Санкт-Петербургский политехнический университет Петра Великого

Кафедра компьютерных систем и программных технологий

**Отчёт по лабораторной работе №2**

**Дисциплина**: Операционные системы

**Тема**: Файловые системы

Выполнил студент гр. 3530901/70203 И.Д. Иванов

(подпись)

Преподаватель Е.В. Душутина (подпись)

“\_\_\_”\_\_\_\_\_\_\_\_\_\_\_\_\_ 2019 г.

Санкт-Петербург

2019

## **1. Цели работы:**

1) Изучение основных типов файлов и их назначения.

2) Изучение утилит работы с файлами и получение навыка написания скриптов.

3) Изучение функций пользователей и исследование различных прав доступа пользователя к файлам.

4) Исследование состава файловой системы и принципов её организации.

**2. Программа работы:**

1. Ознакомиться с типами файлов исследуемой ФС. Применяя утилиту ls, отфильтровать по одному примеру каждого типа файла используемой вами ФС. Комбинируя различные ключи утилиты рекурсивно просканировать все дерево, анализируя крайнюю левую позицию выходной информации, полученной посредством ls –l. Результат записать в выходной файл с указанием полного пути каждого примера. Выполнить задание сначала в консоли построчно, выбирая необходимые сочетания ключей (в командной строке), а затем оформить как скрипт с задаваемым в командной строке именем файла как параметр.

2. Получить все жесткие ссылки на заданный файл, находящиеся в разных каталогах пользовательского пространства (разными способами, не применяя утилиты file и find). Использовать конвееризацию и фильтрацию. Оформить в виде скрипта.

3. Проанализировать все возможные способы формирования символьных ссылок (ln, link,cp и т.д.), продемонстрировать их экспериментально. Предложить скрипт, подсчитывающий и перечисляющий все полноименные символьные ссылки на файл, размещаемые в разных местах файлового дерева.

4. Получить все символьные ссылки на заданный в качестве входного параметра файл, не используя file (разными способами, не применяя утилиту file).

5. Изучить утилиту find, используя ее ключи получить расширенную информацию о всех типах файлов. Создать примеры вложенных команд.

6. Проанализировать содержимое заголовка файла, а также файла-каталога с помощью утилит od и \*dump. Если доступ к файлу-каталогу возможен (для отдельных модификаций POSIXсовместимых ОС), проанализировать изменение его содержимого при различных операциях над элементами, входящими в его состав (файлами и подкаталогами).

7. Определить максимальное количество записей в каталоге. Изменить размер каталога, варьируя количество записей (для этого создать программу, порождающую новые файлы и каталоги, а затем удаляющую их, предусмотрев промежуточный и конечный вывод информации о размере подопытного каталога).

8. Ознакомиться с содержимым /etc/passwd, /etc/shadow, с утилитой /usr/bin/passwd, проанализировать права доступа к этим файлам.

9. Исследовать права владения и доступа, а также их сочетаемость

9.1. Привести примеры применения утилит chmod, chown к специально созданному для этих целей отдельному каталогу с файлами.

9.2. Расширить права исполнения экспериментального файла с помощью флага SUID.

9.3. Экспериментально установить, как формируются итоговые права на использование файла, если права пользователя и группы, в которую он входит, различны.

9.4. Сопоставить возможности исполнения наиболее часто используемых операций, варьируя правами доступа к файлу и каталогу.

10. Разработать «программу-шлюз» для доступа к файлу другого пользователя при отсутствии прав на чтение информации из этого файла. Провести эксперименты для случаев, когда пользователи принадлежат одной и разным группам. Сравнить результаты. Для выполнения задания применить подход, аналогичный для обеспечения функционирования утилиты /usr/bin/passwd (манипуляции с правами доступа, флагом SUID, а также размещением файлов).

11. Применяя утилиту df и аналогичные ей по функциональности утилиты, а также информационные файлы типа fstab, получить информацию о файловых системах, возможных для монтирования, а также установленных на компьютере реально.

11.1. Привести информацию об исследованных утилитах и информационных файлах с анализом их содержимого и форматов.

11.2. Привести образ диска с точки зрения состава и размещения всех ФС на испытуемом компьютере, а также образ полного дерева ФС, включая присоединенные ФС съемных и несъемных носителей. Проанализировать и указать формат таблицы монтирования.

11.3. Привести «максимально возможное» дерево ФС, проанализировать, где это указывается

12. Проанализировать и пояснить принцип работы утилиты file.

12.1. Привести алгоритм её функционирования на основе информационной базы, размещение и полное имя которой указывается в описании утилиты в технической документации ОС (как правило, /usr/share/file/magic.\*), а также содержимого заголовка файла, к которому применяется утилита. Определить, где находятся магические числа и иные характеристики, идентифицирующие тип файла, применительно к исполняемым файлам, а также файлам других типов.

12.2. Утилиту file выполнить с разными ключами.

12.3. Привести экспериментальную попытку с добавлением в базу собственного типа файла и его дальнейшей идентификацией. Описать эксперимент и привести последовательность действий для расширения функциональности утилиты file и возможности встраивания дополнительного типа файла в ФС (согласовать содержимое информационной базы и заголовка файла нового типа)

## **Система, на которой производится работа:**

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$uname -a

Linux BigAwesomeTurtle 5.2.0-kali2-686-pae #1 SMP Debian 5.2.9-2kali1 (2019-08-22) i686 GNU/Linux

Дистрибутив – Kali Linux

**3. Ход работы:**

Исправленный скрипт для автологирования, корректно обрабатывающий команду cd:

PS1=$(whoami)@$(hostname):$(pwd)$  
logfile=$(pwd)/logfile.txt  
  
echo -n "${PS1@P}";  
  
while read command;  
do  
echo "" » $logfile;  
echo "${PS1@P}$command" » $logfile;  
cdcommand=$(echo "$command" | cut -d" " -f1);  
  
if [[ $cdcommand != "cd" ]] ; then  
bash -c "$command" | tee -a $logfile  
else  
$command  
PS1=$(whoami)@$(hostname):$(pwd)$  
fi  
echo -n "${PS1@P}";  
done;  
printf '%s' "Exiting"  
echo;

Работа с интерпретаторами:

Интерпретаторы, установленные на данной системе:

asdf@BigAwesomeTurtle:~/lab\_1$ cat /etc/shells

# /etc/shells: valid login shells

/bin/sh

/bin/bash

/usr/bin/bash

/bin/rbash

/usr/bin/rbash

/bin/dash

/usr/bin/dash

/usr/bin/tmux

Использование оболочки, отличной от оболочки по умолчанию:

asdf@BigAwesomeTurtle:~$ sh

$ ls

autoLog Documents lab\_1 logfile.txt Pictures Report.txt Templates Videos

Desktop Downloads lab\_2 Music Public res\_link test

Смена оболочки по умолчанию:

asdf@BigAwesomeTurtle:~$ chsh -s /bin/sh

Для того, чтобы изменения вступили в силу, необходимо перезагрузить систему.

Возврат к исходной оболочке по умолчанию:

$ chsh -s /bin/bash

Изменение оболочки для текущего сеанса:

asdf@BigAwesomeTurtle:~$ SH=/bin/sh  
asdf@BigAwesomeTurtle:~$ echo $SH

/bin/sh

## **3.1. Ознакомление с типами файлов исследуемой ФС**

Комбинируя различные ключи утилиты рекурсивно просканировать все дерево, анализируя крайнюю левую позицию выходной информации, полученной посредством ls-l. Результат записать в выходной файл с указанием полного пути каждого примера. Оформить как скрипт с задаваемым в командной строке именем файла как параметр.

Скрипт:

#!/bin/bash  
ls -R -l /dev |sort -d |uniq -w1 > $1

Результат работы:

brw-rw--— 1 root disk 8, 0 Oct 13 10:55 sda  
c-------— 1 root root 5, 2 Oct 13 10:55 ptmx  
/dev:  
drwxrwxrwt 2 root root 40 Oct 13 10:55 mqueue  
lrwxrwxrwx 1 root root 10 Oct 13 10:55 1:9 -> ../urandom

## **3.2. Жесткие ссылки**

Получить все жесткие ссылки на заданный файл, находящиеся в разных каталогах пользовательского пространства (разными способами, не применяя утилиты file и find).

Скрипт:

#!/bin/bash

b=`ls -li $1 | cut -d " " -f 1`

c=`ls -R -li / 2> /dev/null | grep -E "$b"`

echo "$c"

Результат работы:

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$ln res.txt res\_link

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$./script\_hard\_link res.txt

4198862 -rw-r--r— 2 asdf asdf 229 Oct 13 13:18 res\_link

4198862 -rw-r--r— 2 asdf asdf 229 Oct 13 13:18 res.txt

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$cd ..  
asdf@BigAwesomeTurtle:/home/asdf$ln lab\_2/res.txt res\_link  
asdf@BigAwesomeTurtle:/home/asdf$cd lab\_2  
asdf@BigAwesomeTurtle:/home/asdf/lab\_2$./script\_hard\_link  
asdf@BigAwesomeTurtle:/home/asdf/lab\_2$./script\_hard\_link res.txt  
4198862 -rw-r--r— 3 asdf asdf 229 Oct 13 13:18 res\_link  
4198862 -rw-r--r— 3 asdf asdf 229 Oct 13 13:18 res\_link  
4198862 -rw-r--r— 3 asdf asdf 229 Oct 13 13:18 res.txt

## **3.3. Символьные ссылки**

Проанализировать все возможные способы формирования символьных ссылок (ln, link,cp ), продемонстрировать их экспериментально. Предложить скрипт, подсчитывающий и перечисляющий все полноименные символьные ссылки на файл, размещаемые в разных местах файлового дерева.

Скрипт:

#!/bin/bash

c=`ls -R -l / 2> /dev/null | grep -P "\-\>.{1,}$1$"`

echo "$c"

Результат работы:

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$ln -s file\_1 res\_on\_file\_1  
asdf@BigAwesomeTurtle:/home/asdf/lab\_2$cp -s file\_1 another\_link  
Файл another\_link был скопирован в другую директорию, после чего был запущен скрипт:   
  
asdf@BigAwesomeTurtle:/home/asdf/lab\_2$./script\_soft\_link file\_1  
lrwxrwxrwx 1 asdf asdf 6 Oct 13 13:54 another\_link -> file\_1  
lrwxrwxrwx 1 asdf asdf 6 Oct 13 13:53 another\_link -> file\_1  
lrwxrwxrwx 1 asdf asdf 6 Oct 13 13:52 res\_on\_file\_1 -> file\_1

Получили все символьные ссылки на заданный файл.

## **3.4. Получить все символьные ссылки**

Получить все символьные ссылки на заданный в качестве входного параметра файл, не используя file (разными способами, не применяя утилиту file).

См. пункт 3.3.

## **3.5. Утилита find.**

Изучить утилиту find, используя ее ключи получить расширенную информацию о всех типах файлов. Создать примеры вложенных команд.

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$find / -type d -ls 2>/dev/null | head -n 10

2 4 drwxr-xr-x 18 root root 4096 Sep 4 23:48 /

3670017 4 drwxr-xr-x 2 root root 4096 Aug 27 16:50 /srv

6553601 4 drwxr-xr-x 13 root root 4096 Sep 4 23:47 /usr

7212149 4 drwxr-xr-x 11 root root 4096 Sep 19 11:11 /usr/local

9307811 4 drwxr-xr-x 3 root root 4096 Sep 19 11:11 /usr/local/MATLAB

9307928 4 drwxr-xr-x 12 root root 4096 Sep 19 11:13 /usr/local/MATLAB/R2012a

9437206 4 drwxr-xr-x 21 root root 4096 Sep 19 11:13 /usr/local/MATLAB/R2012a/resources

9437207 4 drwxr-xr-x 3 root root 4096 Sep 19 11:12 /usr/local/MATLAB/R2012a/resources/MATLAB

9437208 40 drwxr-xr-x 64 root root 36864 Sep 19 11:12 /usr/local/MATLAB/R2012a/resources/MATLAB/en

9437691 4 drwxr-xr-x 3 root root 4096 Sep 19 11:12 /usr/local/MATLAB/R2012a/resources/MATLAB/en/graphics

Флаг type с параметром d указывает искать каталоги. Флаг ls генерирует вывод как команда ls –dgils. Данная команда выводит первые 10 найденных по заданному шаблону путей.

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$find / -type p -ls 2>/dev/null | head -n 10

1564 0 prw-----— 1 root root 0 Oct 13 10:55 /run/initctl

1512 0 prw-----— 1 root root 0 Oct 13 10:55 /run/dmeventd-client

1511 0 prw-----— 1 root root 0 Oct 13 10:55 /run/dmeventd-server

498699 0 prw-----— 1 root root 0 Oct 13 14:12 /run/systemd/inhibit/10.ref

24737 0 prw-----— 1 root root 0 Oct 13 11:01 /run/systemd/inhibit/4.ref

22206 0 prw-----— 1 root root 0 Oct 13 11:01 /run/systemd/inhibit/3.ref

19385 0 prw-----— 1 root root 0 Oct 13 10:55 /run/systemd/inhibit/2.ref

19919 0 prw-----— 1 root root 0 Oct 13 10:55 /run/systemd/inhibit/1.ref

20430 0 prw-----— 1 root root 0 Oct 13 11:01 /run/systemd/sessions/2.ref

Флаг type с параметром p указывает искать каналы. Флаг ls генерирует вывод как команда ls –dgils. Данная команда также выводит первые 10 найденных по заданному шаблону путей.

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$find / -type f -ls 2> /dev/null |head -n 10

9309671 72 -r--r--r— 1 root root 72740 Jan 15 2012 /usr/local/MATLAB/R2012a/license.txt

9438206 4 -r--r--r— 1 root root 603 Jul 21 2011 /usr/local/MATLAB/R2012a/resources/MATLAB/en/graphics/rotate3d.xml

9438743 4 -r--r--r— 1 root root 242 Jul 21 2011 /usr/local/MATLAB/R2012a/resources/MATLAB/en/graphics/exploreaccessor.xml

9438308 4 -r--r--r— 1 root root 219 Jul 21 2011 /usr/local/MATLAB/R2012a/resources/MATLAB/en/graphics/setpp.xml

9438882 4 -r--r--r— 1 root root 556 Sep 20 2011 /usr/local/MATLAB/R2012a/resources/MATLAB/en/graphics/axeslayoutmanager.xml

9438014 4 -r--r--r— 1 root root 701 Jul 21 2011 /usr/local/MATLAB/R2012a/resources/MATLAB/en/graphics/createlistener.xml

9437989 4 -r--r--r— 1 root root 269 May 13 2011 /usr/local/MATLAB/R2012a/resources/MATLAB/en/graphics/shape/internal/util/PrimitivePoint.xml

9437695 4 -r--r--r— 1 root root 356 May 13 2011 /usr/local/MATLAB/R2012a/resources/MATLAB/en/graphics/shape/internal/util/ImagePoint.xml

9438166 4 -r--r--r— 1 root root 259 Sep 20 2011 /usr/local/MATLAB/R2012a/resources/MATLAB/en/graphics/pointdatacursor.xml

9437869 4 -r--r--r— 1 root root 248 Sep 20 2011 /usr/local/MATLAB/R2012a/resources/MATLAB/en/graphics/scribestackmanager.xml

Флаг type с параметром f указывает искать файлы. Флаг ls генерирует вывод как команда ls –dgils. Данная команда также выводит первые 10 найденных по заданному шаблону путей.

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$find / -type c -ls 2>/dev/null |head -n 10

24243 0 crw-rw--— 1 root tty 7, 136 Oct 13 11:17 /dev/vcsa8

24242 0 crw-rw--— 1 root tty 7, 72 Oct 13 11:17 /dev/vcsu8

24241 0 crw-rw--— 1 root tty 7, 8 Oct 13 11:17 /dev/vcs8

19941 0 crw-rw--— 1 root tty 7, 135 Oct 13 10:55 /dev/vcsa7

19939 0 crw-rw--— 1 root tty 7, 71 Oct 13 10:55 /dev/vcsu7

19937 0 crw-rw--— 1 root tty 7, 7 Oct 13 10:55 /dev/vcs7

19676 0 crw-rw--— 1 root tty 7, 134 Oct 13 10:55 /dev/vcsa6

19675 0 crw-rw--— 1 root tty 7, 70 Oct 13 10:55 /dev/vcsu6

19674 0 crw-rw--— 1 root tty 7, 6 Oct 13 10:55 /dev/vcs6

19673 0 crw-rw--— 1 root tty 7, 133 Oct 13 10:55 /dev/vcsa5

Флаг type с параметром ‘c’ указывает искать специальные символы. Флаг ls генерирует вывод как команда ls –dgils. Данная команда также выводит первые 10 найденных по заданному шаблону путей.

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$find / -type b -ls 2>/dev/null | head -n 10

12693 0 brw-rw--— 1 root disk 8, 5 Oct 13 10:55 /dev/sda5

12692 0 brw-rw--— 1 root disk 8, 2 Oct 13 10:55 /dev/sda2

12691 0 brw-rw--— 1 root disk 8, 1 Oct 13 10:55 /dev/sda1

12690 0 brw-rw--— 1 root disk 8, 0 Oct 13 10:55 /dev/sd

Флаг type с параметром b указывает искать специальные блоки. Флаг ls генерирует вывод как команда ls –dgils. Данная команда также выводит первые 10 найденных по заданному шаблону путей.

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$find / -type s -ls 2>/dev/null | head -n 10

17749 0 srw-rw-rw- 1 root root 0 Oct 13 10:55 /run/dbus/system\_bus\_socket

17745 0 srw-rw-rw- 1 root root 0 Oct 13 10:55 /run/uuidd/request

17741 0 srw-rw-rw- 1 root root 0 Oct 13 10:55 /run/pcscd/pcscd.comm

21385 0 srw-rw-rw- 1 asdf asdf 0 Oct 13 11:01 /run/user/1000/bus

21393 0 srw-----— 1 asdf asdf 0 Oct 13 11:01 /run/user/1000/gnupg/S.gpg-agent.ssh

21391 0 srw-----— 1 asdf asdf 0 Oct 13 11:01 /run/user/1000/gnupg/S.dirmngr

21389 0 srw-----— 1 asdf asdf 0 Oct 13 11:01 /run/user/1000/gnupg/S.gpg-agent

21387 0 srw-----— 1 asdf asdf 0 Oct 13 11:01 /run/user/1000/gnupg/S.gpg-agent.browser

21383 0 srw-----— 1 asdf asdf 0 Oct 13 11:01 /run/user/1000/gnupg/S.gpg-agent.extra

21380 0 srw-rw-rw- 1 asdf asdf 0 Oct 13 11:01 /run/user/1000/pulse/native

Флаг type с параметром s указывает искать сокеты. Флаг ls генерирует вывод как команда ls –dgils. Данная команда также выводит первые 10 найденных по заданному шаблону путей.

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$find / -type l -ls 2> /dev/null | head -n 10  
9309364 0 lrwxrwxrwx 1 root root 16 Sep 19 11:12 /usr/local/MATLAB/R2012a/bin/glnx86/libtiff.so.3 -> libtiff.so.3.9.5  
9309360 0 lrwxrwxrwx 1 root root 18 Sep 19 11:12 /usr/local/MATLAB/R2012a/bin/glnx86/[libpng12.so](https://vk.com/away.php?utf=1&to=http%3A%2F%2Flibpng12.so).0 -> libpng12.so.0.39.0  
9309228 0 lrwxrwxrwx 1 root root 22 Sep 19 11:12 /usr/local/MATLAB/R2012a/bin/glnx86/libosgTerrain.so.65 -> libosgTerrain.so.2.8.3  
9309296 0 lrwxrwxrwx 1 root root 21 Sep 19 11:12 /usr/local/MATLAB/R2012a/bin/glnx86/[libportaudio.so](https://vk.com/away.php?utf=1&to=http%3A%2F%2Flibportaudio.so).2 -> [libportaudio.so](https://vk.com/away.php?utf=1&to=http%3A%2F%2Flibportaudio.so).2.0.0  
9309249 0 lrwxrwxrwx 1 root root 23 Sep 19 11:12 /usr/local/MATLAB/R2012a/bin/glnx86/libosgParticle.so.65 -> [libosgParticle.so](https://vk.com/away.php?utf=1&to=http%3A%2F%2FlibosgParticle.so).2.8.3  
9307967 0 lrwxrwxrwx 1 root root 17 Sep 19 11:11 /usr/local/MATLAB/R2012a/bin/glnx86/[libexpat.so](https://vk.com/away.php?utf=1&to=http%3A%2F%2Flibexpat.so).1 -> [libexpat.so](https://vk.com/away.php?utf=1&to=http%3A%2F%2Flibexpat.so).1.5.0  
9309357 0 lrwxrwxrwx 1 root root 16 Sep 19 11:12 /usr/local/MATLAB/R2012a/bin/glnx86/[libcurl.so](https://vk.com/away.php?utf=1&to=http%3A%2F%2Flibcurl.so).4 -> [libcurl.so](https://vk.com/away.php?utf=1&to=http%3A%2F%2Flibcurl.so).4.2.0  
9307972 0 lrwxrwxrwx 1 root root 19 Sep 19 11:11 /usr/local/MATLAB/R2012a/bin/glnx86/[libhdf5\_hl.so](https://vk.com/away.php?utf=1&to=http%3A%2F%2Flibhdf5_hl.so).6 -> [libhdf5\_hl.so](https://vk.com/away.php?utf=1&to=http%3A%2F%2Flibhdf5_hl.so).6.0.5  
9309240 0 lrwxrwxrwx 1 root root 15 Sep 19 11:12 /usr/local/MATLAB/R2012a/bin/glnx86/libosg.so.65 -> [libosg.so](https://vk.com/away.php?utf=1&to=http%3A%2F%2Flibosg.so).2.8.3  
9309350 0 lrwxrwxrwx 1 root root 14 Sep 19 11:12 /usr/local/MATLAB/R2012a/bin/glnx86/[libdf.so](https://vk.com/away.php?utf=1&to=http%3A%2F%2Flibdf.so).0 -> [libdf.so](https://vk.com/away.php?utf=1&to=http%3A%2F%2Flibdf.so).0.0.0

Флаг type с параметром l указывает искать ссылки. Флаг ls генерирует вывод как команда ls –dgils. Данная команда также выводит первые 10 найденных по заданному шаблону путей

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$find / -name res.txt 2>/dev/null

/home/asdf/lab\_2/res.txt

/home/asdf/lab\_1/res.txt

Флаг name позволяет найти файл с указанным именем.

Пример вложенной команды:

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$find ~ -maxdepth 1 -type f -exec ls {} +

/home/asdf/autoLog

/home/asdf/.bash\_history

/home/asdf/.bash\_logout

/home/asdf/.bashrc

/home/asdf/.bashrc.original

/home/asdf/.ICEauthority

/home/asdf/logfile.txt

/home/asdf/.profile

/home/asdf/Report.txt

/home/asdf/res\_link

/home/asdf/.viminfo

/home/asdf/.Xauthority

/home/asdf/.xfce4-session.verbose-log

/home/asdf/.xfce4-session.verbose-log.last

/home/asdf/.xsession-errors

/home/asdf/.xsession-errors.old

Выход команды find передаётся на вход команде ls.

## **3.6. Утилиты od и \*dump**

Проанализировать содержимое заголовка файла, а также файла-каталога с помощью утилит od и \*dump.

**od** — утилита в Unix системах для вывода дампа файла в восьмеричном формате. С разными параметрами, с помощью od можно увидеть содержимое файла в шестнадцатеричном, восьмеричном, десятичном и прочем (можно даже одновременно во всех форматах) виде с любого места любой длины.

Создадим файл task6.txt и запишем туда “some text”:

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$echo 'some text' » task6.txt

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$cat task6.txt

some text

Просмотрим содержимое созданного файла в восьмеричном формате с помощью утилиты od:

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$od -b task6.txt

0000000 163 157 155 145 040 164 145 170 164 012

0000012

Флаг ‘b’ позволяет вывести восьмеричное представление файла

В восьмеричном и символьном форматах:

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$od -bc task6.txt

0000000 163 157 155 145 040 164 145 170 164 012

s o m e t e x t \n

0000012

Флаг ‘с’ позволяет вывести символьное представление файла

Просмотрим содержимое файла something.c в восьмеричном и символьном форматах:

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$od -bc something.c

0000000 043 151 156 143 154 165 144 145 040 074 163 164 144 151 157 056

# i n c l u d e < s t d i o .

0000020 150 076 012 151 156 164 040 155 141 151 156 050 051 173 012 011

h > \n i n t m a i n ( ) { \n \t

0000040 160 162 151 156 164 146 050 042 045 163 134 156 042 054 040 042

p r i n t f ( " % s \ n " , "

0000060 110 145 154 154 157 042 051 073 012 011 162 145 164 165 162 156

H e l l o " ) ; \n \t r e t u r n

0000100 040 061 073 012 175

1 ; \n }

0000105

Просмотрим содержимое файла something.c в шестнадцатеричном формате:

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$od -h something.c

0000000 6923 636e 756c 6564 3c20 7473 6964 2e6f

0000020 3e68 690a 746e 6d20 6961 286e 7b29 090a

0000040 7270 6e69 6674 2228 7325 6e5c 2c22 2220

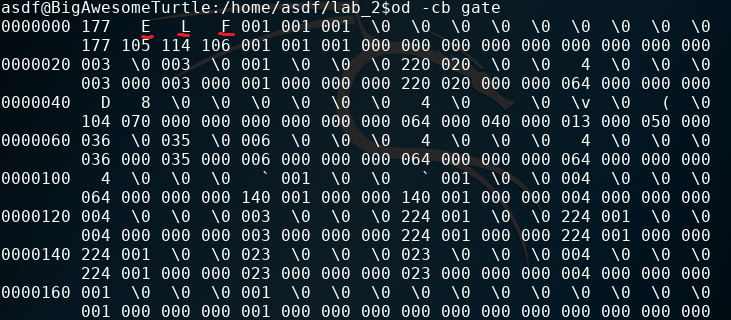
0000060 6548 6c6c 226f 3b29 090a 6572 7574 6e72

0000100 3120 0a3b 007d

0000105

Флаг ‘h’ позволяет вывести шестнадцатеричное представление файла

Применение утилиты od для получения заголовка исполняемого файла:

 ELF (Executable and Linkable Format) в начале файла означает, что перед нами исполняемый файл для UNIX-подобных систем.

Заголовок файла также можно получить с помощью утилиты objdump:

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$objdump -f gate

gate: file format elf32-i386

architecture: i386, flags 0x00000150:

HAS\_SYMS, DYNAMIC, D\_PAGED

start address 0x00001090

При попытке применить утилиту od к каталогу выводится сообщение об ошибке:

asdf@BigAwesomeTurtle:/home/asdf$od Documents

od: Documents: read error: Is a directory

0000000

При попытке применить утилиту objdump к каталогу также выводится сообщение об ошибке:

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$objdump -f dir\_for\_dirs

objdump: Warning: 'dir\_for\_dirs' is a directory

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$objdump -a dir\_for\_dirs

objdump: Warning: 'dir\_for\_dirs' is a directory

## **3.7. Максимальное количество записей в каталоге**

Определить максимальное количество записей в каталоге. Изменить размер каталога, варьируя количество записей (для этого создать программу, порождающую новые файлы и каталоги, а затем удаляющую их, предусмотрев промежуточный и конечный вывод информации о размере подопытного каталога).

Скрипт:

#!/bin/bash

counter=1

FileCounter1=0

FileCounter2=1

while [ $FileCounter1 -ne $FileCounter2 ]

do

FileCounter1=$FileCounter2

cp -R ~/lab\_2/dir\_for\_dirs "$counter"

FileCounter2=(`ls | wc -l`)

counter=$(($counter + 1))

done

while [ $counter -ne 0 ]

do

rm -r "$counter"

counter=$(($counter -1))

done

echo "$FileCounter1"

echo "$FileCounter2"

Данная программа копирует каталог dir\_for\_dirs до тех пор, пока не заполнится вся память устройства, либо пока не будет достигнуто ограничение на кол-во файлов для одного раздела. Т.к размер каталога крайне маленький, то он создает очень большое кол-во файлов. Количество файлов возможных для одной папки отличается для различных файловых систем. Например, в ext4 выделяется 655360 inodes на раздел и когда они кончаются, записать ничего нельзя.

## **3.8. Содержимое /etc/passwd и /etc/shadow**

Ознакомиться с содержимым /etc/passwd, /etc/shadow, с утилитой /usr/bin/passwd, проанализировать права доступа к этим файлам

Содержимое файла /etc/passwd:

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$cat /etc/passwd

root:x:0:0:root:/root:/bin/bash

daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin

bin:x:2:2:bin:/bin:/usr/sbin/nologin

sys:x:3:3:sys:/dev:/usr/sbin/nologin

sync:x:4:65534:sync:/bin:/bin/sync

games:x:5:60:games:/usr/games:/usr/sbin/nologin

man:x:6:12:man:/var/cache/man:/usr/sbin/nologin

lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin

mail:x:8:8:mail:/var/mail:/usr/sbin/nologin

news:x:9:9:news:/var/spool/news:/usr/sbin/nologin

uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin

proxy:x:13:13:proxy:/bin:/usr/sbin/nologin

www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin

backup:x:34:34:backup:/var/backups:/usr/sbin/nologin

list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin

irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin

gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin

nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin

\_apt:x:100:65534::/nonexistent:/usr/sbin/nologin

systemd-timesync:x:101:102:systemd Time Synchronization,,,:/run/systemd:/usr/sbin/nologin

systemd-network:x:102:103:systemd Network Management,,,:/run/systemd:/usr/sbin/nologin

systemd-resolve:x:103:104:systemd Resolver,,,:/run/systemd:/usr/sbin/nologin

messagebus:x:104:110::/nonexistent:/usr/sbin/nologin

uuidd:x:105:111::/run/uuidd:/usr/sbin/nologin

dnsmasq:x:106:65534:dnsmasq,,,:/var/lib/misc:/usr/sbin/nologin

usbmux:x:107:46:usbmux daemon,,,:/var/lib/usbmux:/usr/sbin/nologin

rtkit:x:108:114:RealtimeKit,,,:/proc:/usr/sbin/nologin

\_rpc:x:109:65534::/run/rpcbind:/usr/sbin/nologin

statd:x:110:65534::/var/lib/nfs:/usr/sbin/nologin

sshd:x:111:65534::/run/sshd:/usr/sbin/nologin

pulse:x:112:116:PulseAudio daemon,,,:/var/run/pulse:/usr/sbin/nologin

lightdm:x:113:119:Light Display Manager:/var/lib/lightdm:/bin/false

systemd-coredump:x:999:999:systemd Core Dumper:/:/usr/sbin/nologin

asdf:x:1000:1000::/home/asdf:/bin/bash

tcpdump:x:114:120::/nonexistent:/usr/sbin/nologin

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$ls -l /etc/passwd

-rw-r--r— 1 root root 1903 Sep 16 16:24 /etc/passwd

Владелец может читать и изменять этот файл, остальные могут только читать.

Файл /etc/passwdсодержит список пользователей, которые известны системе. В процессе регистрации пользователя система обращается к этому файлу в поисках идентификатора пользователя и его домашнего каталога. Каждая строка файла описывает одного пользователя и содержит семь полей, разделенных двоеточиями:

1) Регистрационное имя.

2) Зашифрованный пароль или «заполнитель» пароля.

3) Идентификатор пользователя

4) Идентификатор группы по умолчанию.

5) Поле персональных данных.

6) Домашний каталог.

7) Командный интерпретатор.

Содержимое файла /etc/shadow:

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$sudo cat /etc/shadow  
root:$6$0XUrmSMePpGSVzPr$1bQmBamCzqdnZCjmzVNcPzBJ4arhqHc6Br.4zjaPEDqJA5Au6JDWdzyvhglOcIPWHGXazEIiy/GI3QRD4D9/X1:18143:0:99999:7:::  
daemon:\*:18135:0:99999:7:::  
bin:\*:18135:0:99999:7:::  
sys:\*:18135:0:99999:7:::  
sync:\*:18135:0:99999:7:::  
games:\*:18135:0:99999:7:::  
man:\*:18135:0:99999:7:::  
lp:\*:18135:0:99999:7:::  
mail:\*:18135:0:99999:7:::  
news:\*:18135:0:99999:7:::  
uucp:\*:18135:0:99999:7:::  
proxy:\*:18135:0:99999:7:::  
www-data:\*:18135:0:99999:7:::  
backup:\*:18135:0:99999:7:::  
list:\*:18135:0:99999:7:::  
irc:\*:18135:0:99999:7:::  
gnats:\*:18135:0:99999:7:::  
nobody:\*:18135:0:99999:7:::  
\_apt:\*:18135:0:99999:7:::  
systemd-timesync:\*:18135:0:99999:7:::  
systemd-network:\*:18135:0:99999:7:::  
systemd-resolve:\*:18135:0:99999:7:::  
messagebus:\*:18135:0:99999:7:::  
uuidd:\*:18135:0:99999:7:::  
dnsmasq:\*:18135:0:99999:7:::  
usbmux:\*:18135:0:99999:7:::  
rtkit:\*:18135:0:99999:7:::  
\_rpc:\*:18135:0:99999:7:::  
statd:\*:18135:0:99999:7:::  
sshd:\*:18135:0:99999:7:::  
pulse:\*:18135:0:99999:7:::  
lightdm:\*:18135:0:99999:7:::  
systemd-coredump:!!:18143::::::  
asdf:$6$A5jHB4e4zveMCBtP$NoGh5HkIL1N6VDQ1sMa9XAJrK8FBHbe7rGzcGfAa3a2tpwQj.pmkxrz7b1efDOWRTwWXVFrc3d2iod3E0/094.:18150:0:99999:7:::  
tcpdump:\*:18155:0:99999:7:::

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$ls -l /etc/shadow

-rw-r---— 1 root shadow 1188 Sep 16 16:24 /etc/shadow

Файл /etc/shadow доступен для чтения только root и его группе, для записи только root, и предназначен для хранения зашифрованных паролей. В нем также содержится учетная информация, которая отсутствует в файле /etc/passwd. Каждая строка файла описывает одного пользователя и содержит поля, разделенные двоеточиями:

1) Регистрационное имя. Регистрационное имя берется из файла /etc/passwd. Оно связывает записи файлов /etc/passwd и /etc/shadow.

2) Зашифрованный пароль.

3) Дата последнего изменения пароля. Данное поле обычно заполняется командой passwd.

4) Минимальное число дней между изменениями пароля. В данном поле задается количество дней, спустя которые пользователь сможет снова изменить пароль.

5) Максимальное число дней между изменениями пароля. Максимальное время жизни пароля определяется суммой значений данного и седьмого полей.

6) Количество дней до истечения срока действия пароля, когда выдается предупреждение. В данном поле задано количество дней, оставшихся до момента устаревания пароля, когда программа login должна предупреждать пользователя о необходимости изменить пароль.

7) Количество дней по истечении срока действия пароля, когда учетная запись отключается.

8) Срок действия учетной записи. По окончанию этого срока пользователь не сможет зарегистрироваться в системе, пока администратор не сбросит значение поля. Если поле содержит пустое значение, учетная запись всегда будет активной.

9) Зарезервированное поле, которое в настоящее время всегда пустое.

Утилита /usr/bin/passwd служит для смены пароля:

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$/usr/bin/passwd

Current password:

New password:

Retype new password:

Changing password for asdf.

passwd: password updated successfully

Changing password for asdf.

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$ls -l /usr/bin/passwd

-rwsr-xr-x 1 root root 66284 Jul 16 19:48 /usr/bin/pass

Все пользователи могут запускать и читать этот файл. Создатель файла также может писать в файл.

## **3.9. Утилиты chmod, chown**

Исследовать права владения и доступа, а также их сочетаемость. Привести примеры применения утилит chmod, chown к специально созданному для этих целей отдельному каталогу с файлами. Расширить права исполнения экспериментального файла с помощью флага SUID. Экспериментально установить, как формируются итоговые права на использование файла, если права пользователя и группы, в которую он входит, различны

Устанавливаем: для владельца папки все права, для группы только чтение, для всех остальных только чтение:

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$chmod 744 dir\_task\_9

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$ls -l

total 64

drwxr--r— 2 asdf asdf 4096 Oct 13 19:33 dir\_task\_9

Устанавливаем: для всех все права:

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$chmod 777 dir\_task\_9

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$ls -l

total 64

drwxrwxrwx 2 asdf asdf 4096 Oct 13 19:33 dir\_task\_9

Смена владельца папки на root:

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$sudo chown root dir\_task\_9

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$ls -l

total 64

drwxrwxrwx 2 root asdf 4096 Oct 13 19:33 dir\_task\_9

Смена владельца папки на asdf, а группы на root:

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$sudo chown asdf:root dir\_task\_9

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$ls -l

total 64

drwxrwxrwx 2 asdf root 4096 Oct 13 19:33 dir\_task\_9

Возврат к изначальному состянию:

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$sudo chown asdf:asdf dir\_task\_9

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$ls -l

total 64

drwxrwxrwx 2 asdf asdf 4096 Oct 13 19:33 dir\_task\_9

Существуют два специальных бита: SUID (Set User ID - бит смены идентификатора пользователя) и SGID (Set Group ID - бит смены идентификатора группы).

Когда пользователь или процесс запускает исполняемый файл с установленным одним из этих битов, файлу временно назначаются права его владельца или группы (в зависимости от того, какой бит задан).

Установим бит SUID для исполняемого файла gate:

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$ls -l

total 100

-rw-r--r— 7 asdf asdf 229 Oct 2 15:05 aaa

lrwxrwxrwx 1 asdf asdf 6 Oct 13 13:53 another\_link -> file\_1

-rwxr-xr-x 1 asdf asdf 301 Oct 2 16:08 autoLog

drwxr-xr-x 2 asdf asdf 4096 Oct 13 16:12 dir\_for\_dirs

drwxrwxrwx 2 asdf asdf 4096 Oct 13 20:21 dir\_task\_9

-rw-r--r— 7 asdf asdf 229 Oct 2 15:05 file\_1

-rw-r--r— 1 asdf asdf 29 Oct 13 21:06 file\_task\_12

-rwxr-xr-x 1 asdf asdf 15604 Oct 13 20:10 gate

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$chmod u+s gate

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$ls -l

total 100

-rw-r--r— 7 asdf asdf 229 Oct 2 15:05 aaa

lrwxrwxrwx 1 asdf asdf 6 Oct 13 13:53 another\_link -> file\_1

-rwxr-xr-x 1 asdf asdf 301 Oct 2 16:08 autoLog

drwxr-xr-x 2 asdf asdf 4096 Oct 13 16:12 dir\_for\_dirs

drwxrwxrwx 2 asdf asdf 4096 Oct 13 20:21 dir\_task\_9

-rw-r--r— 7 asdf asdf 229 Oct 2 15:05 file\_1

-rw-r--r— 1 asdf asdf 29 Oct 13 21:06 file\_task\_12

**-rwsr-xr-x 1 asdf asdf 15604 Oct 13 20:10 gate**

В правах доступа для владельца x заменилась на s, говорящую о том, что для исполняемого файла установлен бит SUID.

В случае, если бит SUID или SGID установлен для неисполняемого файла, вместо s будет выведено S:

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$chmod u+s task6.txt  
asdf@BigAwesomeTurtle:/home/asdf/lab\_2$ls -l  
total 100  
-rw-r--r— 1 asdf asdf 61 Oct 13 15:28 something.h  
-rw-r--r— 1 asdf asdf 9 Oct 13 21:13 task\_13.txt  
**-rwSr--r— 1 asdf asdf 10 Oct 13 15:14 task6.txt**

Таким образом пользователь может даже запускать файлы от имени суперпользователя

(Пример продемонстрирован в пункте 3.10).

Если права на использование файла для владельца и для группы, в которую он входит различны, то для самого владельца файла права определяются исключительно его правами, независимо от прав группы-владельца, а для всех остальных членов этой группы права определяются правами группы

\* Сопоставить возможности исполнения наиболее часто используемых операций, варьируя правами доступа к файлу и каталогу

Команда

cd Для того чтобы перейти в директорию нужны права x

rm Чтобы удалить каталог или файл нужны права wx

cp Скопировать файл r ; Скопировать папку с файлом wx

cat Содержимое файла r ; Содержимое папки x

cat >> Записать в файл w ; Записать в папку x

./ Для файла rx ; Для папки X

./file.out Для файла x ; Для папки X

ls Для того чтобы вывести содержимое директории нужны права r

ls –l Для того чтобы вывести содержимое директории с итоговой строкой нужны права rx

## **3.10. Программа-шлюз**

Разработать «программу-шлюз» для доступа к файлу другого пользователя при отсутствии прав на чтение информации из этого файла.

Скрипт:

#include <stdio.h>

main (int argc, char \*argv []){

if (argc>1) {

FILE \*file;

file=fopen(argv[1],"r");

if (file) {

char str[64];

while ( fgets(str, sizeof(str), file) ) {

printf("%s", str);

}

fclose(file);

}

else printf("%s: Can't open file\n", argv[0], argv[1]);

}

else printf("%s: There is no filename\n", argv[0]);

}

Демонстрация работы:

asdf@BigAwesomeTurtle:/home/asdf/lab\_2/dir\_task\_9$ls -l

total 20

-rwxr-xr-x 1 asdf asdf 15604 Oct 13 20:21 gate

-rw-r--r— 1 asdf asdf 348 Oct 13 20:09 gate.c

-rw-----— 1 asdf asdf 0 Oct 27 13:26 read\_me\_as\_asdf

asdf@BigAwesomeTurtle:/home/asdf/lab\_2/dir\_task\_9$chmod u+s gate

asdf@BigAwesomeTurtle:/home/asdf/lab\_2/dir\_task\_9$ls -l

total 20

-rwsr-xr-x 1 asdf asdf 15604 Oct 13 20:21 gate

-rw-r--r— 1 asdf asdf 348 Oct 13 20:09 gate.c

-rw-----— 1 asdf asdf 0 Oct 27 13:26 read\_me\_as\_asdf

Создадим нового пользователя и переключимся на него:

asdf@BigAwesomeTurtle:/home/asdf/lab\_2/dir\_task\_9$sudo useradd task\_10\_user

asdf@BigAwesomeTurtle:/home/asdf/lab\_2/dir\_task\_9$su - task\_10\_user

Попытаемся прочитать файл, который доступен для чтения только пользователю asdf:

task\_10\_user@BigAwesomeTurtle:/home/asdf/lab\_2/dir\_task\_9$ ls -l

total 20

-rwsr-xr-x 1 asdf asdf 15604 Oct 13 20:21 gate

-rw-r--r— 1 asdf asdf 348 Oct 13 20:09 gate.c

-rw-----— 1 asdf asdf 0 Oct 27 13:26 read\_me\_as\_asdf

task\_10\_user@BigAwesomeTurtle:/home/asdf/lab\_2/dir\_task\_9$ cat read\_me\_as\_asdf

cat: read\_me\_as\_asdf: Permission denied

Прочитаем этот же файл через созданную программу:

task\_10\_user@BigAwesomeTurtle:/home/asdf/lab\_2/dir\_task\_9$ ./gate read\_me\_as\_asdf

Something important here

Как видно в примере с помощью данной программы мы смогли прочитать файл, к которому у нас изначально не было доступа.

## **3.11. Утилита df**

Применяя утилиту df и аналогичные ей по функциональности утилиты, а также информационные файлы типа fstab, получить информацию о файловых системах, возможных для монтирования, а также установленных на компьютере реально. Привести информацию об исследованных утилитах и информационных файлах с анализом их содержимого и форматов. Привести образ диска с точки зрения состава и размещения всех ФС на испытуемом компьютере, а также образ полного дерева ФС, включая присоединенные ФС съемных и несъемных носителей. Проанализировать и указать формат таблицы монтирования. Привести «максимально возможное» дерево ФС, проанализировать, где это указывается.

Утилита df редназначена для получения информации о свободном дисковом пространстве.

–h используется для измерения пространства в байтах, килобайтах и т.д.

–P используется для отображения заголовков столбцов таблицы

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$df -Ph

Filesystem Size Used Avail Use% Mounted on

udev 942M 0 942M 0% /dev

tmpfs 195M 5.9M 189M 3% /run

/dev/sda1 291G 12G 264G 5% /

tmpfs 971M 25M 946M 3% /dev/shm

tmpfs 5.0M 0 5.0M 0% /run/lock

tmpfs 971M 0 971M 0% /sys/fs/cgroup

tmpfs 194M 16K 194M 1% /run/user/1000

Файл /etc/fstab имеет большое значение для системы. Он содержит информацию о различных файловых системах и устройствах хранения информации компьютера, а также описывает, как диск (раздел) будет использоваться или как будет интегрирован в систему.

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$cat /etc/fstab

# /etc/fstab: static file system information.

#

# Use 'blkid' to print the universally unique identifier for a

# device; this may be used with UUID= as a more robust way to name devices

# that works even if disks are added and removed. See fstab(5).

#

# <file system> <mount point> <type> <options> <dump> <pass>

# / was on /dev/sda1 during installation

UUID=30fcaa48-2c02-4fd8-9f0a-c6dfad6dc1f7 / ext4 errors=remount-ro 0 1

# swap was on /dev/sda5 during installation

UUID=50cd9b2c-70c4-4e28-aee4-06b18912f183 none swap sw 0 0

Структура fstab:

Поле <file system> (файловая система) сообщает демону монтирования файловых систем mount, что монтировать, имя монтируемого устройства.

Второе поле, <dir> (директория), определяет путь, по которому будет смонтирована <file system>.

Поле <type> (тип) содержит тип файловой системы монтируемого устройства.

Следующее поле называется <options> (опции). Если используются все значения по умолчанию, то используется специальный ключ defaults.

Если хоть одна опция задана явно, то defaults указывать не нужно (defaults служит только для того, что была занята позиция в строке).

<dump> - используется утилитой dump для того чтобы определить, когда делать резервную копию. После установки, dump проверяет эту запись и использует значение, чтобы решить, подключать ли файловую систему. Возможные значения 0 или 1. Если 0, dump игнорирует файловую систему, если 1, dump сделает резервную копию. У большинства пользователей dump не установлен, поэтому в поле <dump> следует задать 0.

<pass> (номер прохода). fsck проверяет число, подставленное в поле <pass> и решает, в каком порядке проверять файловую систему. Возможные значения 0, 1 и 2.

Файловые системы со значением <pass>, равным 0, не будут проверены утилитой fsck. У корневой системы должен быть наибольший приоритет, 1, остальные файловые

системы должны иметь приоритет 2.

Когда программа mount подключает файловую систему, она дописывает соответствующую строку в /etc/mtab. Когда umount отключает файловую систему, из этого файла соответствующая строка удаляется.

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$cat /etc/mtab

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,noexec,relatime,size=964436k,nr\_inodes=186104,mode=755 0 0

devpts /dev/pts devpts rw,nosuid,noexec,relatime,gid=5,mode=620,ptmxmode=000 0 0

tmpfs /run tmpfs rw,nosuid,nodev,noexec,relatime,size=198660k,mode=755 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 0 0

tmpfs /run/lock tmpfs rw,nosuid,nodev,noexec,relatime,size=5120k 0 0

tmpfs /sys/fs/cgroup tmpfs ro,nosuid,nodev,noexec,mode=755 0 0

cgroup2 /sys/fs/cgroup/unified cgroup2 rw,nosuid,nodev,noexec,relatime,nsdelegate 0 0

cgroup /sys/fs/cgroup/systemd cgroup rw,nosuid,nodev,noexec,relatime,xattr,name=systemd 0 0

pstore /sys/fs/pstore pstore rw,nosuid,nodev,noexec,relatime 0 0

bpf /sys/fs/bpf bpf rw,nosuid,nodev,noexec,relatime,mode=700 0 0

cgroup /sys/fs/cgroup/cpu,cpuacct cgroup rw,nosuid,nodev,noexec,relatime,cpu,cpuacct 0 0

cgroup /sys/fs/cgroup/devices cgroup rw,nosuid,nodev,noexec,relatime,devices 0 0

cgroup /sys/fs/cgroup/net\_cls,net\_prio cgroup rw,nosuid,nodev,noexec,relatime,net\_cls,net\_prio 0 0

cgroup /sys/fs/cgroup/rdma cgroup rw,nosuid,nodev,noexec,relatime,rdma 0 0

cgroup /sys/fs/cgroup/cpuset cgroup rw,nosuid,nodev,noexec,relatime,cpuset 0 0

cgroup /sys/fs/cgroup/memory cgroup rw,nosuid,nodev,noexec,relatime,memory 0 0

cgroup /sys/fs/cgroup/freezer cgroup rw,nosuid,nodev,noexec,relatime,freezer 0 0

cgroup /sys/fs/cgroup/perf\_event cgroup rw,nosuid,nodev,noexec,relatime,perf\_event 0 0

cgroup /sys/fs/cgroup/blkio cgroup rw,nosuid,nodev,noexec,relatime,blkio 0 0

cgroup /sys/fs/cgroup/pids cgroup rw,nosuid,nodev,noexec,relatime,pids 0 0

mqueue /dev/mqueue mqueue rw,nosuid,nodev,noexec,relatime 0 0

debugfs /sys/kernel/debug debugfs rw,nosuid,nodev,noexec,relatime 0 0

hugetlbfs /dev/hugepages hugetlbfs rw,relatime,pagesize=2M 0 0

systemd-1 /proc/sys/fs/binfmt\_misc autofs rw,relatime,fd=45,pgrp=1,timeout=0,minproto=5,maxproto=5,direct,pipe\_ino=13635 0 0

binfmt\_misc /proc/sys/fs/binfmt\_misc binfmt\_misc rw,nosuid,nodev,noexec,relatime 0 0

tmpfs /run/user/1000 tmpfs rw,nosuid,nodev,relatime,size=198656k,mode=700,uid=1000,gid=1000 0 0

Смонтируем дополнительную файловую систему, используя файл test.ntfs:

asdf@BigAwesomeTurtle:/home/asdf/Downloads$sudo mount -o loop test.ntfs ~/lab\_2/mounted

Файловая система появилась в выводе команды df:

asdf@BigAwesomeTurtle:/home/asdf/Downloads$df -Ph

Filesystem Size Used Avail Use% Mounted on

udev 942M 0 942M 0% /dev

tmpfs 194M 5.9M 189M 4% /run

/dev/sda1 291G 14G 263G 5% /

tmpfs 970M 33M 938M 4% /dev/shm

tmpfs 5.0M 0 5.0M 0% /run/lock

tmpfs 970M 0 970M 0% /sys/fs/cgroup

tmpfs 194M 12K 194M 1% /run/user/1000

**/dev/loop0 25M 19M 6.3M 76% /home/asdf/lab\_2/mounted**

В файле /etc/mtab появилась новая строка:

asdf@BigAwesomeTurtle:/home/asdf/Downloads$cat /etc/mtab

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,noexec,relatime,size=964148k,nr\_inodes=185961,mode=755 0 0

devpts /dev/pts devpts rw,nosuid,noexec,relatime,gid=5,mode=620,ptmxmode=000 0 0

tmpfs /run tmpfs rw,nosuid,nodev,noexec,relatime,size=198600k,mode=755 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 0 0

tmpfs /run/lock tmpfs rw,nosuid,nodev,noexec,relatime,size=5120k 0 0

tmpfs /sys/fs/cgroup tmpfs ro,nosuid,nodev,noexec,mode=755 0 0

cgroup2 /sys/fs/cgroup/unified cgroup2 rw,nosuid,nodev,noexec,relatime,nsdelegate 0 0

cgroup /sys/fs/cgroup/systemd cgroup rw,nosuid,nodev,noexec,relatime,xattr,name=systemd 0 0

pstore /sys/fs/pstore pstore rw,nosuid,nodev,noexec,relatime 0 0

bpf /sys/fs/bpf bpf rw,nosuid,nodev,noexec,relatime,mode=700 0 0

cgroup /sys/fs/cgroup/freezer cgroup rw,nosuid,nodev,noexec,relatime,freezer 0 0

cgroup /sys/fs/cgroup/pids cgroup rw,nosuid,nodev,noexec,relatime,pids 0 0

cgroup /sys/fs/cgroup/cpu,cpuacct cgroup rw,nosuid,nodev,noexec,relatime,cpu,cpuacct 0 0

cgroup /sys/fs/cgroup/net\_cls,net\_prio cgroup rw,nosuid,nodev,noexec,relatime,net\_cls,net\_prio 0 0

cgroup /sys/fs/cgroup/blkio cgroup rw,nosuid,nodev,noexec,relatime,blkio 0 0

cgroup /sys/fs/cgroup/memory cgroup rw,nosuid,nodev,noexec,relatime,memory 0 0

cgroup /sys/fs/cgroup/rdma cgroup rw,nosuid,nodev,noexec,relatime,rdma 0 0

cgroup /sys/fs/cgroup/cpuset cgroup rw,nosuid,nodev,noexec,relatime,cpuset 0 0

cgroup /sys/fs/cgroup/perf\_event cgroup rw,nosuid,nodev,noexec,relatime,perf\_event 0 0

cgroup /sys/fs/cgroup/devices cgroup rw,nosuid,nodev,noexec,relatime,devices 0 0

systemd-1 /proc/sys/fs/binfmt\_misc autofs rw,relatime,fd=36,pgrp=1,timeout=0,minproto=5,maxproto=5,direct,pipe\_ino=12649 0 0

debugfs /sys/kernel/debug debugfs rw,nosuid,nodev,noexec,relatime 0 0

hugetlbfs /dev/hugepages hugetlbfs rw,relatime,pagesize=2M 0 0

mqueue /dev/mqueue mqueue rw,nosuid,nodev,noexec,relatime 0 0

binfmt\_misc /proc/sys/fs/binfmt\_misc binfmt\_misc rw,nosuid,nodev,noexec,relatime 0 0

tmpfs /run/user/1000 tmpfs rw,nosuid,nodev,relatime,size=198596k,mode=700,uid=1000,gid=1000 0 0

fusectl /sys/fs/fuse/connections fusectl rw,nosuid,nodev,noexec,relatime 0 0

**/dev/loop0 /home/asdf/lab\_2/mounted fuseblk rw,relatime,user\_id=0,group\_id=0,allow\_other,blksize=4096 0 0**

Отключим файловую систему командой umount:

asdf@BigAwesomeTurtle:/home/asdf/Downloads$sudo umount ~/lab\_2/mounted

asdf@BigAwesomeTurtle:/home/asdf/Downloads$df -Ph

Filesystem Size Used Avail Use% Mounted on

udev 942M 0 942M 0% /dev

tmpfs 194M 5.9M 189M 4% /run

/dev/sda1 291G 14G 263G 5% /

tmpfs 970M 34M 937M 4% /dev/shm

tmpfs 5.0M 0 5.0M 0% /run/lock

tmpfs 970M 0 970M 0% /sys/fs/cgroup

tmpfs 194M 12K 194M 1% /run/user/1000

Файловая система была успешно отключена.

## **3.12. Утилита file**

Проанализировать и пояснить принцип работы утилиты file. Привести алгоритм её функционирования на основе информационной базы, размещение и полное имя которой указывается в описании утилиты в технической документации ОС (как правило, /usr/share/file/magic.\*), а также содержимого заголовка файла, к которому применяется утилита. Определить, где находятся магические числа и иные характеристики, идентифицирующие тип файла, применительно к исполняемым файлам, а также файлам других типов. Утилиту file выполнить с разными ключами. Привести экспериментальную попытку с добавлением в базу собственного типа файла и его дальнейшей идентификацией. Описать эксперимент и привести последовательность действий для расширения функциональности утилиты file и возможности встраивания дополнительного типа файла в ФС

Утилита file предназначена для определения содержимого файла. Данная утилита вообще не смотрит на имя и расширение, а пытается идентифицировать файл только по содержимому, вследствие чего данную утилиту сложно обмануть.

Опции:

-z - типы файлов внутри zip архива.

-b - не выводить имя файла перед его типом;

-n - вывести имя файла перед его типом (по умолчанию)

-f - определяет типы файлов, имена которых лежат в файле после опции.

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$file task6.txt

task6.txt: ASCII text

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$file something.c

something.c: C source, ASCII text

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$file gate

gate: ELF 32-bit LSB pie executable, Intel 80386, version 1 (SYSV), dynamically linked, interpreter /lib/ld-linux.so.2, BuildID[sha1]=b0110ed647515393dbd8e7a65df7396b827544d6, for GNU/Linux 3.2.0, not stripped

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$file -L another\_link

another\_link: ASCII text

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$file -b something.c

C source, ASCII text

## asdf@BigAwesomeTurtle:/home/asdf/lab\_2$cat » file\_task\_12

## asdf@BigAwesomeTurtle:/home/asdf/lab\_2$file -f file\_task\_12

## something.c: C source, ASCII text

## gate: ELF 32-bit LSB pie executable, Intel 80386, version 1 (SYSV), dynamically linked, interpreter /lib/ld-linux.so.2, BuildID[sha1]=b0110ed647515393dbd8e7a65df7396b827544d6, for GNU/Linux 3.2.0, not stripped

## something.h: ASCII text

## **Попытка создать свой тип файла:**

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$sudo subl /etc/magic

В открывшийся файл была добавлена строка:

0 string new\_type name\_of\_type

Файл был скомпилирован:

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$file -C -m /etc/magic

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$cat » task\_13.txt

asdf@BigAwesomeTurtle:/home/asdf/lab\_2$file task\_13.txt

task\_13.txt: name\_of\_type

Тип файла был распознан.

**4. Вывод:**

В ходе выполнения лабораторной работы были получены навыки по взаимодействию с файловыми системами в Linux, навыки создания различного типа ссылок и их поиска,

рассмотрены права доступа к различным файлам и бит SUID, механизм определения типа файла. Также, была произведена попытка создания своего типа файла при помощи /etc/magic и были освоены утилиты и специальные файлы для определения файловых систем.

**Дополнения:**

**3.10.**

Пользователь task\_10\_user входит только в группу task\_10\_user. Так как папка dir\_task\_9 доступна для чтения и исполнения всем пользователям, он может в неё зайти. Файл gate также доступен для исполнения всем пользователям, поэтому task\_10\_user может его запустить, но он не может прочитать файл read\_me\_as\_asdf, так как права на чтение этого файла есть только у пользователя asdf.

**3.11.**

Используется петлевое(loop) устройство, образ test.ntfs привязывается к loop-устройству, и система может работать с этим устройством как с обычным блочным. Устройство видно в fdisk и прочих утилитах, взаимодействующих со смонтированными устройствами.