. Параллельно произведем запись информации в файл `pipeline_test4.txt`. Используя редактор `vim` выведем результат работы команды `uniq`

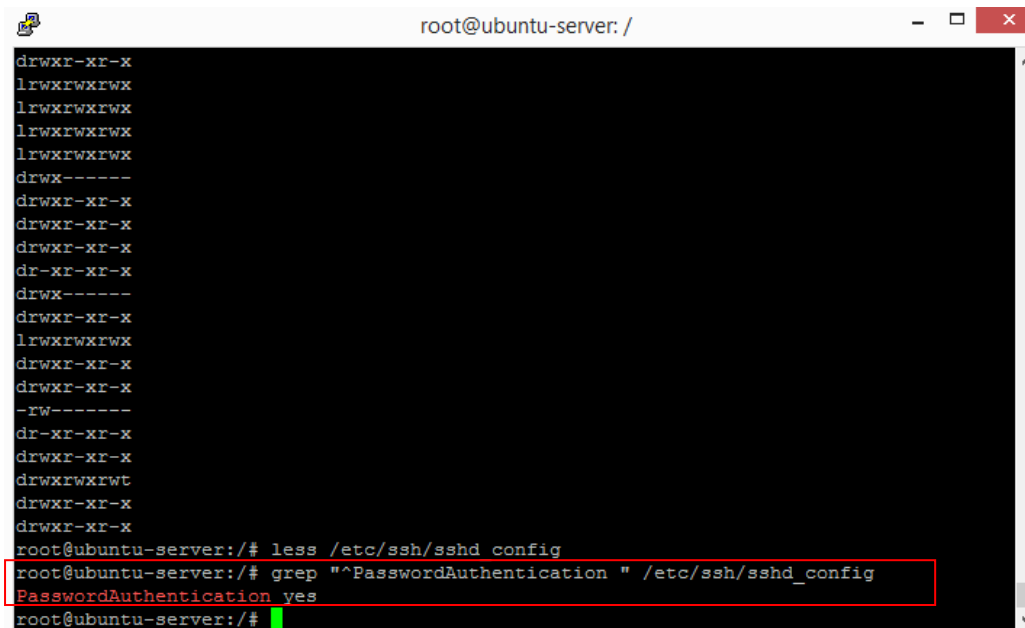
```
root@ubuntu-server: /
drwxr-xr-x 13 root root      4096 Feb  1  2021 var
root@ubuntu-server:/# ls -la > /test/pipeline_test.txt
root@ubuntu-server:/# vim /test/pipeline_test.txt
root@ubuntu-server:/# cut -b 1-10 /test/pipeline_test.txt > /test/pipeline_test2
.txt
root@ubuntu-server:/# vim /test/pipeline_test2.txt
root@ubuntu-server:/# cut -b 1-10 /test/pipeline_test.txt | sort > /test/pipeline_test3.txt
root@ubuntu-server:/# vim /test/pipeline_test3.txt
root@ubuntu-server:/# cut -b 1-10 /test/pipeline_test.txt | uniq > /test/pipeline_test4.txt
root@ubuntu-server:/# vim /test/pipeline_test4.txt
total 1269
drwxr-xr-x
lrwxrwxrwx
drwxr-xr-x
lrwxrwxrwx
drwx-----
drwxr-xr-x
dr-xr-xr-x
drwx-----
drwxr-xr-x
lrwxrwxrwx
drwxr-xr-x
-rw-----
```

**\*Примечание: значений в файле стало меньше, но почему-то все-таки осталось несколько повторяющихся (показаны стрелкой)**

3. Управление процессами. Изменить конфигурационный файл службы SSH: /etc/ssh/sshd\_config, отключив аутентификацию по паролю PasswordAuthentication no. Выполните рестарт службы systemctl restart sshd (service sshd restart), верните аутентификацию по паролю, выполните reload службы systemctl reload sshd (services sshd reload). В чём различие между действиями restart и reload? Создайте файл при помощи команды cat > file\_name, напишите текст и завершите комбинацией ctrl+d. Какой сигнал передадим процессу?

Решение

- С помощью команды grep «^PasswordAuthentication " /etc/ssh/sshd\_config проверяем включена ли на данный момент аутентификация по паролю (функция grep – поиск нужной строки в файле).



```
root@ubuntu-server: /
drwxr-xr-x
lrwxrwxrwx
lrwxrwxrwx
lrwxrwxrwx
lrwxrwxrwx
drwx-----
drwxr-xr-x
drwxr-xr-x
drwxr-xr-x
dr-xr-xr-x
drwx-----
drwxr-xr-x
lrwxrwxrwx
drwxr-xr-x
drwxr-xr-x
-rw-----
dr-xr-xr-x
drwxr-xr-x
drwxrwxrwt
drwxr-xr-x
drwxr-xr-x
root@ubuntu-server:/$ less /etc/ssh/sshd config
root@ubuntu-server:/$ grep "^PasswordAuthentication " /etc/ssh/sshd_config
PasswordAuthentication yes
root@ubuntu-server:/$
```

- С помощью команды `sed -i 's/^PasswordAuthentication yes/PasswordAuthentication no/' /etc/ssh/sshd_config` изменяем параметр аутентификации по паролю на «no» (`sed` – потоковый текстовый редактор, работающий по принципу замены; `-i` – меняем в указанном в конце файле (`ssh...`)); что ищем – между первыми двумя слэшами, что меняем – между вторыми)

```

root@ubuntu-server: /
lrwxrwxrwx
lrwxrwxrwx
lrwxrwxrwx
drwx-----
drwxr-xr-x
drwxr-xr-x
drwxr-xr-x
dr-xr-xr-x
drwx-----
drwxr-xr-x
lrwxrwxrwx
drwxr-xr-x
drwxr-xr-x
-rw-----
dr-xr-xr-x
drwxr-xr-x
drwxrwxrwt
drwxr-xr-x
drwxr-xr-x
root@ubuntu-server:/# less /etc/ssh/sshd_config
root@ubuntu-server:/# grep "^PasswordAuthentication " /etc/ssh/sshd_config
PasswordAuthentication yes
root@ubuntu-server:/# sed -i '^C
root@ubuntu-server:/# sed -i 's/^PasswordAuthentication yes/PasswordAuthentication no/' /etc/ssh/sshd_config

```

- Вновь с помощью команды `grep '^PasswordAuthentication ' /etc/ssh/sshd_config` проверяем включена ли на данный момент аутентификация по паролю. «no» – все работало

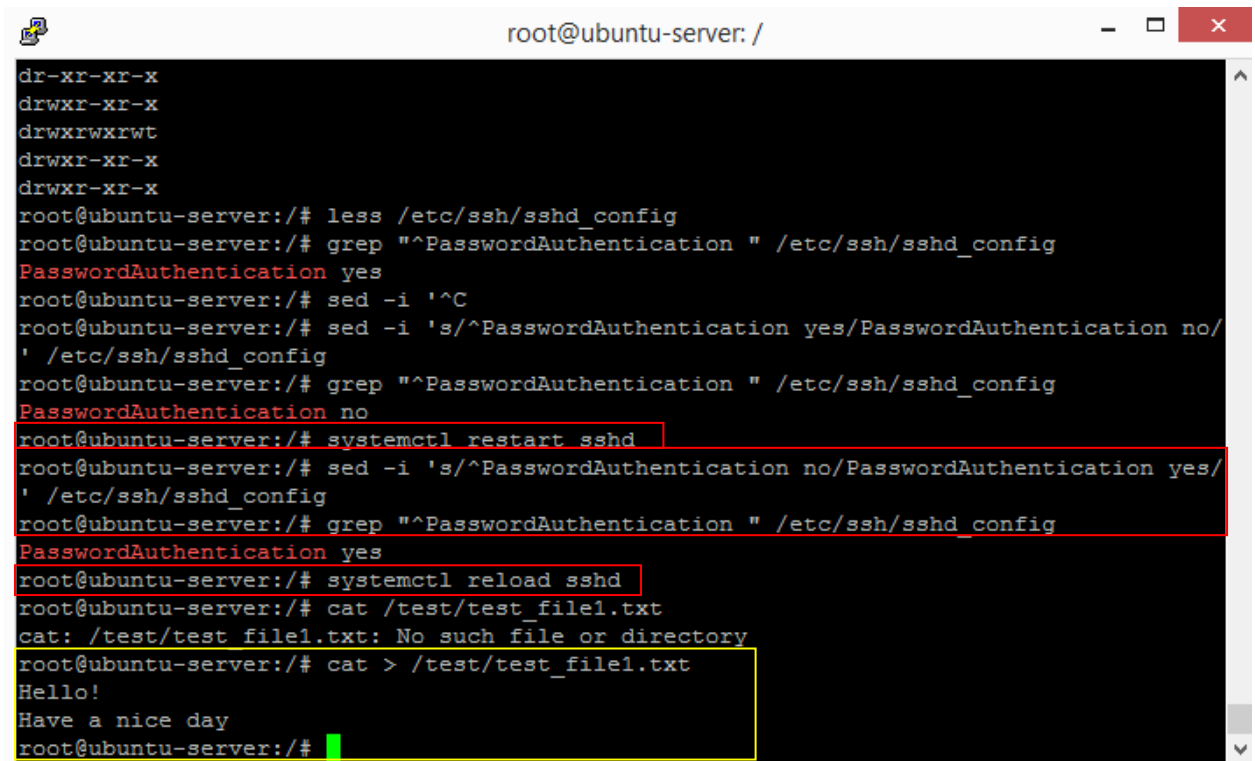
```

root@ubuntu-server: /
drwx-----
drwxr-xr-x
drwxr-xr-x
drwxr-xr-x
dr-xr-xr-x
drwx-----
drwxr-xr-x
lrwxrwxrwx
drwxr-xr-x
drwxr-xr-x
-rw-----
dr-xr-xr-x
drwxr-xr-x
drwxrwxrwt
drwxr-xr-x
drwxr-xr-x
root@ubuntu-server:/# less /etc/ssh/sshd_config
root@ubuntu-server:/# grep "^PasswordAuthentication " /etc/ssh/sshd_config
PasswordAuthentication yes
root@ubuntu-server:/# sed -i '^C
root@ubuntu-server:/# sed -i 's/^PasswordAuthentication yes/PasswordAuthentication no/' /etc/ssh/sshd_config
root@ubuntu-server:/# grep "^PasswordAuthentication " /etc/ssh/sshd_config
PasswordAuthentication no
root@ubuntu-server:/#

```



- Выполняем рестарт службы с помощью команды `systemctl restart sshd`. После этого возвращаем аутентификацию по паролю, проверяем что состояние изменилось на «yes» и выполняем reload (`systemctl reload sshd`). После с помощью команды `cat` создаем файл `test_file1.txt`, и записываем в него текст. Для выхода из режима ввода используем сочетание клавиш `ctrl + d`. Сигнал, который был передан процессу `SIGQUIT`



```
root@ubuntu-server: /
dr-xr-xr-x
drwxr-xr-x
drwxrwxrwt
drwxr-xr-x
drwxr-xr-x
root@ubuntu-server:/# less /etc/ssh/sshd_config
root@ubuntu-server:/# grep "^PasswordAuthentication " /etc/ssh/sshd_config
PasswordAuthentication yes
root@ubuntu-server:/# sed -i '^C
root@ubuntu-server:/# sed -i 's/^PasswordAuthentication yes/PasswordAuthentication no/
' /etc/ssh/sshd_config
root@ubuntu-server:/# grep "^PasswordAuthentication " /etc/ssh/sshd_config
PasswordAuthentication no
root@ubuntu-server:/# systemctl restart sshd
root@ubuntu-server:/# sed -i 's/^PasswordAuthentication no/PasswordAuthentication yes/
' /etc/ssh/sshd_config
root@ubuntu-server:/# grep "^PasswordAuthentication " /etc/ssh/sshd_config
PasswordAuthentication yes
root@ubuntu-server:/# systemctl reload sshd
root@ubuntu-server:/# cat /test/test_file1.txt
cat: /test/test_file1.txt: No such file or directory
root@ubuntu-server:/# cat > /test/test_file1.txt
Hello!
Have a nice day
root@ubuntu-server:/#
```

- Различие между `restart` и `reload`: `restart` останавливает и запускает систему снова, а `reload` просто сообщает о необходимости повторного чтения содержимого файла. `Restart` более глубокая операция, нежели `reload`. Их можно сравнить со следующим: `restart` – все равно, что достать карточку из банкомата и повторно вставить в картоприёмник для совершения операций, а `reload` – повторный заход в операцию (например, показ баланса карты) без необходимости извлечения карты.



4. Сигналы процессам. Запустите mc. Используя ps, найдите PID процесса, завершите процесс, передав ему сигнал 9 (SIGKILL)

Решение

- Запустим mc с помощью соответствующей команды.



```
root@ubuntu-server: /  
root@ubuntu-server:/# mc
```

A terminal window titled 'root@ubuntu-server: /' with standard window controls. The prompt is 'root@ubuntu-server:/#' and the command 'mc' has been entered, followed by a green cursor. The rest of the terminal area is black.

- С помощью утилиты atop (заранее устанавливаем через `sudo apt install atop`) просматриваем процессы)

```

mc [root@ubuntu-server]:/
Left      File      Command  Options  Right
<- /      .[^]>
.n        Name      Size     Modify   time
~bin      7          Feb 1    2021
/boot     4096       Sep 12   06:35
/cdrom    4096       Sep 12   06:20
/dev      4080       Sep 22   15:33
/etc      4096       Sep 21   15:09
/home     4096       Sep 19   17:52
~lib      7          Feb 1    2021
~lib32    9          Feb 1    2021
~lib64    9          Feb 1    2021
~libx32   10         Feb 1    2021
/lost+found 16384      Sep 12   06:18
/media    4096       Feb 1    2021
/mnt      4096       Feb 1    2021
/opt      4096       Feb 1    2021
/proc     0          Sep 22   15:33
/root     4096       Sep 22   16:56
/run      880        Sep 22   17:45
~sbin     8          Feb 1    2021
/snap     4096       Sep 16   18:22
/srv      4096       Feb 1    2021
/sys      0          Sep 22   15:32
/test     4096       Sep 22   17:42
/tmp      4096       Sep 22   18:26
/usr      4096       Feb 1    2021
/var      4096       Feb 1    2021
swap.img  1240M      Sep 12   06:28

-> usr/bin 3091M/6987M (44%)

Hint: Want your plain shell? Press C-o, and get back to MC with C-o again.
root@ubuntu-server:/# atop
1Help 2Menu 3View 4Edit 5Copy 6RenMov 7Mkdir 8Delete 9PullDn 10Quit
  
```

- Все процессы будут иметь следующий вид. Среди них ищем процесс mc (PID 2392)

mc [root@ubuntu-server]:/
ATOP - ubuntu-server
2021/09/22 18:45:18
-----
3h12m20s elapsed
PRC | sys 11.94s | user 11.82s | #proc 105 | #tslpu 0 | #zombie 0 | #exit 0 |
CPU | sys 0% | user 0% | irq 0% | idle 99% | wait 1% | ipc notavail |
CPL | avg1 0.00 | avg5 0.00 | avg15 0.00 | csw 1166784 | intr 531388 | numcpu 1 |
MEM | tot 981.3M | free 405.2M | cache 333.2M | buff 40.3M | slab 86.1M | hptot 0.0M |
SWP | tot 1.2G | free 1.2G | | | vmcom 312.1M | vmlim 1.7G |
PSI | cs 0/0/0 | ms 0/0/0 | mf 0/0/0 | is 0/0/1 | if 0/0/1 | |
LVM | g-ubuntu--lv | busy 1% | read 10485 | write 7760 | MBw/s 0.0 | avio 5.87 ms |
DSK | | sda | busy 1% | read 7816 | write 3581 | MBw/s 0.0 | avio 9.48 ms |
NET | transport | tcp\_i 6016 | tcp\_o 6598 | udp\_i 131 | udp\_o 177 | tcpao 10 |
NET | network | ip\_i 6452 | ip\_o 6523 | ipfrw 0 | deliv 6450 | icmpo 16 |
NET | enp0s3 0% | pcki 6570 | pcko 6770 | sp 1000 Mbps | si 0 Kbps | so 1 Kbps |
NET | lo ---- | pcki 198 | pcko 198 | sp 0 Mbps | si 0 Kbps | so 0 Kbps |
\*\*\* system and process activity since boot \*\*\*
PID VSTEXT VSLIBS VDATA VSTACK VSIZE RSIZE PSIZE VGROW RGROW SWAPSZ MEM CMD 1/5
637 14892K 1824K 90352K 132K 624.3M 28868K 0K 624.3M 28868K 0K 3% snappd
689 2784K 11064K 16088K 132K 105.4M 20920K 0K 105.4M 20920K 0K 2% unattended-upg
633 2784K 7972K 7616K 132K 29076K 18256K 0K 29076K 18256K 0K 2% networkd-dispa
526 96K 5352K 10172K 132K 273.7M 18116K 0K 273.7M 18116K 0K 2% multipathd
344 120K 8908K 19240K 132K 51684K 14924K 0K 51684K 14924K 0K 1% systemd-journa
1 960K 8768K 19628K 1036K 101.1M 13076K 0K 101.1M 13076K 0K 1% systemd
611 296K 11308K 3640K 132K 24092K 12228K 0K 24092K 12228K 0K 1% systemd-resolv
943 960K 8768K 1140K 132K 18692K 9880K 0K 18692K 9880K 0K 1% systemd
964 552K 7144K 1248K 276K 13928K 8944K 0K 13928K 8944K 0K 1% sshd
1840 208K 2576K 4064K 708K 9180K 8540K 0K 9180K 8540K 0K 1% atop
2392 780K 5148K 2984K 132K 17760K 7928K 0K 17760K 7928K 0K 1% mc
647 184K 8916K 840K 132K 17040K 7864K 0K 17040K 7864K 0K 1% systemd-logind
609 1532K 8704K 8720K 132K 26800K 7772K 0K 26800K 7772K 0K 1% systemd-networ
2402 208K 2576K 3412K 132K 7952K 7732K 0K 7952K 7732K 0K 1% atop
623 140K 6648K 25736K 132K 230.1M 7388K 0K 230.1M 7388K 0K 1% accounts-daemo
729 88K 6772K 25624K 132K 227.3M 6988K 0K 227.3M 6988K 0K 1% polkitd
1592 552K 6564K 676K 132K 12176K 6908K 0K 12176K 6908K 0K 1% sshd
568 36K 9028K 8892K 132K 90424K 6492K 0K 90424K 6492K 0K 1% systemd-timesy
375 452K 4228K 1900K 132K 22048K 6008K 0K 22048K 6008K 0K 1% systemd-udev
1043 552K 7144K 1380K 276K 14060K 5896K 0K 14060K 5896K 0K 1% sshd
1056 888K 1564K 2000K 132K 8672K 5728K 0K 8672K 5728K 0K 1% bash
950 888K 1564K 1592K 132K 8264K 5204K 0K 8264K 5204K 0K 1% bash
1044 888K 1564K 1604K 132K 8276K 5192K 0K 8276K 5192K 0K 1% bash
634 468K 3580K 18352K 132K 219.1M 4732K 0K 219.1M 4732K 0K 0% rsyslogd
627 180K 4328K 776K 132K 7492K 4676K 0K 7492K 4676K 0K 0% dbus-daemon

- Для завершения процесса по сигналу 9 (SIGKILL) вводим команду `kill -9 2392`

mc [root@ubuntu-server]:/

Left	File	Command	Options	Right
<- /				<- /
.n	Name	Size	Modify time	.n
~bin		7	Feb 1 2021	~bin
/boot		4096	Sep 12 06:35	/boot
/cdrom		4096	Sep 12 06:20	/cdrom
/dev		4080	Sep 22 15:33	/dev
/etc		4096	Sep 21 15:09	/etc
/home		4096	Sep 19 17:52	/home
~lib		7	Feb 1 2021	~lib
~lib32		9	Feb 1 2021	~lib32
~lib64		9	Feb 1 2021	~lib64
~libx32		10	Feb 1 2021	~libx32
/lost+found		16384	Sep 12 06:18	/lost+found
/media		4096	Feb 1 2021	/media
/mnt		4096	Feb 1 2021	/mnt
/opt		4096	Feb 1 2021	/opt
/proc		0	Sep 22 15:33	/proc
/root		4096	Sep 22 16:56	/root
/run		880	Sep 22 17:45	/run
~sbin		8	Feb 1 2021	~sbin
/snap		4096	Sep 16 18:22	/snap
/srv		4096	Feb 1 2021	/srv
/sys		0	Sep 22 15:32	/sys
/test		4096	Sep 22 17:42	/test
/tmp		4096	Sep 22 18:26	/tmp
/usr		4096	Feb 1 2021	/usr
/var		4096	Feb 1 2021	/var
swap.img		1240M	Sep 12 06:28	swap.img

-> usr/bin 3091M/6987M (44%)

Hint: Want your plain shell? Press C-o, and get back to MC with C-o again.

root@ubuntu-server:/# kill -9 2392

1Help 2Menu 3View 4Edit 5Copy 6RenMov 7Mkdir 8Delete 9PullDn 10Quit

- Процесс завершен (выйдет сообщение "Killed")



A terminal window titled "root@ubuntu-server: /" with standard window controls. The terminal shows the following sequence of commands and output:

```
root@ubuntu-server:/# mc
root@ubuntu-server:/# atop
root@ubuntu-server:/# Killed
root@ubuntu-server:/# █
```

The prompt "root@ubuntu-server:/#" is followed by "mc", then "atop", then "Killed", and finally a green cursor block "█".