

# Лабораторная работа № 2

Дискреционное разграничение прав в Linux. Основные атрибуты

Сухарев Кирилл

# Содержание

Цель работы	5
Условные обозначения и термины	6
Теоретические вводные данные	7
Права доступа к файлам в Linux . . . . .	7
Основные права доступа к файлам в Linux . . . . .	7
Просмотр прав доступа . . . . .	8
Техническое оснащение и выбранные методы проведения работы	10
Выполнение работы	11
Выводы	19

# List of Figures

0.1	Создание нового учётного пользователя . . . . .	11
0.2	Вход в систему . . . . .	11
0.3	Определение текущей директории . . . . .	12
0.4	Определение имени пользователя . . . . .	12
0.5	Команды id и groups . . . . .	12
0.6	Файл /etc/passwd . . . . .	13
0.7	Права на директориях системы . . . . .	14
0.8	Атрибуты поддиректорий . . . . .	14
0.9	Новая директория . . . . .	15
0.10	Снятие атрибутов . . . . .	15
0.11	Попытка создания файла . . . . .	16
0.12	Таблица «Установленные права и разрешённые действия» . . . . .	17
0.13	Таблица «Минимальные права для совершения операций» . . . . .	18

# 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 ОС Linux.

## Условные обозначения и термины

Учетная запись - хранимая в компьютерной системе совокупность данных о пользователе, необходимая для его опознавания (аутентификации) и предоставления доступа к его личным данным и настройкам.

Директория - объект в файловой системе, упрощающий организацию файлов.

uid - номер, назначенный каждому пользователю Linux. Это представление пользователя в ядре Linux.

gid - идентификационный номер основной группы пользователя.

# Теоретические вводные данные

## Права доступа к файлам в Linux

В операционной системе Linux много функций безопасности. Одна из самых важных - это система прав доступа к файлам. Linux, как последователь идеологии ядра Linux в отличие от Windows, изначально проектировался как многопользовательская система, поэтому права доступа к файлам в linux продуманы очень хорошо.

## Основные права доступа к файлам в Linux

Изначально каждый файл имел три параметра доступа. Вот они:

- Чтение - разрешает получать содержимое файла, но на запись нет. Для каталога позволяет получить список файлов и каталогов, расположенных в нем;
- Запись - разрешает записывать новые данные в файл или изменять существующие, а также позволяет создавать и изменять файлы и каталоги;
- Выполнение - вы не можете выполнить программу, если у нее нет флага выполнения. Этот атрибут устанавливается для всех программ и скриптов, именно с помощью него система может понять, что этот файл нужно запускать как программу.

Но все эти права были бы бессмысленными, если бы применялись сразу для всех пользователей. Поэтому каждый файл имеет три категории пользователей, для которых можно устанавливать различные сочетания прав доступа:

- Владелец - набор прав для владельца файла, пользователя, который его создал или сейчас установлен его владельцем. Обычно владелец имеет все права, чтение, запись и выполнение.
- Группа - любая группа пользователей, существующая в системе и привязанная к файлу. Но это может быть только одна группа и обычно это группа владельца, хотя для файла можно назначить и другую группу.
- Остальные - все пользователи, кроме владельца и пользователей, входящих в группу файла.

Именно с помощью этих наборов полномочий устанавливаются права файлов в linux. Каждый пользователь может получить полный доступ только к файлам, владельцем которых он является или к тем, доступ к которым ему разрешен. Только пользователь Root может работать со всеми файлами независимо от их набора их полномочий.

## Просмотр прав доступа

Узнать права на файл linux можно командой `ls -l`. За права файлов в linux тут отвечают черточки. Первая это тип файла, который рассмотрен в отдельной статье. Дальше же идут группы прав сначала для владельца, для группы и для всех остальных. Всего девять черточек на права и одна на тип. Рассмотрим значения черточек:

- — - нет прав, совсем;
- -x - разрешено только выполнение файла, как программы но не изменение и не чтение;
- -w- - разрешена только запись и изменение файла;
- -wx - разрешено изменение и выполнение, но в случае с каталогом, вы не можете посмотреть его содержимое;
- r- - права только на чтение;
- r-x - только чтение и выполнение, без права на запись;



- `rw-` - права на чтение и запись, но без выполнения;
- `rwX` - все права;

# Техническое оснащение и выбранные методы проведения работы

В качестве среды выполнения лабораторной работы используется менеджер виртуальных машин VirtualBox и установленная с его помощью ОС Centos 7 на базе Linux.

# Выполнение работы

1. Используя учётную запись администратора создадим учётную запись пользователя `guest` и зададим для него пароль (fig. 0.1).

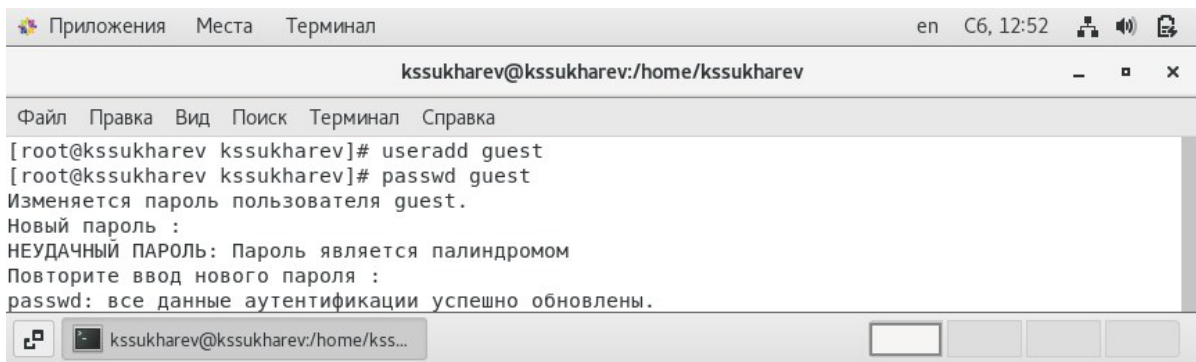


Figure 0.1: Создание нового учётного пользователя

2. Войдем в систему под пользователям `guest` (fig. 0.2).

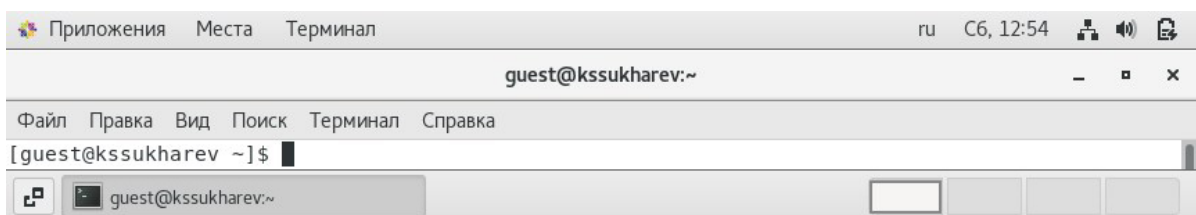


Figure 0.2: Вход в систему

3. Определим текущую директорию командой `pwd` и сравним ее с домашней директорией, которая выводится командой `echo ~`. Текущая директория является домашней (fig. 0.3).

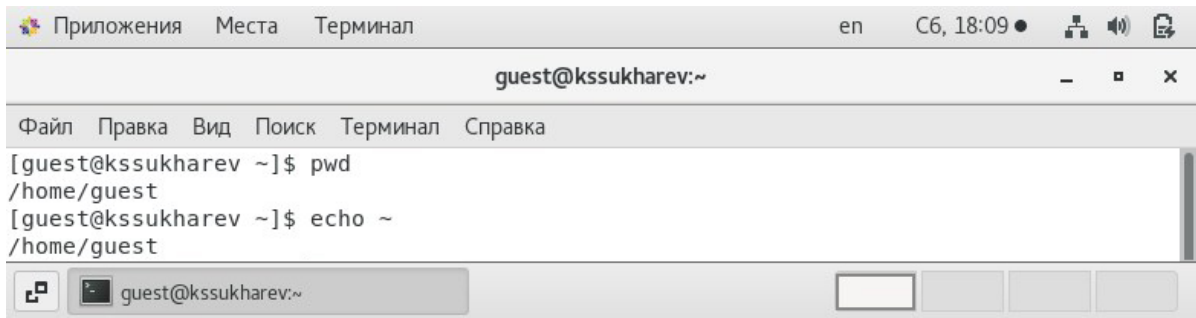


Figure 0.3: Определение текущей директории

4. Командой `whoami` уточним имя пользователя. Команда вывела `guest` (fig. 0.4).

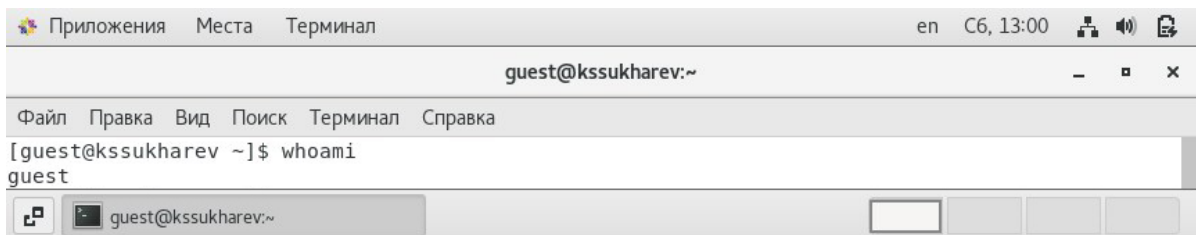


Figure 0.4: Определение имени пользователя

5. Теперь узнаем более подробные сведения о пользователе командой `id`. `uid - 1001`, `gid - 1001`. Сравним вывод с выводом команды `groups`. Команда вывела список всех групп, но поскольку новых групп не создавалось, команда вывела только `guest`. Вывод команды `id` полностью соответствует приглашению командной строки (fig. 0.5).

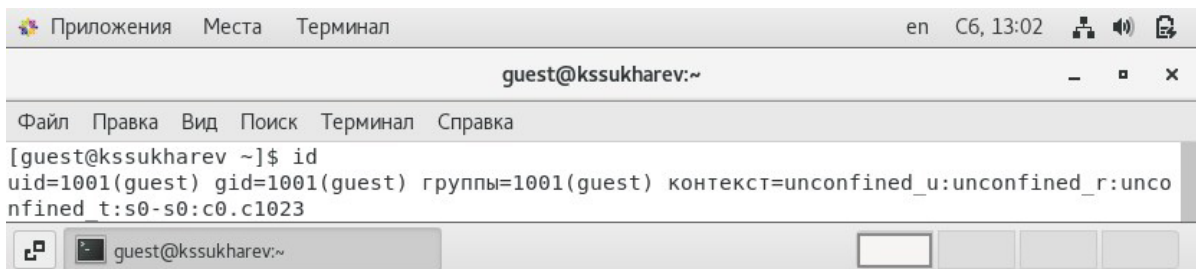


Figure 0.5: Команды `id` и `groups`

6. Просмотрим сведения о пользователе `guest` в файле `/etc/passwd` командой `cat`

/etc/passwd | grep guest. Выведенные uid и gid совпадают с определенными ранее (fig. 0.6).



```
Приложения  Места  Терминал  en  C6, 13:13  [иконки]
guest@kssukharev:~
Файл  Правка  Вид  Поиск  Терминал  Справка
[guest@kssukharev ~]$ cat /etc/passwd | grep guest
guest:x:1001:1001:./home/guest:/bin/bash
```

Figure 0.6: Файл /etc/passwd

7. Определим существующие в системе директории командой `ls -R /home/`. Доступ к поддиректориям kssukharev получить не удалось. Выведем расширенную информацию о директориях командой `ls -l /home/`. Видим, что у владельцев есть полный доступ к каталогам. У остальных же пользователей доступа нет совсем (fig. 0.7).

```
Приложения  Места  Терминал  en  C6, 18:11
guest@kssukharev:~
Файл  Правка  Вид  Поиск  Терминал  Справка
[guest@kssukharev ~]$ ls -R /home
/home:
guest  kssukharev

/home/guest:
change_mod.sh  full_test.sh  Видео  Загрузки  Музыка  Рабочий стол
dir1           mod_test.sh   Документы  Изображения  Общедоступные  Шаблоны

/home/guest/dir1:
file1

/home/guest/Видео:

/home/guest/Документы:

/home/guest/Загрузки:

/home/guest/Изображения:

/home/guest/Музыка:

/home/guest/Общедоступные:

/home/guest/Рабочий стол:

/home/guest/Шаблоны:
ls: невозможно открыть каталог /home/kssukharev: Отказано в доступе
[guest@kssukharev ~]$ ls -l /home
итого 8
drwx-----. 16 guest      guest      4096 окт  2 18:09 guest
drwx-----. 15 kssukharev kssukharev 4096 окт  2 17:58 kssukharev
[guest@kssukharev ~]$
```

Figure 0.7: Права на директориях системы

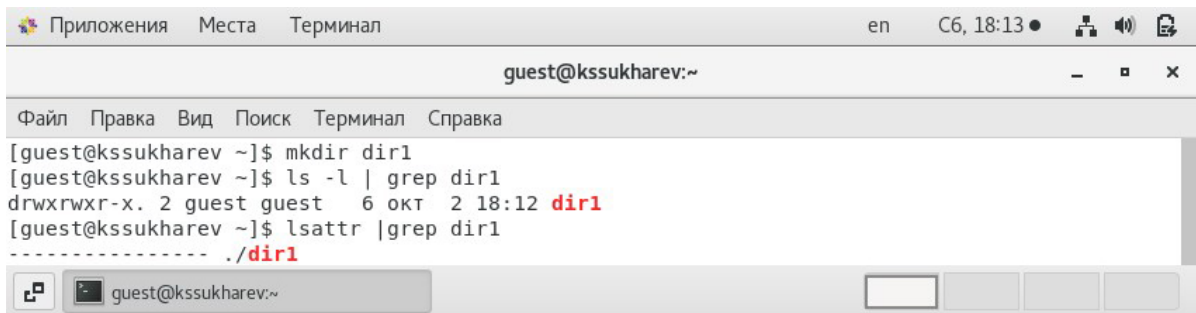
8. Проверим атрибуты на поддиректориях в директории `/home` командой `lsattr` `/home`. Нам удалось увидеть атрибуты лишь на директории `guest` (fig. 0.8).

```
Приложения  Места  Терминал  en  C6, 13:27
guest@kssukharev:~
Файл  Правка  Вид  Поиск  Терминал  Справка
[guest@kssukharev ~]$ lsattr /home
lsattr: Отказано в доступе While reading flags on /home/kssukharev
----- /home/guest
[guest@kssukharev ~]$
```

Figure 0.8: Атрибуты поддиректорий

9. Создадим в домашней директории поддиректорию `dir1` командой `mkdir dir1`.

Снова применим команды `ls -l` и `lsattr` чтобы увидеть, какие права доступа и расширенные атрибуты были выставлены на директорию `dir1`. Права доступа предоставлены пользователю `guest` и группе `guest` в полном размере, а остальным пользователям - только на чтение и выполнение. Каких-либо атрибутов на этой директории нет (fig. 0.9).



```
guest@kssukharev:~  
[guest@kssukharev ~]$ mkdir dir1  
[guest@kssukharev ~]$ ls -l | grep dir1  
drwxrwxr-x. 2 guest guest 6 окт 2 18:12 dir1  
[guest@kssukharev ~]$ lsattr |grep dir1  
----- ./dir1
```

Figure 0.9: Новая директория

10. Снимем с созданной директории все права командой `chmod 000 dir1`. Командой `ls -l` убедимся в правильности выполненных действий (fig. 0.10).



```
guest@kssukharev:~  
[guest@kssukharev ~]$ chmod 000 dir1  
[guest@kssukharev ~]$ ls -l | grep dir1  
d-----. 2 guest guest 6 окт 2 18:12 dir1
```

Figure 0.10: Снятие атрибутов

11. Попытаемся создать в директории `dir1` файл `file1` командой `echo "test" > /home/guest/dir1/file1`. Получаем закономерную ошибку, поскольку мы запретили любые взаимодействия с директорией (fig. 0.11).

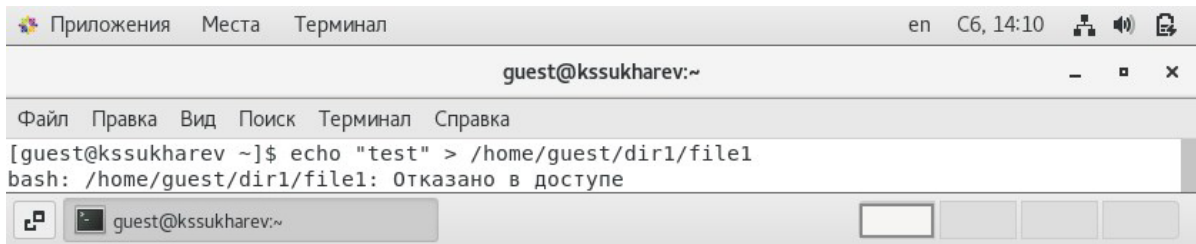


Figure 0.11: Попытка создания файла

12. Заполним таблицу «Установленные права и разрешённые действия» (fig. 0.12).

До заполнения таблицы создадим в dir1 файл file1. При заполнении таблицы будем использовать следующие команды:

- Создание файла: `echo "test" > dir1/file2`
- Удаление файла: `rm dir1/file1`
- Запись в файл: `echo "test" > dir1/file1`
- Чтение файла: `cat dir1/file1`
- Смена директории: `cd dir1`
- Просмотр файлов в директории: `ls dir1`
- Переименование файла: `mv dir1/file1 dir1/file2`
- Смена атрибутов файла: `chattr +d dir1/file1`

Процесс заполнения таблицы подробно описан в скринкасте к данной лабораторной работе.



Права директории	Права файла	Создание файла	Удаление файла	Запись в файл	Чтение файла	Смена директории	Просмотр файлов в директории	Переименования файла	Смена атрибутов файла
d--- (000)	--- (000)	-	-	-	-	-	-	-	-
d--- (000)	-x--- (100)	-	-	-	-	-	-	-	-
d--- (000)	-w--- (200)	-	-	-	-	-	-	-	-
d--- (000)	-wx--- (300)	-	-	-	-	-	-	-	-
d--- (000)	r--- (400)	-	-	-	-	-	-	-	-
d--- (000)	r-x--- (500)	-	-	-	-	-	-	-	-
d--- (000)	rw--- (600)	-	-	-	-	-	-	-	-
d--- (000)	rwX--- (700)	-	-	-	-	-	-	-	-
d-x--- (100)	--- (000)	-	-	-	-	+	-	-	-
d-x--- (100)	-x--- (100)	-	-	-	-	+	-	-	-
d-x--- (100)	-w--- (200)	-	-	+	-	+	-	-	-
d-x--- (100)	-wx--- (300)	-	-	+	-	+	-	-	-
d-x--- (100)	r--- (400)	-	-	-	+	+	-	-	+
d-x--- (100)	r-x--- (500)	-	-	-	+	+	-	-	+
d-x--- (100)	rw--- (600)	-	-	+	+	+	-	-	+
d-x--- (100)	rwX--- (700)	-	-	-	-	-	-	-	-
d-w--- (200)	--- (000)	-	-	-	-	-	-	-	-
d-w--- (200)	-x--- (100)	-	-	-	-	-	-	-	-
d-w--- (200)	-w--- (200)	-	-	-	-	-	-	-	-
d-w--- (200)	-wx--- (300)	-	-	-	-	-	-	-	-
d-w--- (200)	r--- (400)	-	-	-	-	-	-	-	-
d-w--- (200)	r-x--- (500)	-	-	-	-	-	-	-	-
d-w--- (200)	rw--- (600)	-	-	-	-	-	-	-	-
d-w--- (200)	rwX--- (700)	-	-	-	-	-	-	-	-
d-wx--- (300)	--- (000)	+	+	-	-	+	-	+	-
d-wx--- (300)	-x--- (100)	+	+	-	-	+	-	+	-
d-wx--- (300)	-w--- (200)	+	+	+	-	+	-	+	-
d-wx--- (300)	-wx--- (300)	+	+	+	-	+	-	+	-
d-wx--- (300)	r--- (400)	+	+	-	+	+	-	+	+
d-wx--- (300)	r-x--- (500)	+	+	-	+	+	-	+	+
d-wx--- (300)	rw--- (600)	+	+	+	+	+	-	+	+
d-wx--- (300)	rwX--- (700)	+	+	+	+	+	-	+	+
dr--- (400)	--- (000)	-	-	-	-	-	+	-	-
dr--- (400)	-x--- (100)	-	-	-	-	-	+	-	-
dr--- (400)	-w--- (200)	-	-	-	-	-	+	-	-
dr--- (400)	-wx--- (300)	-	-	-	-	-	+	-	-
dr--- (400)	r--- (400)	-	-	-	-	-	+	-	-
dr--- (400)	r-x--- (500)	-	-	-	-	-	+	-	-
dr--- (400)	rw--- (600)	-	-	-	-	-	+	-	-
dr--- (400)	rwX--- (700)	-	-	-	-	-	+	-	-
dr-x--- (500)	--- (000)	-	-	-	-	+	+	-	-
dr-x--- (500)	-x--- (100)	-	-	-	-	+	+	-	-
dr-x--- (500)	-w--- (200)	-	-	+	-	+	+	-	-
dr-x--- (500)	-wx--- (300)	-	-	+	-	+	+	-	-
dr-x--- (500)	r--- (400)	-	-	-	+	+	+	-	+
dr-x--- (500)	r-x--- (500)	-	-	-	+	+	+	-	+
dr-x--- (500)	rw--- (600)	-	-	+	+	+	+	-	+
dr-x--- (500)	rwX--- (700)	-	-	+	+	+	+	-	+
drw--- (600)	--- (000)	-	-	-	-	-	+	-	-
drw--- (600)	-x--- (100)	-	-	-	-	-	+	-	-
drw--- (600)	-w--- (200)	-	-	-	-	-	+	-	-
drw--- (600)	-wx--- (300)	-	-	-	-	-	+	-	-
drw--- (600)	r--- (400)	-	-	-	-	-	+	-	-
drw--- (600)	r-x--- (500)	-	-	-	-	-	+	-	-
drw--- (600)	rw--- (600)	-	-	-	-	-	+	-	-
drw--- (600)	rwX--- (700)	-	-	-	-	-	+	-	-
drwx--- (700)	--- (000)	+	+	-	-	+	+	+	-
drwx--- (700)	-x--- (100)	+	+	-	-	+	+	+	-
drwx--- (700)	-w--- (200)	+	+	+	-	+	+	+	-
drwx--- (700)	-wx--- (300)	+	+	+	-	+	+	+	-
drwx--- (700)	r--- (400)	+	+	-	+	+	+	+	+
drwx--- (700)	r-x--- (500)	+	+	-	+	+	+	+	+
drwx--- (700)	rw--- (600)	+	+	+	+	+	+	+	+
drwx--- (700)	rwX--- (700)	+	+	+	+	+	+	+	+

Figure 0.12: Таблица «Установленные права и разрешённые действия»

13. По результатам таблицы «Установленные права и разрешённые действия» заполним таблицу «Минимальные права для совершения операций» (fig. 0.13).

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-w- --- --- (200)	--- --- --- (000)
Удаление поддиректории	d-w- --- --- (200)	--- --- --- (000)

Figure 0.13: Таблица «Минимальные права для совершения операций»

## Выводы

Права доступа используются для управления возможностями различных групп пользователей системы по отношению к директориям и файлам.