

Лабораторная работа №2

Дисциплина: Основы информационной безопасности

Феоктистов Владислав Сергеевич

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	10
4.1	Исполнение команд в консоли	10
4.2	Создание и использование скрипта	15
4.3	Таблицы прав и разрешенных действий	18
5	Выводы	27
	Список литературы	28

Список иллюстраций

4.1	Создание нового пользователя и пароля	10
4.2	Смена пользователя	11
4.3	Инфомация о текущем пользователе	12
4.4	Инфомация о всех пользователях в файле /etc/passwd	12
4.5	Права и расшрренные атрибуты каталогов в директории /home . .	13
4.6	Права и расшрренные атрибуты нового каталога ~/dir1	14
4.7	Изменение прав для каталога dir1	15
4.8	Создание bash-скрипта для автоматизации проверки	16
4.9	Создание bash-скрипта для автоматизации проверки	16
4.10	Создание bash-скрипта для автоматизации проверки	17
4.11	Создание bash-скрипта для автоматизации проверки	17
4.12	Запуск bash-скрипта	18

Список таблиц

3.1	Описание некоторых каталогов файловой системы GNU Linux . .	8
3.2	Описание некоторых используемых в работе команд	8
4.1	Установленные права и разрешенные действия	18
4.2	Минимальные права для совершения операций	26

1 Цель работы

Целью данной работы является: приобретение практических навыков работы в консоли с правами и атрибутами файлов и каталогов, закрепление теоретических основ дискреционного разграничения доступа в современных системах с открытым кодом на базе ОС Linux, проверка необходимых наборов прав для выполнения различных действий над файлами и каталогами, получение навыков чтения выделенных прав через консоль.

2 Задание

Создать нового пользователя под именем guest с паролем; войти в систему от имени нового пользователя и получить о нем всю необходимую информацию через консоль; проверить и изменить права на существующие и созданные файлы и каталоги; проверить как снятие всех атрибутов с директории повлияло на возможности работы с ним.

3 Теоретическое введение

В ОС Linux права доступа к файлам, атрибуты и владение управляют уровнем доступа, который система обрабатывает, а пользователи имеют к файлам. Это гарантирует, что только авторизованные пользователи и процессы могут получить доступ к определенным файлам и каталогам. Атрибуты состоят из девяти битов, которые и определяют права для разных групп пользователей. Первая тройка битов определяет права доступа для владельца, вторая тройка - для членов группы, последняя тройка - для всех остальных пользователей в системе. Каждая тройка битов (класс пользователей) определяет права на чтение, запись и исполнение. Эта концепция позволяет контролировать, какие пользователи могут читать, записывать (изменять) или выполнять файлы/каталоги.

Чтобы просмотреть права доступа к файлу, используется команда `ls` с опцией `-l`. Первый символ указывает тип файла. Это может быть обычный файл (`-`), каталог (`d`), символическая ссылка (`l`) или другие специфические типы файлов. Следующие девять символов предоставляют доступ к файлу, три тройки по три символа каждая (три пользователя, три типа прав: `r` - чтение, `w` - запись, `x` - исполнение).

Права доступа к файлу/каталогу можно изменить с помощью команды `chmod`. Только `root`, владелец файла или пользователь с привилегией `sudo` могут изменять права доступа к файлу или каталогу. Разрешения можно указывать с помощью символического, числового или справочного режимов.

Таблица 3.1: Описание некоторых каталогов файловой системы GNU Linux

Имя каталога	Описание каталога
/	Корневая директория, содержащая всю файловую систему
/bin	Основные системные утилиты, необходимые как в однопользовательском режиме, так и при обычной работе всем пользователям
/etc	Общесистемные конфигурационные файлы и файлы конфигурации установленных программ
/home	Содержит домашние директории пользователей, которые, в свою очередь, содержат персональные настройки и данные пользователя
/media	Точки монтирования для сменных носителей
/root	Домашняя директория пользователя root
/tmp	Временные файлы
/usr	Вторичная иерархия для данных пользователя

Таблица 3.2: Описание некоторых используемых в работе команд

Команда	Описание команды
useradd	Создание пользователя в Linux. Необходимо будет указать имя нового пользователя.
passwd	Создание и изменение пользовательских паролей. Необходимо будет указать имя пользователя, для которого нужно создать/изменить пароль.
pwd	Выводит полный путь от корневого каталога к текущему рабочему каталогу: в контексте которого (по умолчанию) будут исполняться выводимые команды.
whoami	Отображает имя вошедшего в систему пользователя.

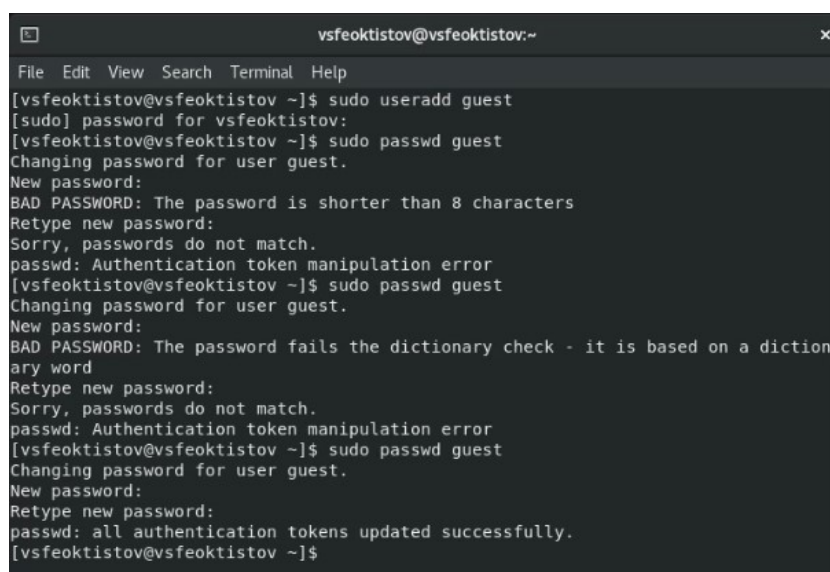
Ко-	
манда	Описание команды
<hr/>	
id	Выводи UID (идентификатор пользователя), GID (идентификатор группы пользователя), groups (основные группы пользователя)
groups	Выводит список групп, в которых состоит текущий пользователь или пользователь с указанным именем.
cat	Вывод содержимого указанного файла.
ls	Выводит содержимое каталога. Опция -l выводит дополнительную информацию, -a отображает скрытые файлы, в названии которых в самом начале стоит символ '.'
lsattr	Просмотр атрибутов файлов/каталогов в файловой системе Linux.
mkdir	Создание каталога по указанному пути и с указанным именем внутри пути.
chmod	Изменение прав доступа к файлам и каталогам, используемых в Unix-подобных операционных системах.
echo	Вывод переданных аргументов, строки, текста.
chattr	Изменяет атрибуты файлов/каталогов в файловой системе Linux.
touch	Создает текстовый файл по указанному пути и с указанным именем внутри пути.
rm	Удаляет файл(ы) (каталог(и) при указании опции -r) по указанному(ым) пути(ям).
rename	Переименование файла/каталога. Формат rename [старое имя] [новое имя] [путь до файла].
cd	Перемещение по файловой системе.
grep	Дает возможность вести поиск строк. Также можно передать вывод любой команды в grep, что сильно упрощает работу во время поиска

Более подробно об Unix см. в [1–6].

4 Выполнение лабораторной работы

4.1 Исполнение команд в консоли

Создаю нового пользователя под именем *guest* через команду `useradd` [cmd: `sudo useradd guest`] и создаю для него пароль с помощью команды `passwd` [cmd: `sudo passwd guest`]. После терминал попросит указать и подтвердить новый пароль (рис. 4.1).



```
vsfeoktistov@vsfeoktistov:~  
File Edit View Search Terminal Help  
[vsfeoktistov@vsfeoktistov ~]$ sudo useradd guest  
[sudo] password for vsfeoktistov:  
[vsfeoktistov@vsfeoktistov ~]$ sudo passwd guest  
Changing password for user guest.  
New password:  
BAD PASSWORD: The password is shorter than 8 characters  
Retype new password:  
Sorry, passwords do not match.  
passwd: Authentication token manipulation error  
[vsfeoktistov@vsfeoktistov ~]$ sudo passwd guest  
Changing password for user guest.  
New password:  
BAD PASSWORD: The password fails the dictionary check - it is based on a dictionary word  
Retype new password:  
Sorry, passwords do not match.  
passwd: Authentication token manipulation error  
[vsfeoktistov@vsfeoktistov ~]$ sudo passwd guest  
Changing password for user guest.  
New password:  
Retype new password:  
passwd: all authentication tokens updated successfully.  
[vsfeoktistov@vsfeoktistov ~]$
```

Рис. 4.1: Создание нового пользователя и пароля

Далее захожу в систему под только что созданным пользователем, вводя установленный ранее пароль. Сменить пользователя можно нажав на кнопку с иконкой замка в меню с выключением системы. В случае, если Вам предлагается ввести пароль для предыдущего пользователя, то нажмите на кнопку “Log in as

another user” (рис. 4.2).

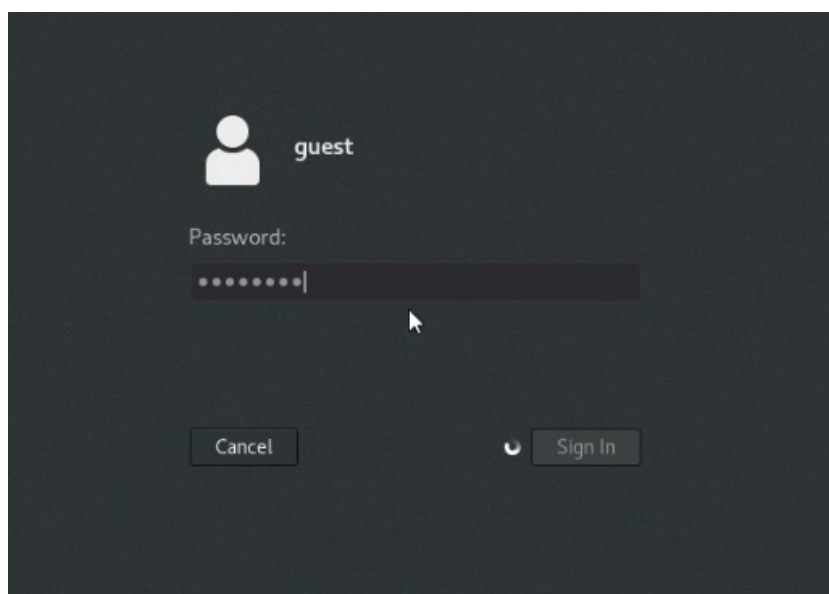


Рис. 4.2: Смена пользователя

После входа в систему под именем пользователя *guest*, я запустил терминал и выполнил команду *pwd*, тем самым получим путь до текущего каталога (домашняя директория пользователя *guest*). Далее можно уточнить имя пользователя, через которого я вошел в систему, для этого ввожу команду *whoami*. Очевидно, что выведется имя пользователя *guest*. С помощью команды *id* можно узнать UID пользователя (идентификатор пользователя = 1001), GID (идентификатор группы пользователя = 1001), а также к какой группе пользователей он принадлежит (на данный момент только к группе пользователей *guest*). Информацию о принадлежности текущего пользователя к группам пользователей можно также получить через команду *groups* (рис. 4.3).

```
guest@vsfeoktistov:~  
File Edit View Search Terminal Help  
[guest@vsfeoktistov ~]$ pwd  
/home/guest  
[guest@vsfeoktistov ~]$ whoami  
guest  
[guest@vsfeoktistov ~]$ id  
uid=1001(guest) gid=1001(guest) groups=1001(guest) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023  
[guest@vsfeoktistov ~]$ groups  
guest  
[guest@vsfeoktistov ~]$
```

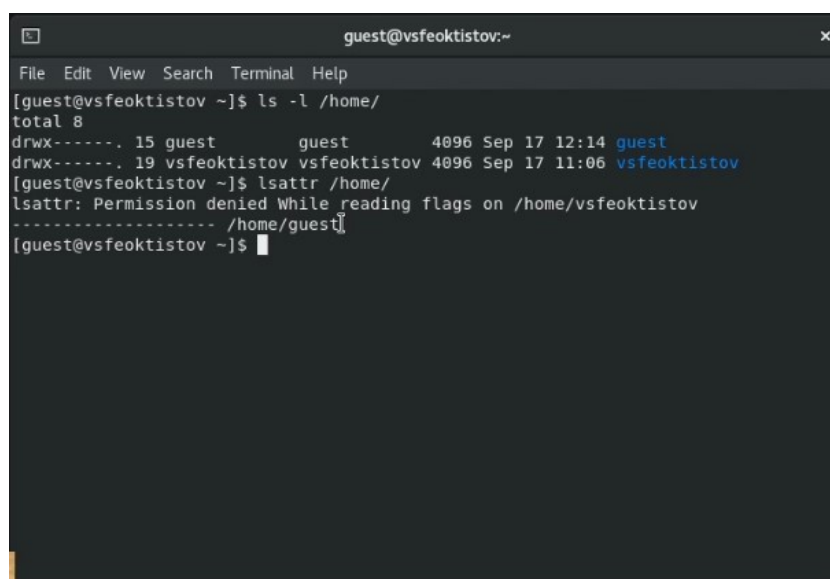
Рис. 4.3: Информация о текущем пользователе

Схожую информацию можно найти в содержании файла `/etc/passwd` (рис. 4.4).

```
guest:x:1001:1001::/home/guest:/bin/bash  
[guest@vsfeoktistov ~]$ cat /etc/passwd | grep -i "guest"  
guest:x:1001:1001::/home/guest:/bin/bash  
[guest@vsfeoktistov ~]$ cat /etc/passwd | grep guest  
guest:x:1001:1001::/home/guest:/bin/bash  
[guest@vsfeoktistov ~]$
```

Рис. 4.4: Информация о всех пользователях в файле `/etc/passwd`

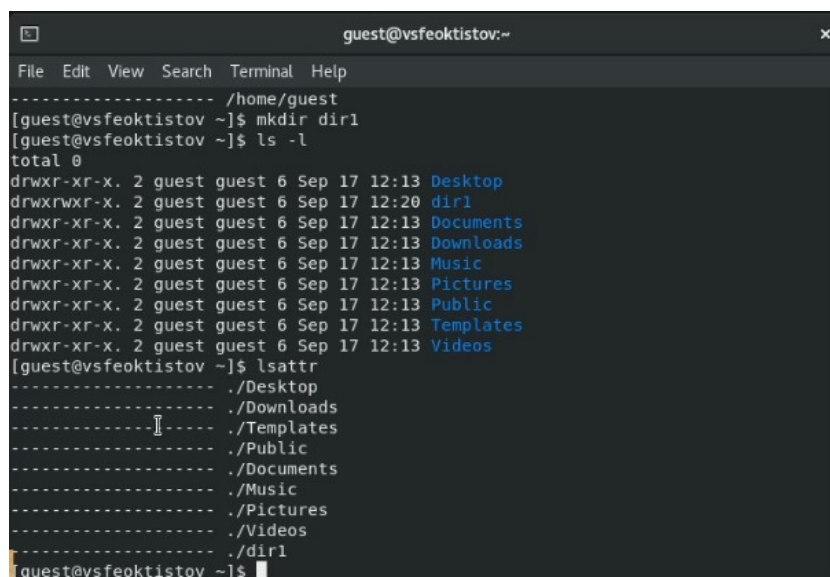
Далее я определил содержимое каталога `/home` с помощью команды `ls` [**cmd:** `ls -l /home`]. Получил информацию о двух каталогах: `vsfeoktistov` и `guest` - , которые находятся в указанной директории. Также через команду `lsattr` посмотрел расширенные атрибуты каталогов в директории `/home` [**cmd:** `lsattr /home`]. По выводу команды можно определить, что расширенные атрибуты можно увидеть только у файлов, владельцем которых является текущий пользователь (рис. 4.5).



```
guest@vsfeoktistov:~  
File Edit View Search Terminal Help  
[guest@vsfeoktistov ~]$ ls -l /home/  
total 8  
drwx-----. 15 guest      guest      4096 Sep 17 12:14 guest  
drwx-----. 19 vsfeoktistov vsfeoktistov 4096 Sep 17 11:06 vsfeoktistov  
[guest@vsfeoktistov ~]$ lsattr /home/  
lsattr: Permission denied While reading flags on /home/vsfeoktistov  
----- /home/guest  
[guest@vsfeoktistov ~]$
```

Рис. 4.5: Права и расширенные атрибуты каталогов в директории /home

После я создал каталог `dir1` в домашней директории [**cmd:** `mkdir dir1`] и посмотрел чем он отличается по правам и расширенным атрибутам от других файлов в домашней директории [**cmds:** `ls -l` и `lsattr`]. Как видно, каталог `dir1` имеет чуть больше прав, чем другие каталоги: члены группы имеют право редактирования/изменения (вторая тройка символов прав). В плане расширенных атрибутов каталог `dir1` ни чем не отличается от других: не имеет никаких расширенных атрибутов (рис. 4.6).



```
guest@vsfeoktistov:~  
File Edit View Search Terminal Help  
----- /home/guest  
[guest@vsfeoktistov ~]$ mkdir dir1  
[guest@vsfeoktistov ~]$ ls -l  
total 0  
drwxr-xr-x. 2 guest guest 6 Sep 17 12:13 Desktop  
drwxrwxr-x. 2 guest guest 6 Sep 17 12:20 dir1  
drwxr-xr-x. 2 guest guest 6 Sep 17 12:13 Documents  
drwxr-xr-x. 2 guest guest 6 Sep 17 12:13 Downloads  
drwxr-xr-x. 2 guest guest 6 Sep 17 12:13 Music  
drwxr-xr-x. 2 guest guest 6 Sep 17 12:13 Pictures  
drwxr-xr-x. 2 guest guest 6 Sep 17 12:13 Public  
drwxr-xr-x. 2 guest guest 6 Sep 17 12:13 Templates  
drwxr-xr-x. 2 guest guest 6 Sep 17 12:13 Videos  
[guest@vsfeoktistov ~]$ lsattr  
----- ./Desktop  
----- ./Downloads  
----- ./Templates  
----- ./Public  
----- ./Documents  
----- ./Music  
----- ./Pictures  
----- ./Videos  
----- ./dir1  
[guest@vsfeoktistov ~]$
```

Рис. 4.6: Права и расширенные атрибуты нового каталога ~/dir1

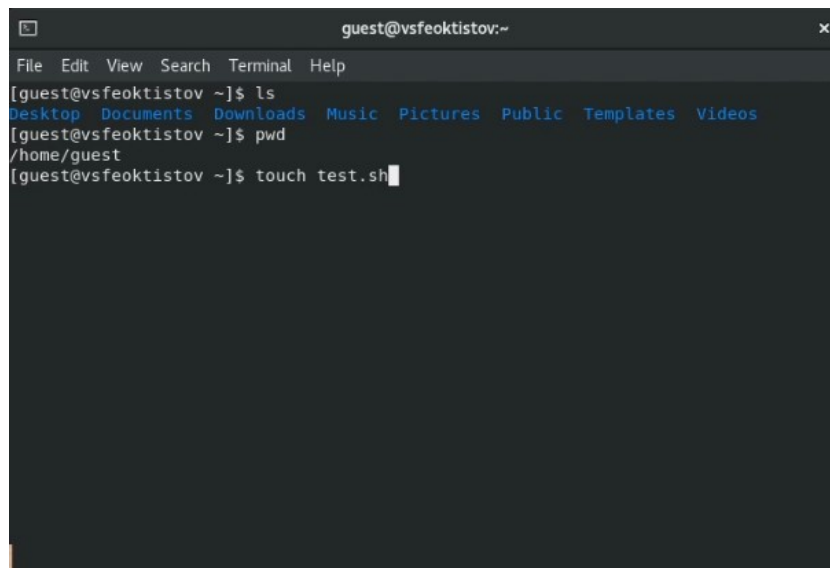
Далее я снял все атрибуты прав с директории dir1 [**cmd:** *chmod 000 dir1*] и проверил изменения, к которым эта команда привела: посмотрел какие права доступа теперь имеет этот каталог [**cmd:** *ls -l*], попробовал создать и внести данные в файл внутри этого каталога (файл не создался, поскольку все изменения в каталоге запрещены) [**cmd:** *echo "test" > /home/guest/dir1/file1*], попробовал просмотреть содержимое каталога (просмотр не удался, поскольку у каталога забрал права на чтение, т.е. просмотр) [**cmd:** *ls -l /home/guest/dir1*]. Проверить факт того, что файл file1 не было создан можно только после переназначения прав для каталога dir1 [**cmd:** *chmod 700 dir1*], либо командой [**cmd:** *chmod 400 dir1*] (дает права только на просмотр содержимого каталога) (рис. 4.7).

```
guest@vsfeoktistov:~  
File Edit View Search Terminal Help  
drwxr-xr-x. 2 guest guest 6 Sep 17 12:13 Music  
drwxr-xr-x. 2 guest guest 6 Sep 17 12:13 Pictures  
drwxr-xr-x. 2 guest guest 6 Sep 17 12:13 Public  
drwxr-xr-x. 2 guest guest 6 Sep 17 12:13 Templates  
drwxr-xr-x. 2 guest guest 6 Sep 17 12:13 Videos  
[guest@vsfeoktistov ~]$ echo "test" > /home/guest/dir1/file1  
bash: /home/guest/dir1/file1: Permission denied  
[guest@vsfeoktistov ~]$ ls -l /home/guest/dir1/  
ls: cannot open directory '/home/guest/dir1/': Permission denied  
[guest@vsfeoktistov ~]$ chmod 700 dir1  
[guest@vsfeoktistov ~]$ ls -l  
total 0  
drwxr-xr-x. 2 guest guest 6 Sep 17 12:13 Desktop  
drwx-----. 2 guest guest 6 Sep 17 12:20 dir1  
drwxr-xr-x. 2 guest guest 6 Sep 17 12:13 Documents  
drwxr-xr-x. 2 guest guest 6 Sep 17 12:13 Downloads  
drwxr-xr-x. 2 guest guest 6 Sep 17 12:13 Music  
drwxr-xr-x. 2 guest guest 6 Sep 17 12:13 Pictures  
drwxr-xr-x. 2 guest guest 6 Sep 17 12:13 Public  
drwxr-xr-x. 2 guest guest 6 Sep 17 12:13 Templates  
drwxr-xr-x. 2 guest guest 6 Sep 17 12:13 Videos  
[guest@vsfeoktistov ~]$ ls -l /home/guest/dir1/  
total 0  
[guest@vsfeoktistov ~]$
```

Рис. 4.7: Изменение прав для каталога dir1

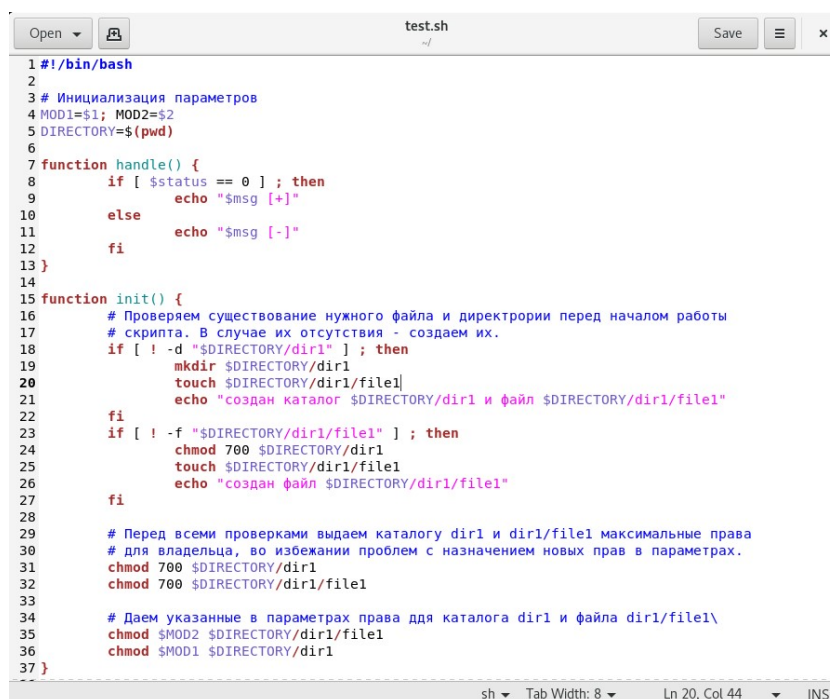
4.2 Создание и использование скрипта

Далее я изучил какие действия можно будет совершать над файлами/каталогами при различных комбинациях атрибутов прав доступа. Для этого можно последовательно выполнить ряд команд: *touch* - попытка создать файл, *rm* - попытка удалить файл, *echo "" > /path* - попытка записать данные в файл, *cat* - попытка прочитать информацию из файла, *cd* - попытка перейти в директорию, *ls* - попытка просмотреть содержимое директории, *rename* - попытка переименовать файл, *chattr* - попытка изменить расширенные атрибуты файла. Но поскольку всего таких комбинаций атрибутов $88=64$, то учитывая то, что нужно еще заполнить 8 колонок, то понадобится исполнить не менее $88*8=512$ команд, что достаточно много. Поэтому я написал bash скрипт, который упрощает проверку (рис. 4.8).



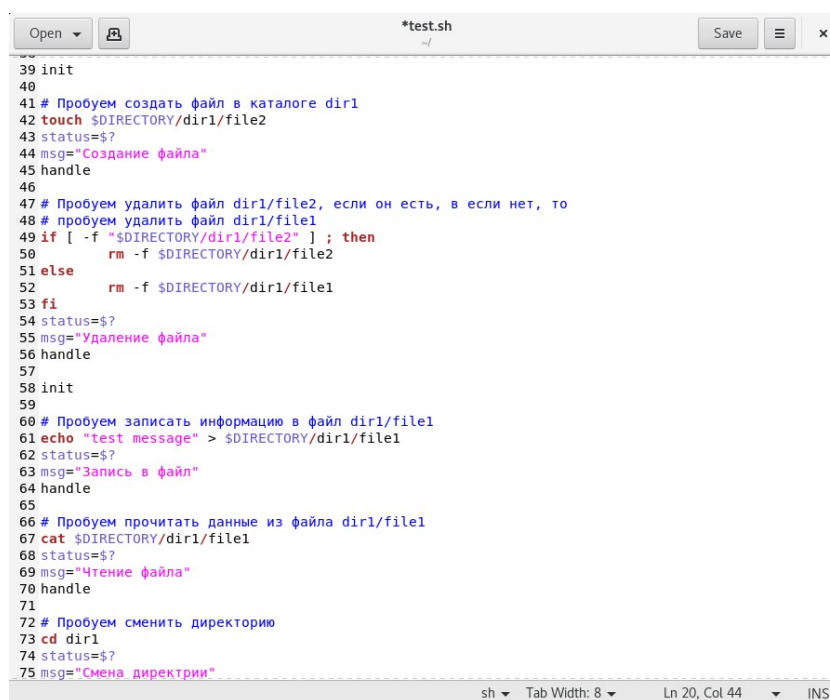
```
guest@vsfeoktistov:~  
File Edit View Search Terminal Help  
[guest@vsfeoktistov ~]$ ls  
Desktop Documents Downloads Music Pictures Public Templates Videos  
[guest@vsfeoktistov ~]$ pwd  
/home/guest  
[guest@vsfeoktistov ~]$ touch test.sh
```

Рис. 4.8: Создание bash-скрипта для автоматизации проверки



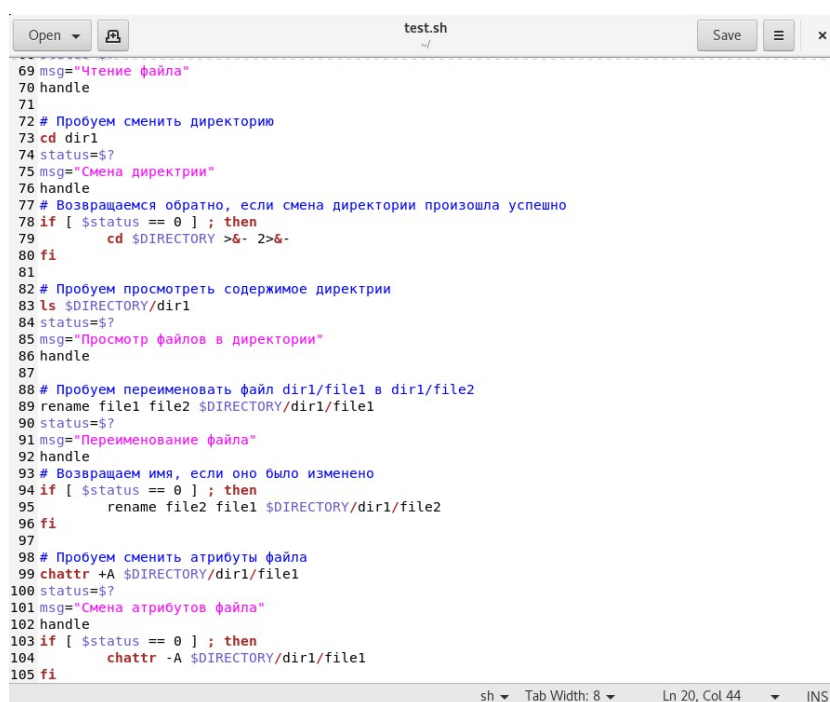
```
test.sh  
1 #!/bin/bash  
2  
3 # Инициализация параметров  
4 MOD1=$1; MOD2=$2  
5 DIRECTORY=$(pwd)  
6  
7 function handle() {  
8     if [ $status == 0 ] ; then  
9         echo "$msg [+]"  
10    else  
11        echo "$msg [-]"  
12    fi  
13 }  
14  
15 function init() {  
16     # Проверяем существование нужного файла и директории перед началом работы  
17     # скрипта. В случае их отсутствия - создаем их.  
18     if [ ! -d "$DIRECTORY/dir1" ] ; then  
19         mkdir $DIRECTORY/dir1  
20         touch $DIRECTORY/dir1/file1  
21         echo "создан каталог $DIRECTORY/dir1 и файл $DIRECTORY/dir1/file1"  
22     fi  
23     if [ ! -f "$DIRECTORY/dir1/file1" ] ; then  
24         chmod 700 $DIRECTORY/dir1  
25         touch $DIRECTORY/dir1/file1  
26         echo "создан файл $DIRECTORY/dir1/file1"  
27     fi  
28  
29     # Перед всеми проверками выдаем каталогу dir1 и dir1/file1 максимальные права  
30     # для владельца, во избежании проблем с назначением новых прав в параметрах.  
31     chmod 700 $DIRECTORY/dir1  
32     chmod 700 $DIRECTORY/dir1/file1  
33  
34     # Даем указанные в параметрах права для каталога dir1 и файла dir1/file1\  
35     chmod $MOD2 $DIRECTORY/dir1/file1  
36     chmod $MOD1 $DIRECTORY/dir1  
37 }
```

Рис. 4.9: Создание bash-скрипта для автоматизации проверки



```
39 init
40
41 # Пробуем создать файл в каталоге dir1
42 touch $DIRECTORY/dir1/file2
43 status=$?
44 msg="Создание файла"
45 handle
46
47 # Пробуем удалить файл dir1/file2, если он есть, в если нет, то
48 # пробуем удалить файл dir1/file1
49 if [ -f "$DIRECTORY/dir1/file2" ] ; then
50     rm -f $DIRECTORY/dir1/file2
51 else
52     rm -f $DIRECTORY/dir1/file1
53 fi
54 status=$?
55 msg="Удаление файла"
56 handle
57
58 init
59
60 # Пробуем записать информацию в файл dir1/file1
61 echo "test message" > $DIRECTORY/dir1/file1
62 status=$?
63 msg="Запись в файл"
64 handle
65
66 # Пробуем прочитать данные из файла dir1/file1
67 cat $DIRECTORY/dir1/file1
68 status=$?
69 msg="Чтение файла"
70 handle
71
72 # Пробуем сменить директорию
73 cd dir1
74 status=$?
75 msg="Смена директрии"
```

Рис. 4.10: Создание bash-скрипта для автоматизации проверки

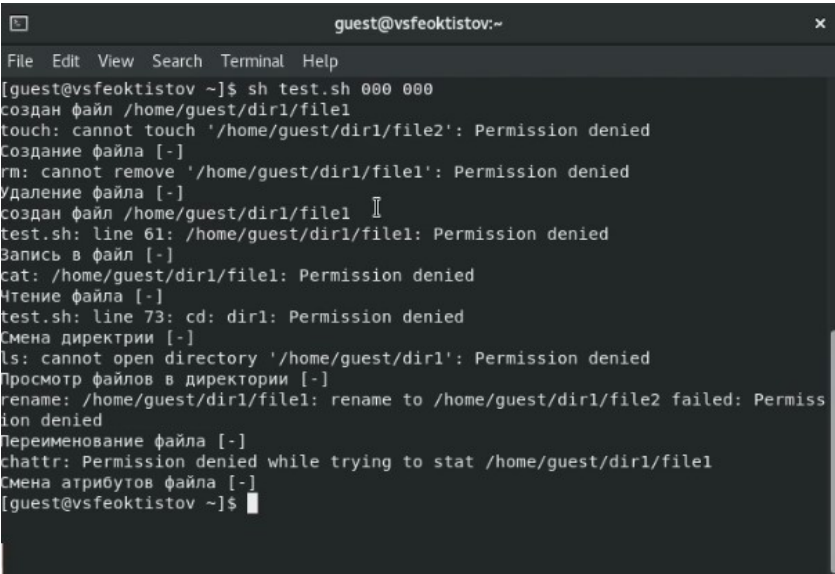


```
69 msg="Чтение файла"
70 handle
71
72 # Пробуем сменить директорию
73 cd dir1
74 status=$?
75 msg="Смена директрии"
76 handle
77 # Возвращаемся обратно, если смена директории произошла успешно
78 if [ $status == 0 ] ; then
79     cd $DIRECTORY >&- 2>&-
80 fi
81
82 # Пробуем просмотреть содержимое директрии
83 ls $DIRECTORY/dir1
84 status=$?
85 msg="Просмотр файлов в директории"
86 handle
87
88 # Пробуем переименовать файл dir1/file1 в dir1/file2
89 rename file1 file2 $DIRECTORY/dir1/file1
90 status=$?
91 msg="Переименование файла"
92 handle
93 # Возвращаем имя, если оно было изменено
94 if [ $status == 0 ] ; then
95     rename file2 file1 $DIRECTORY/dir1/file2
96 fi
97
98 # Пробуем сменить атрибуты файла
99 chatter +A $DIRECTORY/dir1/file1
100 status=$?
101 msg="Смена атрибутов файла"
102 handle
103 if [ $status == 0 ] ; then
104     chatter -A $DIRECTORY/dir1/file1
105 fi
```

Рис. 4.11: Создание bash-скрипта для автоматизации проверки

Запустить этот скрипт можно с помощью команды **[cmd: sh test.sh 000 000]**.

Таким образом, он снимет все права для файла file1 и каталога dir1 и покажет какие действия можно будет над ними выполнить (рис. 4.12).



```
guest@vsfeoktistov:~  
File Edit View Search Terminal Help  
[guest@vsfeoktistov ~]$ sh test.sh 000 000  
создан файл /home/guest/dir1/file1  
touch: cannot touch '/home/guest/dir1/file2': Permission denied  
Создание файла [-]  
rm: cannot remove '/home/guest/dir1/file1': Permission denied  
Удаление файла [-]  
создан файл /home/guest/dir1/file1  
test.sh: line 61: /home/guest/dir1/file1: Permission denied  
Запись в файл [-]  
cat: /home/guest/dir1/file1: Permission denied  
Чтение файла [-]  
test.sh: line 73: cd: dir1: Permission denied  
Смена директории [-]  
ls: cannot open directory '/home/guest/dir1': Permission denied  
Просмотр файлов в директории [-]  
rename: /home/guest/dir1/file1: rename to /home/guest/dir1/file2 failed: Permission denied  
Переименование файла [-]  
chattr: Permission denied while trying to stat /home/guest/dir1/file1  
Смена атрибутов файла [-]  
[guest@vsfeoktistov ~]$
```

Рис. 4.12: Запуск bash-скрипта

4.3 Таблицы прав и разрешенных действий

Таблица 4.1: Установленные права и разрешенные действия

Права					Смена				
ди-рек-то-рии	Прав	Со-зда-ние фай-ла	Уда-ле-ние фай-ла	За-пись в файл	Чте-ние фай-ла	ди-рек-то-рии	Про-смотр файло-в директо-рии	Пере-имено-вание фай-ла	Смена атрибу-тов фай-ла
d---	----	-	-	-	-	-	-	-	-
(000)	(000)								
d-x-	----	-	-	-	-	+	-	-	-
(100)	(000)								

Права		Смешанные				Смешанные			
ди-рек-то-рии	Права файла	Создание файла	Удаление файла	Запись в файл	Чтение файла	ди-рек-то-рии	Про-смотр файлов в директо-рии	Пере-имено-вание файла	Смена атрибу-тов файла
d-w(200)	----	-	-	-	-	-	-	-	-
d-wx(300)	----	+	+	-	-	+	-	+	-
dr(400)	----(000)	-	-	-	-	-	+	-	-
dr-x(500)	----(000)	-	-	-	-	+	+	-	-
drw(600)	----- (000)	-	-	-	-	-	+	-	-
drwx(700)	----- (000)	+	+	-	-	+	+	+	-

d-x(100)	---- -x---	-	-	-	-	-	-	-	-
d-x(100)	-x--- -x---	-	-	-	-	+	-	-	-
d-w(200)	-x--- (100)	-	-	-	-	-	-	-	-

Пра-						Сме-			
ва		Со-	Уда-			на	Про-		
ди-		зда-	ле-	За-	Чте-	ди-	смотр	Пере-	Смена
рек-		ние	ние	пись	ние	рек-	файлов в	имено-	атрибу-
то-	Права	фай-	фай-	в	фай-	то-	директо-	вание	тов
рии	файла	ла	ла	файл	ла	рии	рии	файла	файла

d-	-x---	+	+	-	-	+	-	+	-
----	-------	---	---	---	---	---	---	---	---

wx---	(100)								
-------	-------	--	--	--	--	--	--	--	--

(300)									
-------	--	--	--	--	--	--	--	--	--

dr---	-x---	-	-	-	-	-	+	-	-
-------	-------	---	---	---	---	---	---	---	---

(400)	(100)								
-------	-------	--	--	--	--	--	--	--	--

dr-	-x---	-	-	-	-	+	+	-	-
-----	-------	---	---	---	---	---	---	---	---

x---	(100)								
------	-------	--	--	--	--	--	--	--	--

(500)									
-------	--	--	--	--	--	--	--	--	--

drw---	-x---	-	-	-	-	-	+	-	-
--------	-------	---	---	---	---	---	---	---	---

(600)	(100)								
-------	-------	--	--	--	--	--	--	--	--

drwx---	-x---	+	+	-	-	+	+	+	-
---------	-------	---	---	---	---	---	---	---	---

(700)	(100)								
-------	-------	--	--	--	--	--	--	--	--

d---	-w---	-	-	-	-	-	-	-	-
------	-------	---	---	---	---	---	---	---	---

(000)	(200)								
-------	-------	--	--	--	--	--	--	--	--

d-x---	-w---	-	-	+	-	+	-	-	-
--------	-------	---	---	---	---	---	---	---	---

(100)	(200)								
-------	-------	--	--	--	--	--	--	--	--

d-	-w---	-	-	-	-	-	-	-	-
----	-------	---	---	---	---	---	---	---	---

w---	(200)								
------	-------	--	--	--	--	--	--	--	--

(200)									
-------	--	--	--	--	--	--	--	--	--

d-	-w---	+	+	+	-	+	-	+	-
----	-------	---	---	---	---	---	---	---	---

wx---	(200)								
-------	-------	--	--	--	--	--	--	--	--

(300)									
-------	--	--	--	--	--	--	--	--	--

Пра-						Сме-			
ва		Со-	Уда-			на	Про-		
ди-		зда-	ле-	За-	Чте-	ди-	смотр	Пере-	Смена
рек-		ние	ние	пись	ние	рек-	файлов в	имено-	атрибу-
то-	Права	фай-	фай-	в	фай-	то-	директо-	вание	тов
рии	файла	ла	ла	файл	ла	рии	рии	файла	файла
dr---	-w---	-	-	-	-	-	+	-	-
(400)	(200)								
dr-	-w---	-	-	+	-	+	+	-	-
x---	(200)								
(500)									
drw---	-w---	-	-	-	-	-	+	-	-
(600)	(200)								
drwx---	-w---	+	+	+	-	+	+	+	-
(700)	(200)								
<hr/>									
d---	-wx---	-	-	-	-	-	-	-	-
(000)	(300)								
d-x---	-wx---	-	-	+	-	+	-	-	-
(100)	(300)								
d-	-wx---	-	-	-	-	-	-	-	-
w---	(300)								
(200)									
d-	-wx---	+	+	+	-	+	-	+	-
wx---	(300)								
(300)									
dr---	-wx---	-	-	-	-	-	+	-	-
(400)	(300)								

Права		Создание		Удаление		Запись		Чтение		Смена		Про- смотр		Переименование		Смена атрибутов	
ди-рек-то-рии	Права файла	фай-ла	фай-ла	в файл	ла	фай-ла	ла	то-рии	то-рии	файлов в директо-рии	файла	файла	файла	файла	файла	файла	файла
dr- x---	-wx---	-	-	+	-	+	+	+	+	+	-	-	-	-	-	-	-
(500)	(300)																
drw---	-wx---	-	-	-	-	-	-	-	-	+	-	-	-	-	-	-	-
(600)	(300)																
drwx---	-wx---	+	+	+	-	+	+	+	+	+	+	-	-	+	-	-	-
(700)	(300)																
<hr/>																	
d---	r---	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
(000)	(400)																
d-x---	r---	-	-	-	+	+	+	+	+	-	-	-	-	-	-	+	+
(100)	(400)																
d- w---	r---	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
(200)	(400)																
d- wx---	r---	+	+	-	+	+	+	+	+	-	+	+	+	+	+	+	+
(300)	(400)																
dr----	r----	-	-	-	-	-	-	-	-	+	-	-	-	-	-	-	-
(400)	(400)																
dr- x---	r---	-	-	-	+	+	+	+	+	+	-	-	-	-	-	+	+
(500)	(400)																

Права		Смешанные				Смешанные			
ди-рек-то-рии	Права файла	Создание файла	Удаление файла	Запись в файл	Чтение файла	ди-рек-то-рии	Про-смотр файлов в директо-рии	Пере-имено-вание файла	Смена атрибу-тов файла
drw---	r---	-	-	-	-	-	+	-	-
(600)	(400)								
drwx---	r---	+	+	-	+	+	+	+	+
(700)	(400)								
<hr/>									
d----	r-x---	-	-	-	-	-	-	-	-
(000)	(500)								
d-x---	r-x---	-	-	-	+	+	-	-	+
(100)	(500)								
d- w----	r-x---	-	-	-	-	-	-	-	-
(200)	(500)								
d- wx---	r-x---	+	+	-	+	+	-	+	+
(300)	(500)								
dr----	r-x---	-	-	-	-	-	+	-	-
(400)	(500)								
dr- x---	r-x---	-	-	-	+	+	+	-	+
(500)	(500)								
drw---	r-x---	-	-	-	-	-	+	-	-
(600)	(500)								

Права		Смещение			Смена			Смена	
ва		Со-	Уда-	За-	Чте-	на	Про-	Пер-	Смена
ди-		зда-	ле-	пись	ние	ди-	смотр	имено-	атрибу-
рек-		ние	ние	в	ние	рек-	файлов в	вание	тов
то-	Права	фай-	фай-	файл	фай-	то-	директо-	файла	файла
рии	файла	ла	ла	файл	ла	рии	рии	файла	файла
drwx	r-x	+	+	-	+	+	+	+	+
(700)	(500)								
d	rw	-	-	-	-	-	-	-	-
(000)	(600)								
d-x	rw	-	-	+	+	+	-	-	+
(100)	(600)								
d-	rw	-	-	-	-	-	-	-	-
w	(600)								
(200)									
d-	rw	+	+	+	+	+	-	+	+
wx	(600)								
(300)									
dr	rw	-	-	-	-	-	+	-	-
(400)	(600)								
dr-	rw	-	-	+	+	+	+	-	+
x	(600)								
(500)									
drw	rw	-	-	-	-	-	+	-	-
(600)	(600)								
drwx	rw	+	+	+	+	+	+	+	+
(700)	(600)								

Права		Смешанные				Смешанные			
ди-рек-то-рии	Права файла	Создание файла	Удаление файла	Запись в файл	Чтение файла	ди-рек-то-рии	Про-смотр файлов в директо-рии	Пере-имено-вание файла	Смена атрибу-тов файла
d— — — (000)	rwX— — (700)	-	-	-	-	-	-	-	-
d-x— — (100)	rwX— — (700)	-	-	+	+	+	-	-	+
d- — — — (200)	rwX— — (700)	-	-	-	-	-	-	-	-
d- — — — (300)	rwX— — (700)	+	+	+	+	+	-	+	+
dr— — — (400)	rwX— — (700)	-	-	-	-	-	+	-	-
dr- — — (500)	rwX— — (700)	-	-	+	+	+	+	-	+
drw— — — (600)	rwX— — (700)	-	-	-	-	-	+	-	-
drwx— — — (700)	rwX— — (700)	+	+	+	+	+	+	+	+

Таблица 4.2: Минимальные права для совершения операций

Операция	Минимальные права на директорию	Минимальные права на файл
Создание файла	d-wx (300)	— (000)
Удаление файла	d-wx (300)	— (000)
Чтение файла	d-x (100)	r- (400)
Запись в файл	d-x (100)	-w- (200)
Переименование файла	d-wx (300)	— (000)
Создание поддиректории	d-wx (300)	— (000)
Удаление поддиректории	d-wx (300)	— (000)

5 Выводы

В процессе выполнения лабораторной работы я приобрел практические навыки работы в консоли с правами и атрибутами файлов и каталогов, закрепил теоретические основы дискреционного разграничения доступа в современных системах с открытым кодом на базе ОС Linux, проверил необходимый наборов прав для выполнения различных действий над файлами и каталогами, получил навыки чтения выделенных прав через консоль.

Список литературы

1. GNU Bash Manual [Электронный ресурс]. Free Software Foundation, 2016.
URL: <https://www.gnu.org/software/bash/manual/>.
2. Newham C. Learning the bash Shell: Unix Shell Programming. O'Reilly Media, 2005. 354 с.
3. Zarrelli G. Mastering Bash. Packt Publishing, 2017. 502 с.
4. Robbins A. Bash Pocket Reference. O'Reilly Media, 2016. 156 с.
5. Таненбаум Э. Архитектура компьютера. 6-е изд. СПб.: Питер, 2013. 874 с.
6. Таненбаум Э., Бос Х. Современные операционные системы. 4-е изд. СПб.: Питер, 2015. 1120 с.