Лабораторная работа №2

Дискреционное разграничение прав в Linux. Основные атрибуты

Смородова Дарья Владимировна

2022 Sep 16th

Содержание

# 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.

# Теоретическое введение

В операционной системе Linux есть много отличных функций безопасности, но она 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.

Изначально каждый файл имеет три параметра доступа:

* 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. [[1]](#footnote-21)

# Выполнение лабораторной работы [[2]](#footnote-23)

1. В установленной при выполнении предыдущей лабораторной работы операционной системе создим учётную запись пользователя guest (используя учётную запись администратора) и зададим пароль (рис. 1):

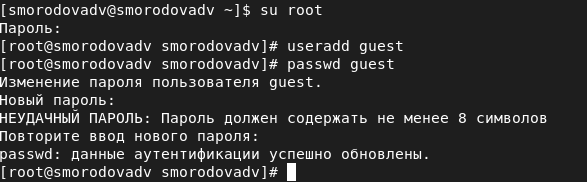


Figure 1: Создание пользователя guest и задание ему пароля

1. Войдем в систему от имени пользователя guest (рис. 2):

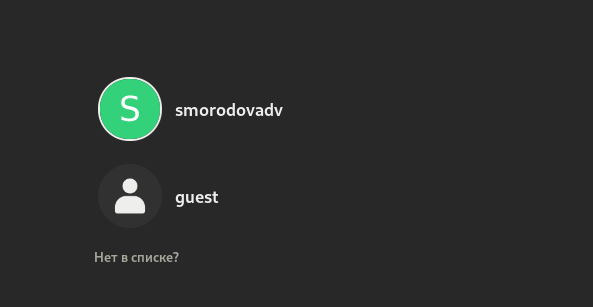


Figure 2: Вход в систему под guest

1. Определим директорию, в которой мы находимся при помощи команды pwd, и переход в домашнюю директорию (рис. 3):



Figure 3: Проверка директории и переход в домашнюю директорию

1. Уточним имя нашего пользователя при помощи команды whoami (рис. 4):

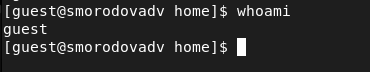


Figure 4: Уточнение имени пользователя

1. Уточним имя нашего пользователя, его группу, а также группы, куда он входит, командой id. Выведенные значения uid, gid и др. запомним. Выполним команду groups. Полученные значения совпадают с тем, что выдала id. Полученная информация об имени пользователя частично совпадает с данными, выводимыми в приглашении командной строки, но является более подробной (рис. 5):

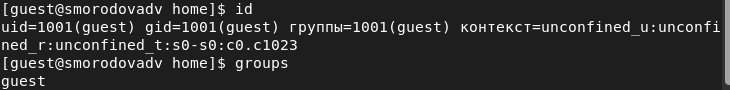


Figure 5: Проверка id

1. Посмторим файл /etc/passwd (рис. 6):

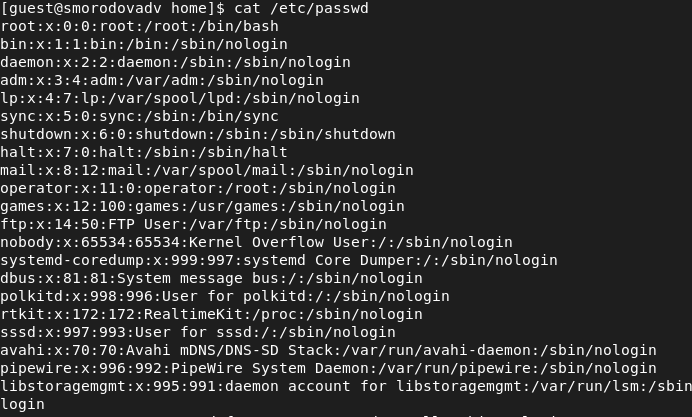


Figure 6: Файл /etc/passwd

1. Найдем в нем нашу учётную запись при помощи команды cat /etc/passwd | grep guest. Uid пользователя: 1001, gid пользователя: 1001. Эти значения совпадают с полученными в ранее. (рис. 7):

Figure 7: Учётная запись guest в файле /etc/passwd

Figure 7: Учётная запись guest в файле /etc/passwd

1. При помощи команды ls -l /home/ определим существующие в системе директории. Владельцы директорий имеют на них полные права, а группы и другие пользователи не имеют никаких прав на эти директории (рис. 8):

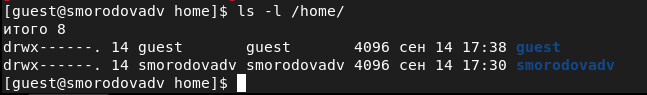


Figure 8: Существующие директории

1. Проверим, какие расширенные атрибуты установлены на поддиректориях, находящихся в директории /home при помощи команды lsattr /home. Нам удалось посмотреть только расширенные атрибуты директории guest, а расширенные атрибуты директорий других пользователей нам не доступны 9):

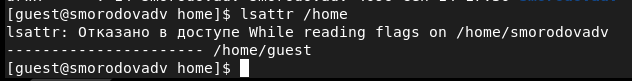


Figure 9: Расширенные атрибуты поддиректорий

1. Создадим в домашней директории поддиректорию dir1 при помощи команды mkdir dir1. Определим командами ls -l и lsattr, какие права доступа и расширенные атрибуты были выставлены на директорию dir1 (рис. 10 - 11):

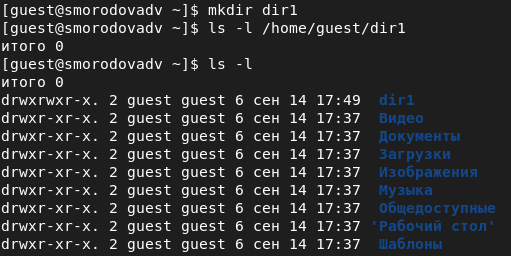


Figure 10: Создание поддиректории dir1 и команда ls -l

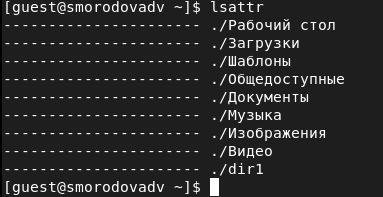


Figure 11: Просмотр расширенных атрибутов поддиректорий

1. Снимем с директории dir1 все атрибуты при помощи команды chmod 000 dir1, и проверим с её помощью правильность выполнения команды ls -l (рис. 12):

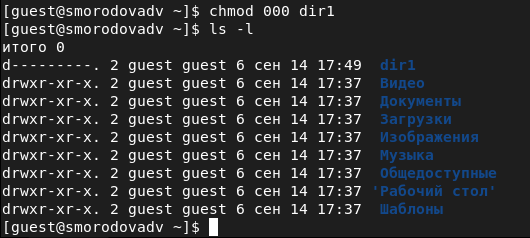


Figure 12: Снятие с директории dir1 всех атрибуов

1. Попытаемся создать в директории dir1 файл file1 при помощи команды echo “test” > /home/guest/dir1/file1. При попытке создания был получен отказ, так как до этого я сняла с директории все атрибуты. Сообщение об ошибке никак не отразилось на создании файла, потому что он не был создан (рис. 13):

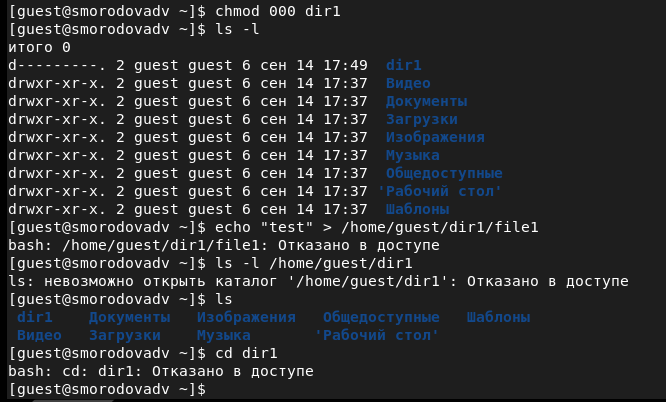


Figure 13: Попытка создать файл file1 в директории dir1

1. Заполним таблицу «Установленные права и разрешённые действия», выполняя действия от имени владельца директории (файлов), определив опытным путём, какие операции разрешены, а какие нет. Если операция разрешена, занесем в таблицу знак «+», если не разрешена – знак «-».
2. Порядок команд, при помощи которых проводилась проверка (рис. 14):

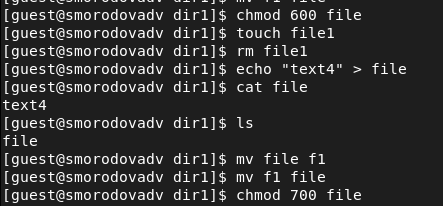


Figure 14: Проверка установленных прав и разрешенных действий

1. Получившаяся таблица «Установленные права и разрешённые действия»(рис. 15 - 16):

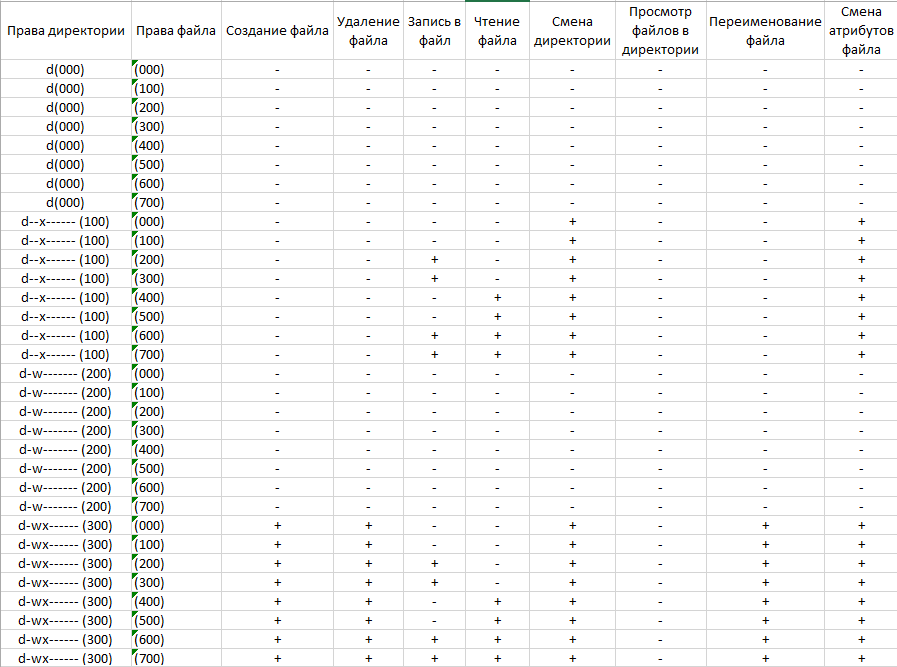


Figure 15: Установленные права и разрешённые действия 1

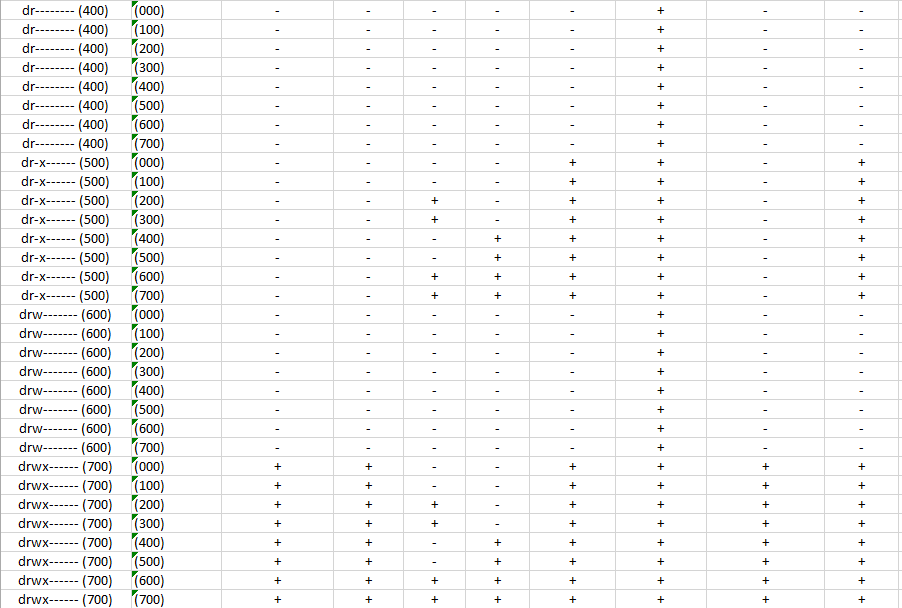


Figure 16: Установленные права и разрешённые действия 2

1. На основании заполненной таблицы определим те или иные минимально необходимые права для выполнения операций внутри директории dir1, заполним таблицу «Минимальные права для совершения операций» (рис. 17):

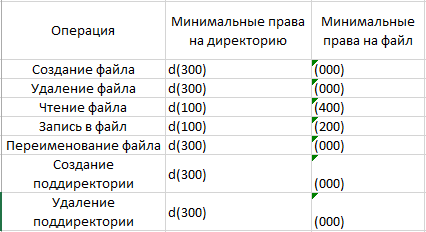


Figure 17: Минимальные права для совершения операций

1. Для проверки минимальных прав на директорию и минимальных прав на файл при создании и удалении поддиректорий выполним следующие команды (рис. 18):

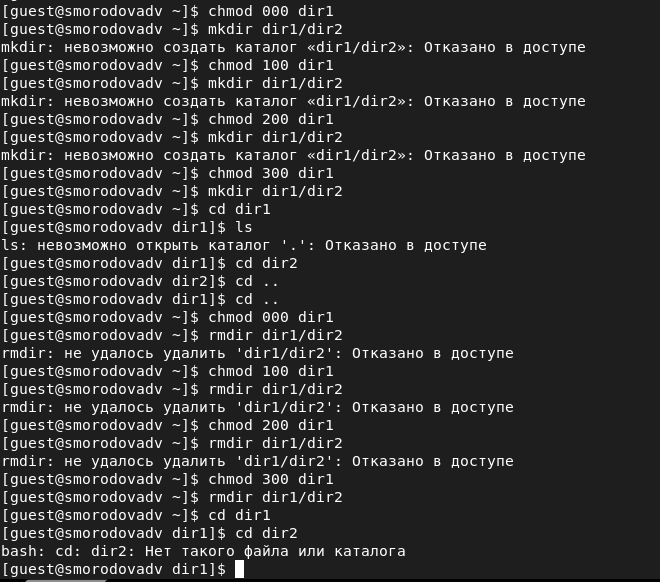


Figure 18: Создание и удаление поддиректорий

# Выводы

В ходе данной лабораторной работы, мы получили практические навыки работы в консоли с атрибутами файлов, закрепили теоретические основы дискреционного разграничения доступа в современных системах с открытым кодом на базе ОС Linux.

# Список литературы

1. [Методические материалы к лабораторной работе, представленные на сайте “ТУИС РУДН”](https://esystem.rudn.ru/)
2. [Права доступа к файлам в Linux](https://losst.ru/prava-dostupa-k-fajlam-v-linux)

1. Права доступа к файлам в Linux [↑](#footnote-ref-21)
2. Методические материалы к лабораторной работе [↑](#footnote-ref-23)