

Лабортная номер 6

Поиск файлов. Перенаправление ввода-вывода. Просмотр запущенных процессов

Осокин Г.И. Нммбд-02-22

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
2.1	Осуществим вход	6
2.2	Впишем файлы в file.txt	6
2.3	Выведем имена всех файлов с расширением .conf	7
2.4	Определим, какие файлы начинаются с с	8
2.5	Выведем имена файлов из каталога /etc	9
2.6	Запустим фоновый просцц	9
2.7	Удалим ~/logfile	9
2.8	Запустим geidt в фоновом режиме	10
2.9	Определим Id процесса	10
2.10	УБЬЕМ gedit	10
2.11	df и du	11
2.12	Выведем все директории домашнего каталога	11
3	Ответы на контрольные вопросы	13
3.1	Какие потоки ввода вывода вы знаете?	13
3.2	Объясните разницу между операцией > и ».	13
3.3	Что такое конвейер?	13
3.4	Что такое процесс? Чем это понятие отличается от программы?	13
3.5	Что такое PID и GID?	14
3.6	Что такое задачи и какая команда позволяет ими управлять?	14
3.7	Найдите информацию об утилитах top и htop. Каковы их функции?	15
3.8	Назовите и дайте характеристику команде поиска файлов. Приведите примеры использования этой команды	15
3.9	Можно ли по контексту (содержанию) найти файл? Если да, то как?	15
3.10	Как определить объем свободной памяти на жёстком диске?	16
3.11	Как определить объем вашего домашнего каталога?	16
4	Заключение	17
4.1	Выводы	17

Список иллюстраций

2.1	Осуществим вход	6
2.2	Запишем названия файлов из /etc, потом из ~	7
2.3	работа с file.txt и .conf	8
2.4	поиск файлов с*	8
2.5	вывод файлов на h*	9
2.6	Ввод в файл в фоновом режиме	9
2.7	rm ~/logfile	10
2.8	Запуск geidt	10
2.9	grep sed	10
2.10	УБИЙСТВО	11
2.11	df du	11
2.12	вывод директорий ~	12

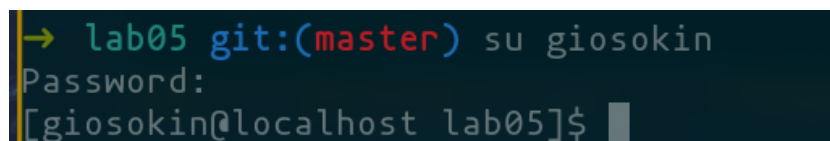
Список таблиц

1 Цель работы

Ознакомление с инструментами поиска файлов и фильтрации текстовых данных. Приобретение практических навыков: по управлению процессами (и заданиями), по проверке использования диска и обслуживанию файловых систем.

2 Выполнение лабораторной работы

2.1 Осуществим вход



```
→ lab05 git:(master) su giosokin
Password:
[giosokin@localhost lab05]$
```

Рис. 2.1: Осуществим вход

2.2 Впишем файлы в file.txt

Запишем названия файлов из /etc, потом из ~

```
[giosokin@localhost lab05]$ cd
[giosokin@localhost ~]$ ls /etc > file.txt
[giosokin@localhost ~]$ ls ~ >> file.txt
[giosokin@localhost ~]$ cat file.txt
adjtime
alsa
ansible
appstream.conf
arch-release
ardour7
audit
avahi
avrdude.conf
bash.bash_logout
bash.bashrc
bindresvport.blacklist
binfmt.d
bluetooth
brlapi.key
brltty.conf
ca-certificates
cifs-utils
conf.d
crypttab
cups
daxctl.conf.d
dconf
debuginfod
default
```

Рис. 2.2: Запишем названия файлов из /etc, потом из ~

2.3 Выведем имена всех файлов с расширением .conf

После выведем их в новый текстовый файл.

```

mdadm.conf
mke2fs.conf
mkinitcpio.conf
ndctl.conf.d
netconfig
nscd.conf
nsswitch.conf
pacman.conf
request-key.conf
resolv.conf
rsyncd.conf
sensors3.conf
sudo.conf
sudo_logsrvd.conf
tlp.conf
ts.conf
xattr.conf
xtables.conf
[giosokin@localhost ~]$ grep .conf file.txt > conf.txt
[giosokin@localhost ~]$ cat conf.txt
appstream.conf
avrdude.conf
brltty.conf
daxctl.conf.d
dconf
e2scrub.conf
fuse.conf

```

Рис. 2.3: работа с file.txt и .conf

2.4 Определим, какие файлы начинаются с с

Выведем с помощью команд `ls c*` и `file -n c*` и `ls | grep c*`

```

[giosokin@localhost ~]$ ls | grep c*
conf.txt
[giosokin@localhost ~]$ ls c*
conf.txt
[giosokin@localhost ~]$ file -n c*
conf.txt: ASCII text
[giosokin@localhost ~]$

```

Рис. 2.4: поиск файлов с*

2.5 Выведем имена файлов из каталога /etc

Выведем те, что начинаются на h, воспользовавшись командой find

```
[giosokin@localhost ~]$ find /etc -name "h*" -print
find: '/etc/pacman.d/gnupg/private-keys-v1.d': Permission denied
find: '/etc/pacman.d/gnupg/openpgp-revocs.d': Permission denied
find: '/etc/pacman.d/gnupg/crls.d': Permission denied
find: '/etc/audit/plugins.d': Permission denied
/etc/security/limits.d/horhik.conf
/etc/systemd/homed.conf
/etc/udev/hwdb.d
/etc/xdg/xfce4/helpers.rc
/etc/xdg/herbstluftwm
/etc/initcpio/hooks
/etc/hostname
find: '/etc/sudoers.d': Permission denied
/etc/hosts
find: '/etc/NetworkManager/system-connections': Permission denied
find: '/etc/polkit-1/rules.d': Permission denied
/etc/healthd.conf
/etc/avahi/hosts
find: '/etc/cups/ssl': Permission denied
/etc/mercurial/hgrc
/etc/host.conf
[giosokin@localhost ~]$
```

Рис. 2.5: вывод файлов на h*

2.6 Запустим фоновый просцц

Запустим процесс, который будет записывать в файл ~/logfile файлы, имена которых начинаются с log.

```
[giosokin@localhost ~]$ find /var/log/ -name "log*" -print > ~/logfile &
[1] 253033
[giosokin@localhost ~]$ find: '/var/log/audit': Permission denied
find: '/var/log/private': Permission denied

[1]+  Exit 1                  find /var/log/ -name "log*" -print > ~/logfile
[giosokin@localhost ~]$
```

Рис. 2.6: Ввод в файл в фоновом режиме

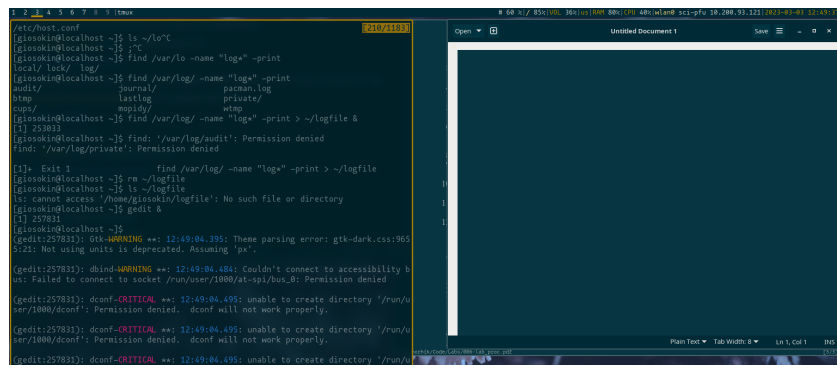
2.7 Удалим ~/logfile

Удалим командой rm

```
[giosokin@localhost ~]$ rm ~/logfile
[giosokin@localhost ~]$ ls ~/logfile
ls: cannot access '/home/giosokin/logfile': No such file or directory
[giosokin@localhost ~]$
```

Рис. 2.7: rm ~/logfile

2.8 Запустим geidt в фоновом режиме



```
/etc/host.conf
[giosokin@localhost ~]$ ls -la /etc
[giosokin@localhost ~]$ find /var -name "log*" -print
local/ lock/ log/
[giosokin@localhost ~]$ find /var/log -name "log*" -print
audit/      journal/
btup/       lastlog/
cups/       mspool/
[giosokin@localhost ~]$ find /var/log -name "log*" -print > ~/logfile &
[1] 257831
[giosokin@localhost ~]$ find: '/var/log/audit': Permission denied
Find: '/var/log/private': Permission denied

[1]+  Exit 1                  find /var/log/ -name "log*" -print > ~/logfile
[giosokin@localhost ~]$ rm ~/logfile
[giosokin@localhost ~]$ ls -la ~/logfile
ls: cannot access '/home/giosokin/logfile': No such file or directory
[giosokin@localhost ~]$ geidt &
[1] 257831
[giosokin@localhost ~]$
geidt:257831: gtk-dark.css:965
5:21: Not using units is deprecated. Assuming 'px'.
geidt:257831: dbus-WARNING **: 12:49:04.495: Couldn't connect to accessibility b
us: Failed to connect to socket /run/user/1000/at-spi/bus_0: Permission denied
geidt:257831: dconf-CRITICAL **: 12:49:04.495: unable to create directory /run/u
ser/1000/dconf: Permission denied. dconf will not work properly.
geidt:257831: dconf-CRITICAL **: 12:49:04.495: unable to create directory /run/u
ser/1000/dconf: Permission denied. dconf will not work properly.
geidt:257831: dconf-CRITICAL **: 12:49:04.495: unable to create directory /run/u
```

Рис. 2.8: Запуск geidt

2.9 Определим Id процесса

Воспользуемся ТРУБАМИ, командами grep и sed Напишем регулярное выражение, что бы вывести только id.

также можно просто воспользоваться командой pidof geidt

```
[giosokin@localhost ~]$ ps | grep geidt | sed 's/^\s*([0-9]*).*\s*/\1/'
261445
[giosokin@localhost ~]$
```

Рис. 2.9: grep | sed

2.10 УБЬЕМ geidt

командой 'kill \$(pidof)

```
[giosokin@localhost ~]$ sleep 3; kill $(pidof gedit)
[giosokin@localhost ~]$ pkill gedit
[giosokin@localhost ~]$ kill $(pidof gedit)
kill: usage: kill [-s sigspec | -n signum | -l corename] pid...
```

Рис. 2.10: УБИЙСТВО

2.11 df и du

Выведем информацию о размерах смонтированных разделов и о размерах файлов домашней директории с помощью df и du

```
[giosokin@localhost ~]$ du -d 1 ~ -h
70M    /home/giosokin/.cache
36M    /home/giosokin/.config
24K    /home/giosokin/.gnupg
68K    /home/giosokin/.pki
78M    /home/giosokin/work
24K    /home/giosokin/.ssh
7.2M   /home/giosokin/.texlive
12K    /home/giosokin/.local
30M    /home/giosokin/.mozilla
0      /home/giosokin/Downloads
16M    /home/giosokin/Desktop
8.0K   /home/giosokin/ski.places
128K   /home/giosokin/play
0      /home/giosokin/fun
236M   /home/giosokin
[giosokin@localhost ~]$ df -h
Filesystem      Size  Used Avail Use% Mounted on
dev              3.4G   0    3.4G   0% /dev
run              3.4G  1.1M   3.4G   1% /run
/dev/nvme0n1p1  128G  107G   19G   86% /
tmpfs            3.4G   23M   3.4G   1% /dev/shm
tmpfs            3.4G  1.7M   3.4G   1% /tmp
tmpfs            688M  112K   688M   1% /run/user/1000
[giosokin@localhost ~]$
```

Рис. 2.11: df || du

2.12 Выведем все директории домашнего каталога

Исполним команду `find ~ -mindepth 1 -maxdepth 1 -type d`

```
[giosokin@localhost ~]$ find ~ -mindepth 1 -maxdepth 1 -type d
/home/giosokin/.cache
/home/giosokin/.config
/home/giosokin/.gnupg
/home/giosokin/.pki
/home/giosokin/work
/home/giosokin/.ssh
/home/giosokin/.texlive
/home/giosokin/.local
/home/giosokin/.mozilla
/home/giosokin/Downloads
/home/giosokin/Desktop
/home/giosokin/ski.places
/home/giosokin/play
/home/giosokin/fun
[giosokin@localhost ~]$
```

Рис. 2.12: вывод директорий ~

3 Ответы на контрольные вопросы

3.1 Какие потоки ввода вывода вы знаете?

`stdin, stdout, stderr`

3.2 Объясните разницу между операцией `>` и `>>`.

`>` перезаписывает файл `>>` добавляет в конец файла

3.3 Что такое конвейер?

КОНВЕЕР ЭТО ТРУБА. Pipe по английски. Позволяет перенаправлять вывод одной команды, в другую.

3.4 Что такое процесс? Чем это понятие отличается от программы?

Процесс, это когда что-то происходит. ЭТО РАЗВИТИЕ, ДЕЙСТВИЕ. ЭВОЛЮЦИЯ. ну или просто программа, каждая програма является процессом.

В Linux процесс - это экземпляр программы, которая запущена в операционной системе. Каждый процесс имеет свой уникальный идентификатор (PID) и

хранит информацию о своем состоянии, включая текущее местоположение в коде, данные, регистры и т.д.

Понятие процесса отличается от понятия программы тем, что программа представляет собой набор инструкций и данных, записанных на диске, тогда как процесс - это активное выполнение программы в оперативной памяти. То есть, процесс может существовать только во время выполнения программы, в то время как программа может быть сохранена на диске в виде исполняемого файла, ожидая запуска.

3.5 Что такое PID и GID?

PID (Process ID) - это уникальный идентификатор процесса в операционной системе. Каждый процесс в операционной системе Linux имеет свой уникальный идентификатор PID, который используется для управления и отслеживания процессов в системе. PID может быть использован, например, для прерывания, приостановки или завершения процесса.

GID (Group ID) - это идентификатор группы пользователей в Linux. Каждый пользователь в Linux может принадлежать одной или нескольким группам пользователей, каждая из которых имеет свой уникальный идентификатор GID. Идентификатор группы используется для управления правами доступа к файлам и другим ресурсам в системе, которые могут быть разрешены или запрещены для конкретных групп пользователей.

3.6 Что такое задачи и какая команда позволяет ими управлять?

ЗАДАЧИ ЭТО ПРОЦЕССЫ

3.7 Найдите информацию об утилитах `top` и `htop`. Каковы их функции?

`htop` это расширенная версия `top`. Она позволяет просматривать запущенные процессы

3.8 Назовите и дайте характеристику команде поиска файлов. Приведите примеры использования этой команды

Команда поиска файлов в Linux называется `find`. Эта команда используется для поиска

Например, чтобы найти все файлы с расширением `.txt` в текущем каталоге и его подкаталогах, можно использовать следующую команду:

```
find . -name "*.txt" -type f
```

3.9 Можно ли по контексту (содержанию) найти файл?

Если да, то как?

можно найти файл по его контексту или содержанию. Для этого можно использовать команду `grep`, которая ищет указанный текст в указанных файлах или потоках данных.

Например, чтобы найти все строки с текстом “hello world” в файле `example.txt`, можно использовать следующую команду:

```
grep "hello world" example.txt
```

3.10 Как определить объем свободной памяти на жёстком диске?

с помощью df

3.11 Как определить объем вашего домашнего каталога?

с помощью du

```
du -h ~
```

12. Как удалить зависший процесс

```
kill -KILL <pid>
```


4 Заключение

4.1 Выводы

Мы ознакомились с инструментами поиска файлов и фильтрации текстовых данных. Приобрели практических навыков: по управлению процессами (и заданиями), по проверке использования диска и обслуживанию файловых систем.