

# Лабораторная работа № 2

Дискреционное разграничение прав в Linux. Основные атрибуты

Аксёнова Алина Владимировна

# Содержание

Цель работы	5
Задание	6
Теоретическое введение	7
Ход работы	9
Выводы	21
Библиографический список	22

# List of Figures

0.1	Создание новой учетной записи . . . . .	9
0.2	Пароль для новой учетной записи . . . . .	10
0.3	Вход в систему . . . . .	11
0.4	Определение текущей директории . . . . .	11
0.5	Уточнение имени пользователя . . . . .	12
0.6	Группы, в которые входит пользователь . . . . .	13
0.7	Вывод команды groups . . . . .	13
0.8	Данные в приглашении командной строки . . . . .	14
0.9	Просмотр файла /etc/passwd и определение uid и gid пользователя .	14
0.10	Поддиректории директории /home . . . . .	15
0.11	Проверка расширенных атрибутов . . . . .	15
0.12	Создание поддиректории . . . . .	16
0.13	Проверка прав доступа и расширенных атрибутов . . . . .	16
0.14	Создание файла в директории dir1 . . . . .	17
0.15	Проверка создания файла . . . . .	17
0.16	Процесс проверки разрешенных операций . . . . .	18
0.17	Заполненная таблица . . . . .	19
0.18	Проверка минимально необходимых прав для выполнения операций внутри директории . . . . .	20

## List of Tables

## Цель работы

Получение практических навыков работы в консоли с атрибутами файлов, закрепление теоретических основ дискреционного разграничения доступа в современных системах с открытым кодом на базе.

# Задание

Закрепить дискреционное разграничение прав в Linux.

# Теоретическое введение

В Linux, как и в любой многопользовательской системе, абсолютно естественным образом возникает задача разграничения доступа субъектов — пользователей к объектам — файлам дерева каталогов. Один из подходов к разграничению доступа — так называемый дискреционный - предполагает назначение владельцев объектов, которые по собственному усмотрению определяют права доступа субъектов (других пользователей) к объектам (файлам), которыми владеют. Дискреционные механизмы разграничения доступа используются для разграничения прав доступа процессов как обычных пользователей, так и для ограничения прав системных программ в (например, служб операционной системы), которые работают от лица псевдопользовательских учетных записей. Чтобы получить доступ к файлам в Linux, используются разрешения. Эти разрешения назначаются трем объектам: файлу, группе и другому объекту. Для управления правами используется команда `chmod`. При использовании `chmod` в относительном режиме вы работаете с тремя индикаторами, чтобы указать, что вы хотите сделать. Сначала вы указываете, для кого вы хотите изменить разрешения. Для этого вы можете выбрать между пользователем (u), группой (g) и другими (o). Затем вы используете оператор для добавления или удаления разрешений из текущего режима или устанавливаете их абсолютно. В конце вы используете r(read), w(write) и x(execute), чтобы указать, какие разрешения вы хотите установить. При использовании `chmod` вы можете устанавливать разрешения для пользователя (user), группы (group) и других (other). Помимо основных разрешений, о которых вы только что прочитали, в Linux также есть набор расширенных разрешений. Это не те разрешения, которые вы устанавливаете по

умолчанию, но иногда они предоставляют полезное дополнение.



## Ход работы

1. В установленной при выполнении предыдущей лабораторной работы операционной системе создаем учётную запись пользователя guest. (Рис. 0.1).

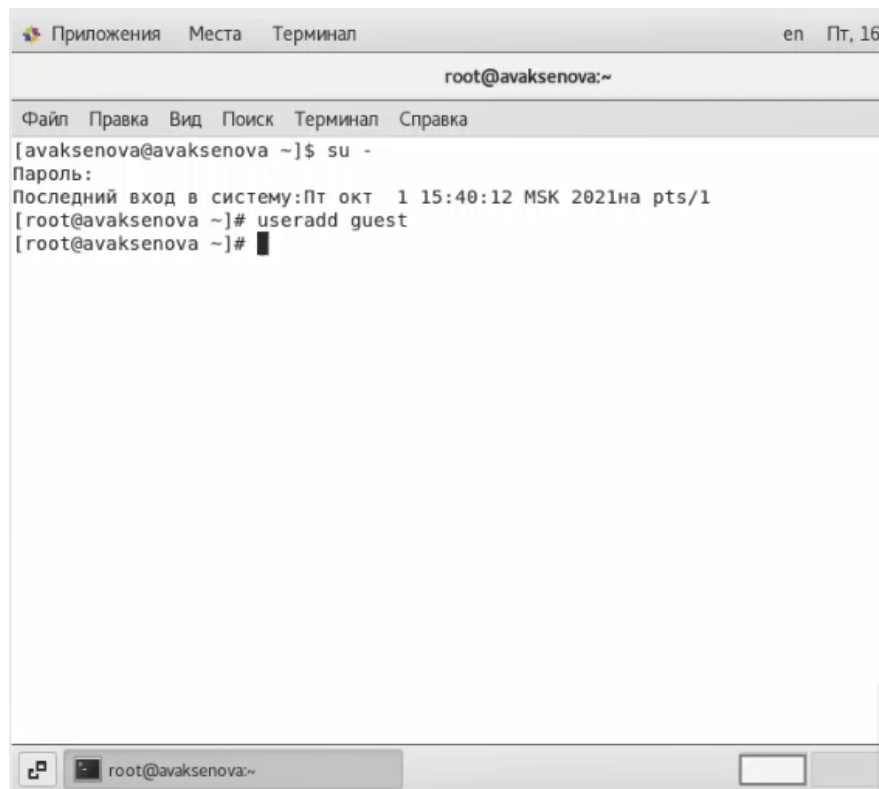


Figure 0.1: Создание новой учетной записи

2. Задаем пароль для пользователя guest (используя учётную запись администратора). (Рис. 0.2).

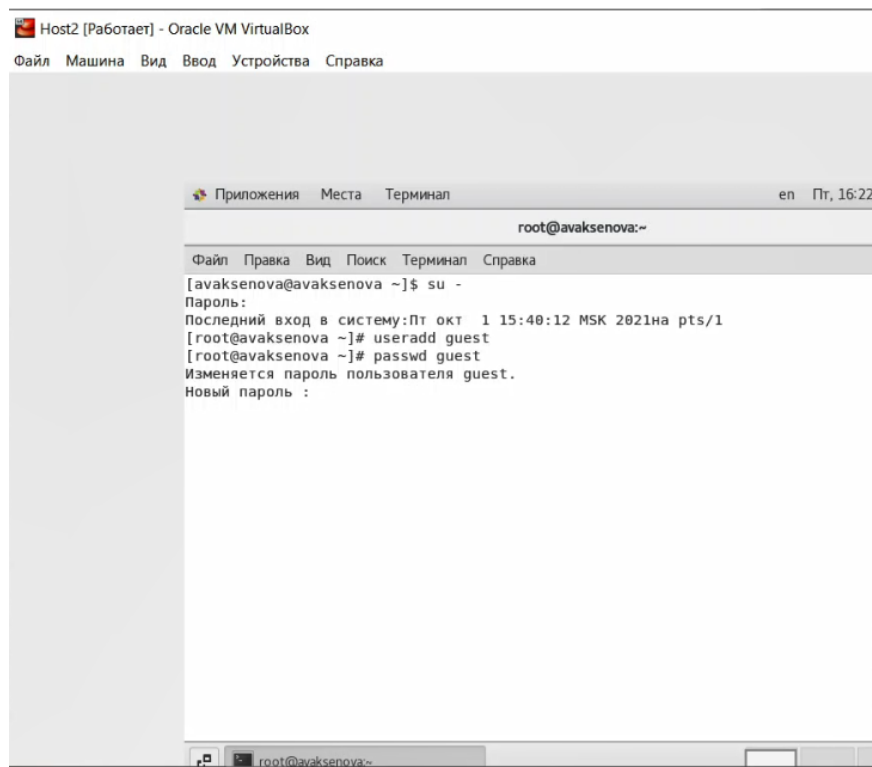


Figure 0.2: Пароль для новой учетной записи

3. Входим в систему от имени пользователя guest и определяем директорию, в которой мы находимся. Как можно заметить, мы находимся в домашней директории, о чем свидетельствует значок тильды, а также результат введения команды pwd. (Рис. 0.3, 0.4).

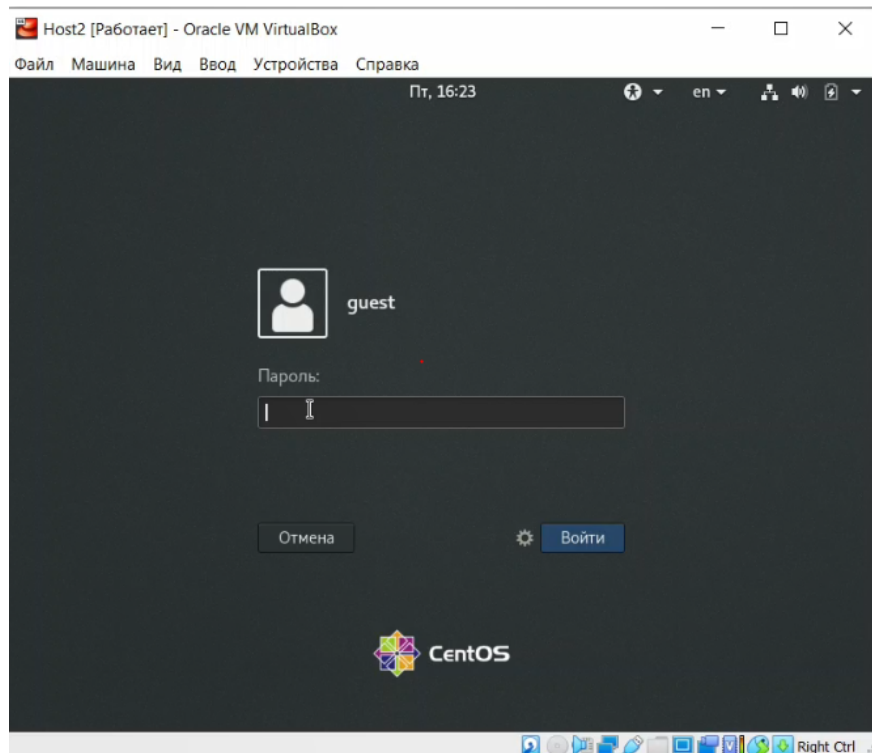


Figure 0.3: Вход в систему

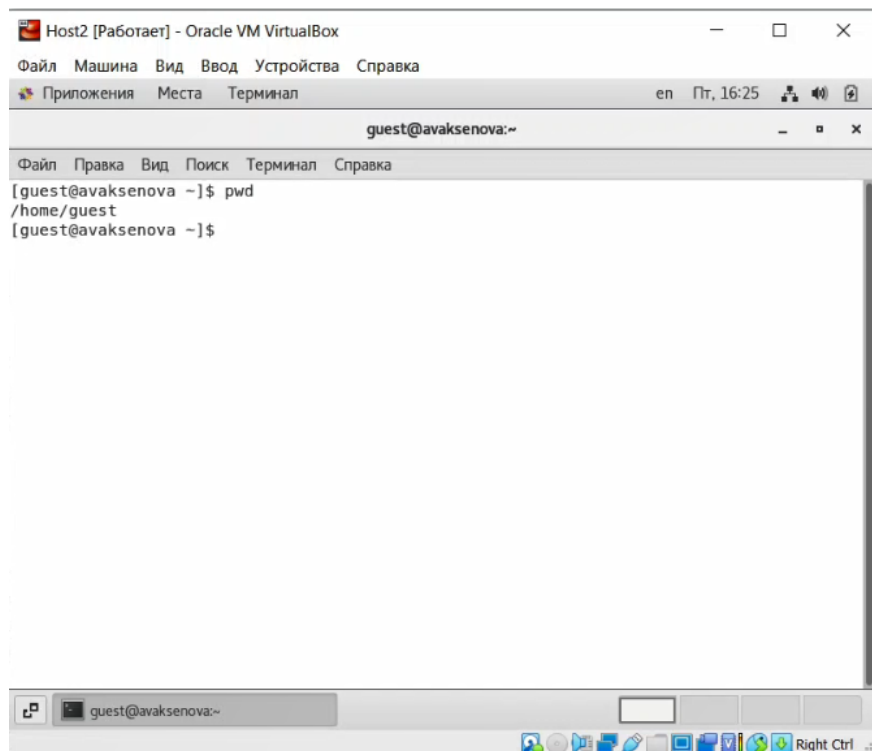
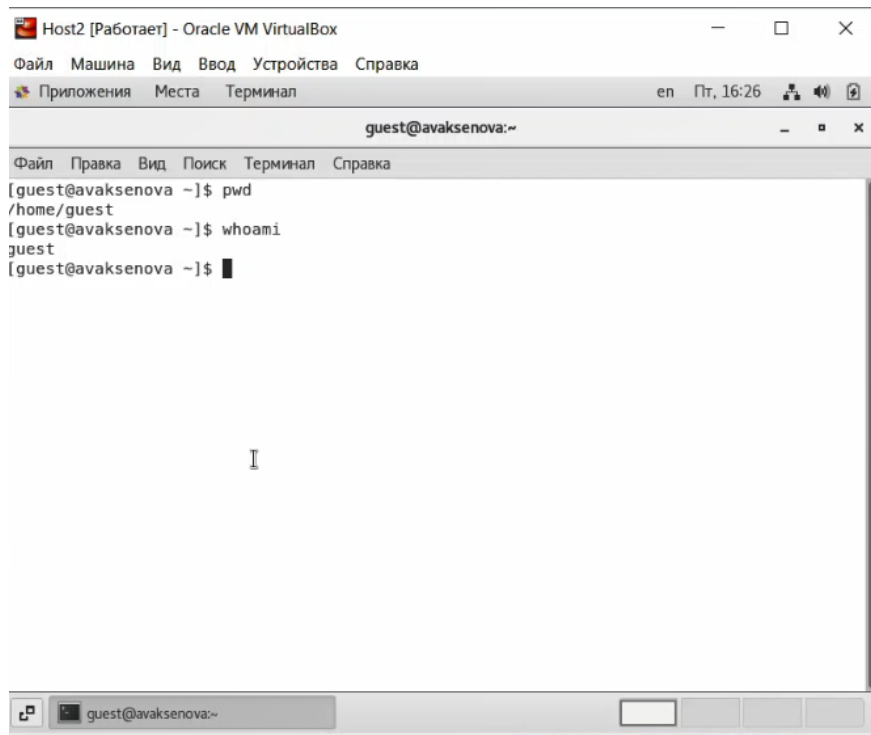


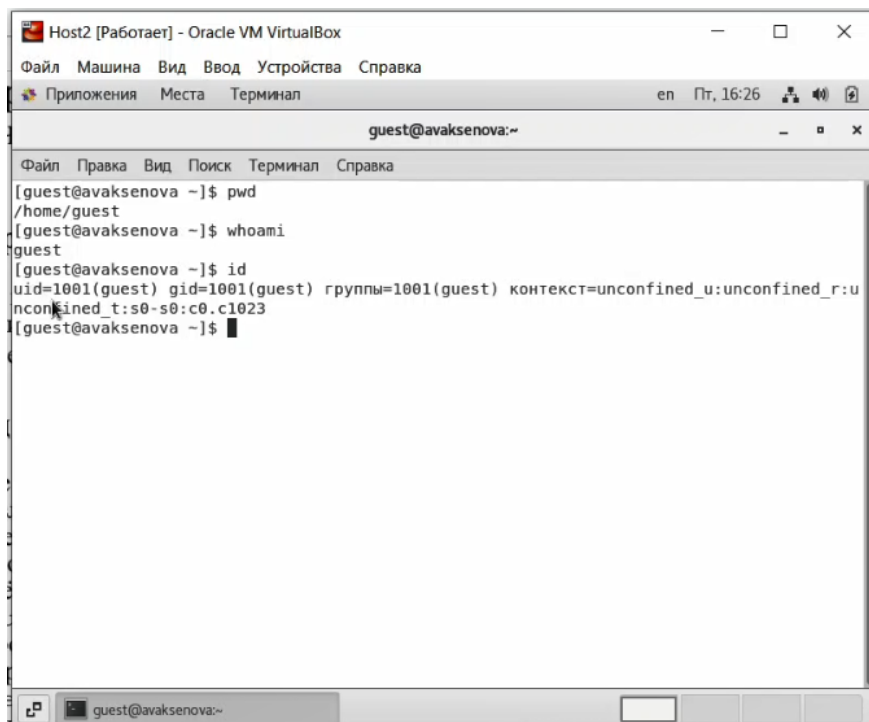
Figure 0.4: Определение текущей директории

4. Уточняем имя пользователя командой `whoami`. Уточняем его группу, а также группы, куда входит пользователь, командой `id`. Выведенные значения `uid`, `gid` сравниваем с выводом команды `groups`. Нетрудно заметить, что они совпадают. Кроме того, полученная информация совпадает с данными, выводимыми в приглашении командной строки. (Рис. 0.5, 0.6, 0.7, 0.8).



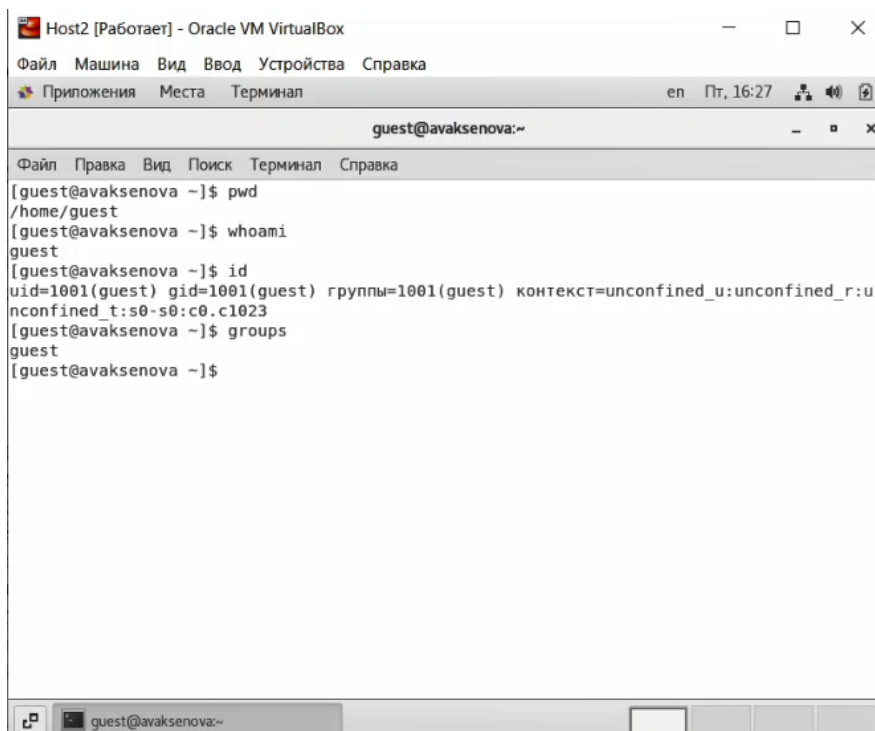
```
Host2 [Работаer] - Oracle VM VirtualBox
Файл  Машина  Вид  Ввод  Устройства  Справка
Приложения  Места  Терминал
en  Пт, 16:26
guest@avaksenova:~
Файл  Правка  Вид  Поиск  Терминал  Справка
[guest@avaksenova ~]$ pwd
/home/guest
[guest@avaksenova ~]$ whoami
guest
[guest@avaksenova ~]$
```

Figure 0.5: Уточнение имени пользователя



```
Host2 [Работает] - Oracle VM VirtualBox
Файл  Машина  Вид  Ввод  Устройства  Справка
Приложения  Места  Терминал  en  Пт, 16:26
guest@avaksenova:~
Файл  Правка  Вид  Поиск  Терминал  Справка
[guest@avaksenova ~]$ pwd
/home/guest
[guest@avaksenova ~]$ whoami
guest
[guest@avaksenova ~]$ id
uid=1001(guest) gid=1001(guest) группы=1001(guest) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[guest@avaksenova ~]$
```

Figure 0.6: Группы, в которые входит пользователь



```
Host2 [Работает] - Oracle VM VirtualBox
Файл  Машина  Вид  Ввод  Устройства  Справка
Приложения  Места  Терминал  en  Пт, 16:27
guest@avaksenova:~
Файл  Правка  Вид  Поиск  Терминал  Справка
[guest@avaksenova ~]$ pwd
/home/guest
[guest@avaksenova ~]$ whoami
guest
[guest@avaksenova ~]$ id
uid=1001(guest) gid=1001(guest) группы=1001(guest) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[guest@avaksenova ~]$ groups
guest
[guest@avaksenova ~]$
```

Figure 0.7: Вывод команды groups

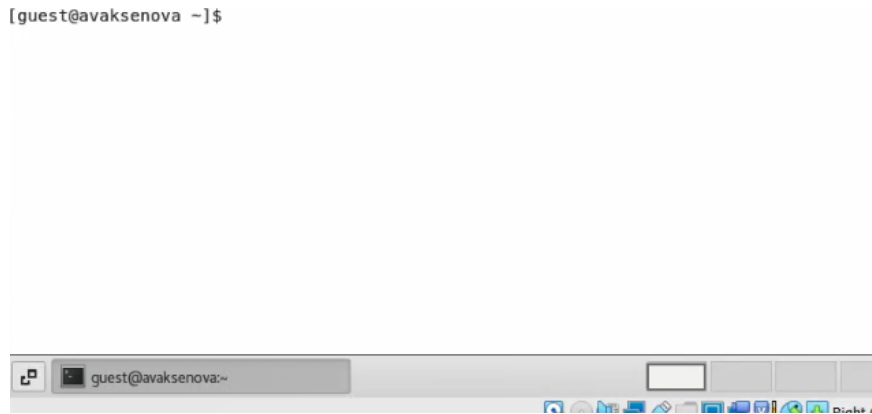


Figure 0.8: Данные в приглашении командной строки

5. Просматриваем файл `/etc/passwd` командой `cat /etc/passwd`. Найдя в нём свою учётную запись, определяем `uid` и `gid` пользователя. Значения совпадают с полученными в предыдущих пунктах. (Рис. 0.9).

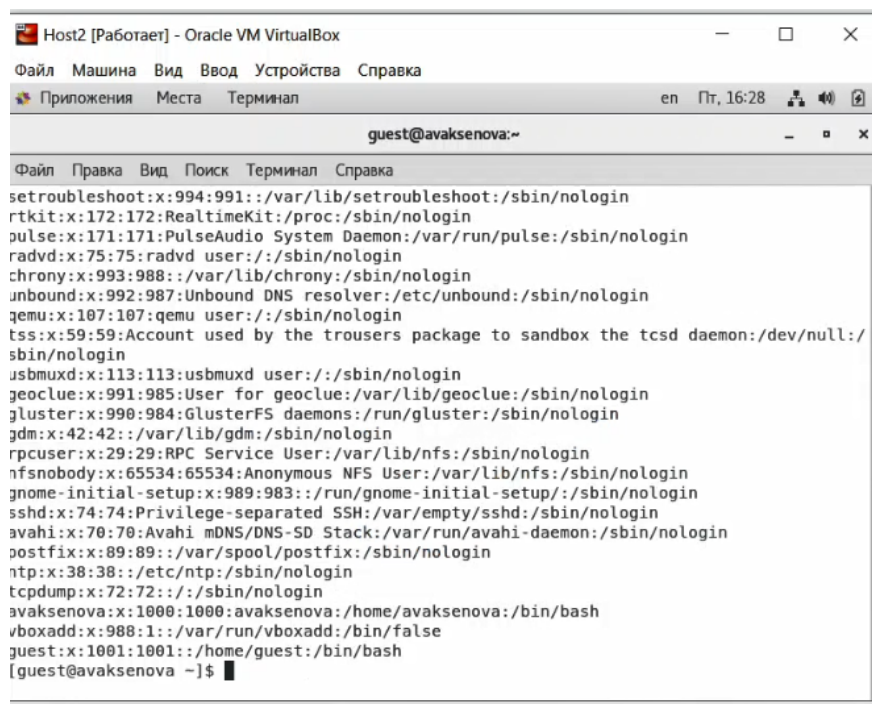


Figure 0.9: Просмотр файла `/etc/passwd` и определение `uid` и `gid` пользователя

6. Определяем существующие в системе директории командой `ls -l /home/`. В результате удалось получить список поддиректорий директории `/home`. На

имеющихся директориях (avaksenova и guest) был установлен полный набор прав. (Рис. 0.10).

```
[guest@avaksenova ~]$ ls -l /home/
итого 8
drwx-----, 15 avaksenova avaksenova 4096 окт 1 15:54 avaksenova
drwx-----, 15 guest      guest      4096 окт 1 16:24 guest
[guest@avaksenova ~]$
```

Figure 0.10: Поддиректории директории /home

7. Командой `lsattr /home` проверяем, какие расширенные атрибуты установлены на поддиректориях, находящихся в директории /home. Расширенные атрибуты директории avaksenova увидеть не удалось из-за нехватки прав доступа, а расширенные атрибуты директории guest отсутствуют. (Рис. 0.11).

```
[guest@avaksenova ~]$ lsattr /home
lsattr: Отказано в доступе While reading flags on /home/avaksenova
----- /home/guest
[guest@avaksenova ~]$
```

Figure 0.11: Проверка расширенных атрибутов

8. Создаем в домашней директории поддиректорию dir1 командой `mkdir dir1`. Определяем командами `ls -l` и `lsattr`, какие права доступа и расширенные атрибуты были выставлены на директорию dir1, а затем снимаем с директории dir1 все атрибуты командой `chmod 000 dir1` и проверяем правильность выполнения с помощью `ls -l`. (Рис. 0.12, 0.13).

```
[guest@avaksenova ~]$ mkdir dir1
[guest@avaksenova ~]$ ls -l
итого 0
drwxrwxr-x. 2 guest guest 6 окт 1 16:31 dir1
drwxr-xr-x. 2 guest guest 6 окт 1 16:23 Видео
drwxr-xr-x. 2 guest guest 6 окт 1 16:23 Документы
drwxr-xr-x. 2 guest guest 6 окт 1 16:23 Загрузки
drwxr-xr-x. 2 guest guest 6 окт 1 16:23 Изображения
drwxr-xr-x. 2 guest guest 6 окт 1 16:23 Музыка
drwxr-xr-x. 2 guest guest 6 окт 1 16:23 Общедоступные
drwxr-xr-x. 2 guest guest 6 окт 1 16:23 Рабочий стол
drwxr-xr-x. 2 guest guest 6 окт 1 16:23 Шаблоны
[guest@avaksenova ~]$
```

Figure 0.12: Создание поддиректории

```
----- /home/guest
[guest@avaksenova ~]$ lsattr
----- ./Рабочий стол
----- ./Загрузки
----- ./Шаблоны
----- ./Общедоступные
----- ./Документы
----- ./Музыка
----- ./Изображения
----- ./Видео
----- ./dir1
[guest@avaksenova ~]$ chmod 000 dir1
[guest@avaksenova ~]$ ls -l
итого 0
d[---]----- 2 guest guest 6 окт 1 16:31 dir1
drwxr-xr-x. 2 guest guest 6 окт 1 16:23 Видео
drwxr-xr-x. 2 guest guest 6 окт 1 16:23 Документы
drwxr-xr-x. 2 guest guest 6 окт 1 16:23 Загрузки
drwxr-xr-x. 2 guest guest 6 окт 1 16:23 Изображения
drwxr-xr-x. 2 guest guest 6 окт 1 16:23 Музыка
drwxr-xr-x. 2 guest guest 6 окт 1 16:23 Общедоступные
drwxr-xr-x. 2 guest guest 6 окт 1 16:23 Рабочий стол
drwxr-xr-x. 2 guest guest 6 окт 1 16:23 Шаблоны
[guest@avaksenova ~]$
```

Figure 0.13: Проверка прав доступа и расширенных атрибутов

9. Пытаемся создать в директории `dir1` файл `file1` командой `echo "test" > /home/guest/dir1/file1`. В результате получаем отказ в выполнении операции по созданию файла, поскольку в предыдущих пунктах с директории `dir1` были сняты все атрибуты. Таким образом, файл не создавался, в чем убеждаемся, применив команду `ls -l /home/guest/dir1`. (Рис. 0.14, 0.15).



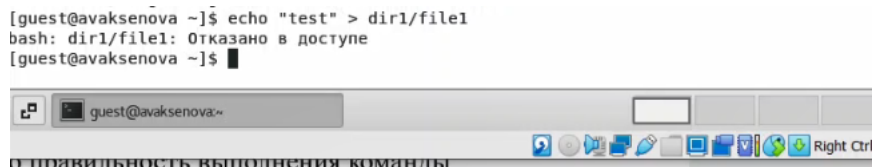


Figure 0.14: Создание файла в директории dir1

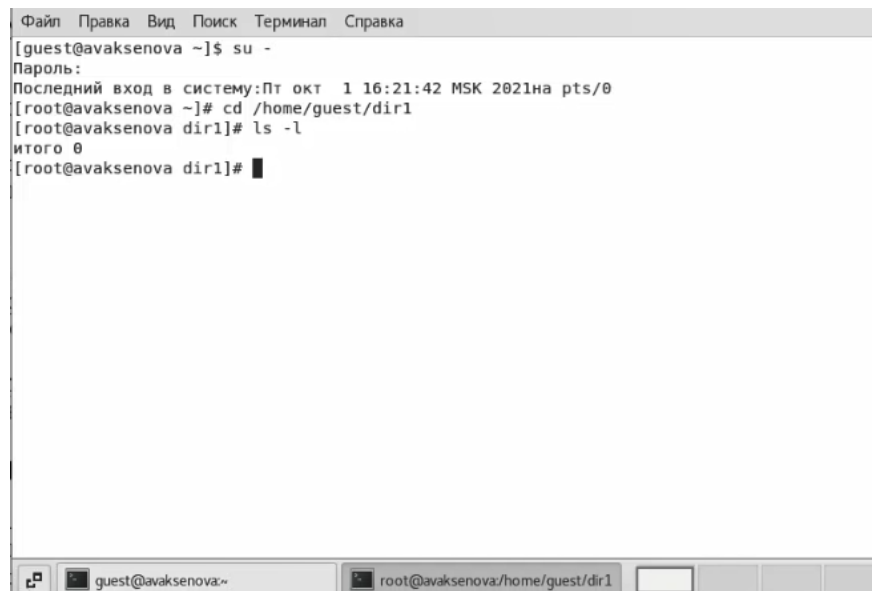
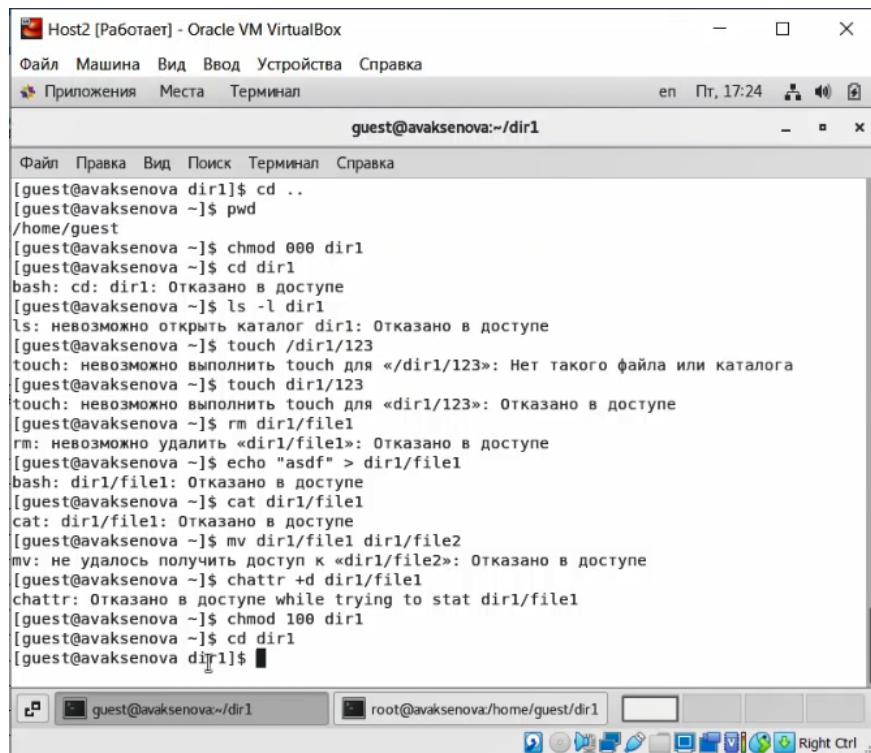


Figure 0.15: Проверка создания файла

10. Заполняем таблицу «Установленные права и разрешённые действия» , выполняя действия от имени владельца директории (файлов), определяем опытным путём, какие операции разрешены, а какие нет. Если операция разрешена, заносим в таблицу знак «+», если не разрешена, знак «-». При заполнении таблицы рассматриваем не все атрибуты файлов и директорий, а лишь «первые три»: г, w, x, для «владельца». В итоге рассматриваем 64 варианта. (Рис. 0.16, 0.17).



```
Host2 [Работае] - Oracle VM VirtualBox
Файл  Машина  Вид  Ввод  Устройства  Справка
Приложения  Места  Терминал  en  Пт, 17:24
guest@avaksenova:~/dir1
Файл  Правка  Вид  Поиск  Терминал  Справка
[guest@avaksenova dir1]$ cd ..
[guest@avaksenova ~]$ pwd
/home/guest
[guest@avaksenova ~]$ chmod 000 dir1
[guest@avaksenova ~]$ cd dir1
bash: cd: dir1: Отказано в доступе
[guest@avaksenova ~]$ ls -l dir1
ls: невозможно открыть каталог dir1: Отказано в доступе
[guest@avaksenova ~]$ touch /dir1/123
touch: невозможно выполнить touch для «/dir1/123»: Нет такого файла или каталога
[guest@avaksenova ~]$ touch dir1/123
touch: невозможно выполнить touch для «dir1/123»: Отказано в доступе
[guest@avaksenova ~]$ rm dir1/file1
rm: невозможно удалить «dir1/file1»: Отказано в доступе
[guest@avaksenova ~]$ echo "asdf" > dir1/file1
bash: dir1/file1: Отказано в доступе
[guest@avaksenova ~]$ cat dir1/file1
cat: dir1/file1: Отказано в доступе
[guest@avaksenova ~]$ mv dir1/file1 dir1/file2
mv: не удалось получить доступ к «dir1/file2»: Отказано в доступе
[guest@avaksenova ~]$ chattr +d dir1/file1
chattr: Отказано в доступе while trying to stat dir1/file1
[guest@avaksenova ~]$ chmod 100 dir1
[guest@avaksenova ~]$ cd dir1
[guest@avaksenova dir1]$
```

Figure 0.16: Процесс проверки разрешенных операций

Права директории	Права файла	Создание файла	Удаление файла	Запись в файл	Чтение файла	Смена директории	Просмотр файлов в директории	Переименование файла	Смена атрибутов файла
000	000	—	—	—	—	—	—	—	—
100	000	—	—	—	—	+	—	—	—
200	000	—	—	—	—	—	—	—	—
300	000	+	+	—	—	+	—	+	—
400	000	—	—	—	—	—	+	—	—
500	000	—	—	—	—	+	+	—	—
600	000	—	—	—	—	—	+	—	—
700	000	+	+	—	—	+	+	+	—
000	100	—	—	—	—	—	—	—	—
100	100	—	—	—	—	+	—	—	—
200	100	—	—	—	—	—	—	—	—
300	100	+	+	—	—	+	—	+	—
400	100	—	—	—	—	—	+	—	—
500	100	—	—	—	—	+	+	—	—
600	100	—	—	—	—	—	+	—	—
700	100	+	+	—	—	+	+	+	—
000	200	—	—	—	—	—	—	—	—
100	200	—	—	+	—	+	—	—	—
200	200	—	—	—	—	—	—	—	—
300	200	+	+	+	—	+	—	+	—
400	200	—	—	—	—	—	+	—	—
500	200	—	—	+	—	+	+	—	—
600	200	—	—	—	—	—	+	—	—
700	200	+	+	+	—	+	+	+	—
000	300	—	—	—	—	—	—	—	—
100	300	—	—	+	—	+	—	—	—
200	300	—	—	—	—	—	—	—	—
300	300	+	+	+	—	+	—	+	—
400	300	—	—	—	—	—	+	—	—
500	300	—	—	+	—	+	+	—	—
600	300	—	—	—	—	—	+	—	—
700	300	+	+	+	—	+	+	+	—
000	400	—	—	—	—	—	—	—	—
100	400	—	—	—	+	+	—	—	+
200	400	—	—	—	—	—	—	—	—
300	400	+	+	—	+	+	—	+	+
400	400	—	—	—	—	—	+	—	—
500	400	—	—	—	+	+	+	—	+
600	400	—	—	—	—	—	+	—	—
700	400	+	+	—	+	+	+	+	+
000	500	—	—	—	—	—	—	—	—
100	500	—	—	—	+	+	—	—	+
200	500	—	—	—	—	—	—	—	—
300	500	+	+	—	+	+	—	+	+
400	500	—	—	—	—	—	+	—	—
500	500	—	—	—	+	+	+	—	+
600	500	—	—	—	—	—	+	—	—
700	500	+	+	—	+	+	+	+	+
000	600	—	—	—	—	—	—	—	—
100	600	—	—	+	+	+	—	—	+
200	600	—	—	—	—	—	—	—	—
300	600	+	+	+	+	+	—	+	+
400	600	—	—	—	—	—	+	—	—
500	600	—	—	+	+	+	+	—	+
600	600	—	—	—	—	—	+	—	—
700	600	+	+	+	+	+	+	+	+
000	700	—	—	—	—	—	—	—	—
100	700	—	—	+	+	+	—	—	+
200	700	—	—	—	—	—	—	—	—
300	700	+	+	+	+	+	—	+	+
400	700	—	—	—	—	—	+	—	—
500	700	—	—	+	+	+	+	—	+
600	700	—	—	—	—	—	+	—	—
700	700	+	+	+	+	+	+	+	+

Figure 0.17: Заполненная таблица

11. На основании заполненной таблицы определяем те или иные минимально необходимые права для выполнения операций внутри директории `dir1`, внося данные во вторую таблицу. (Рис. 0.18).

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

Figure 0.18: Проверка минимально необходимых прав для выполнения операций внутри директории

## Выводы

В результате выполнения данной работы были приобретены практические навыки работы в консоли с атрибутами файлов, закрепление теоретических основ дискреционного разграничения доступа в современных системах с открытым кодом на базе CentOS.

## Библиографический список

1. Острейковский В. А. Информатика: учеб. для вузов / В. А. Острейковский. - 4-е изд., стер. - М.: Высш. шк., 2007. - 511 с.
2. Права в Linux [Электронный ресурс]. – Режим доступа : <https://habr.com/ru/post/469667/>, свободный. – Загл. с экрана.
3. Дискреционное разграничение доступа Linux [Электронный ресурс]. – Режим доступа : <https://debianinstall.ru/iskretsionnoe-razgranichenie-dostupa-linux/>, свободный. – Загл. с экрана.