Лабораторная работа № 4

Дискреционное разграничение прав в Linux. Расширенные атрибуты

Беличева Дарья Михайловна

Содержание

# 1 Цель работы

Получение практических навыков работы в консоли с расширенными атрибутами файлов.

# 2 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 (от англ, discretion — чье-либо усмотрение) — 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 [1].

Основная команда для работы с правами в Linux: chmod. Есть три основных вида прав:

* r – чтение;
* w – запись;
* x – выполнение;
* s – выполнение от имени суперпользователя (дополнительный);

Также есть три категории пользователей, для которых вы можете установить эти права на файл linux:

* u – владелец файла;
* g – группа файла;
* o – все остальные пользователи.

# 3 Выполнение лабораторной работы

От имени пользователя guest определим расширенные атрибуты файла /home/guest/dir1/file1 командой lsattr /home/guest/dir1/file1. Пока что расширенных атрибутов нет. Установим командой chmod 600 file1 на файл file1 права, разрешающие чтение и запись для владельца файла. Попробуем установить на файл /home/guest/dir1/file1 расширенный атрибут a от имени пользователя guest: chattr +a /home/guest/dir1/file1 В ответ получим отказ от выполнения операции (рис. 1).

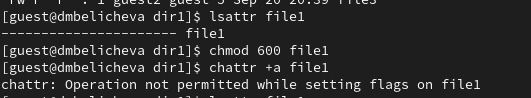


Рис. 1: Просмотр и попытка изменения расширенных атрибутов файла от имени guest

Зайдем на другую консоль с правами администратора. Попробуем установить расширенный атрибут a на файл /home/guest/dir1/file1 от имени суперпользователя: chattr +a /home/guest/dir1/file1. Теперь у нас получилось это сделать (рис. 2).

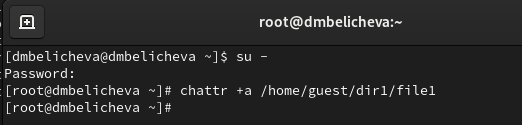


Рис. 2: Изменение расширенных атрибутов файла от имени суперпользователя

От пользователя guest проверим правильность установления атрибута: lsattr /home/guest/dir1/file1. Увидим, что атрибут a действительно установился. Выполним дозапись в файл file1 слова «test» командой echo "test" >> /home/guest/dir1/file1. После этого выполним чтение файла file1 командой cat /home/guest/dir1/file1. Слово test было успешно записано в file1. Попробуем удалить файл file1 либо стереть имеющуюся в нём информацию командой echo "abcd" > /home/guest/dirl/file1, переименовать файл и изменить права файла, но никакие из этих действий у нас не выполняется (доступ запрещен) (рис. 3).

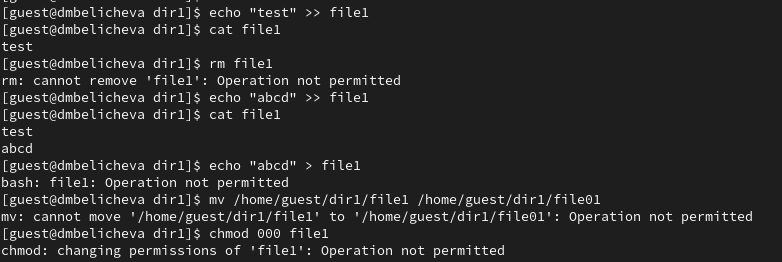


Рис. 3: Проверка прав с расширенным атрибутом a

Снимем расширенный атрибут a с файла /home/guest/dirl/file1 от имени суперпользователя командой chattr -a /home/guest/dir1/file1. Повторим операции, которые ранее не удавалось выполнить. Теперь все операции выполняются успешно (рис. 4).

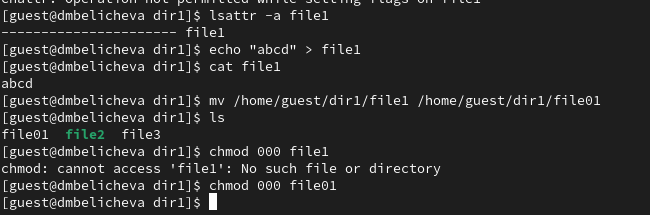


Рис. 4: Проверка прав без расширенных атрибутов

Повторим действия по шагам, заменив атрибут «a» атрибутом «i». Теперь нам запрещены любые действия, в том числе и дозапись информации в файл (рис. 5).

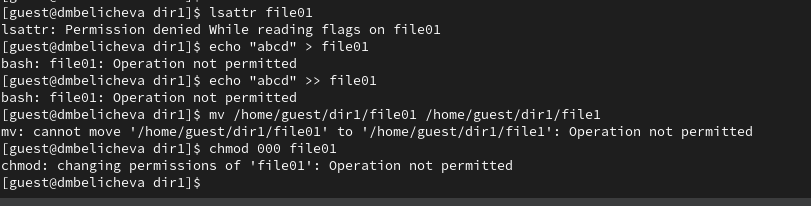


Рис. 5: Проверка прав с расширенным атрибутом i

# 4 Выводы

В процессе выполнения данной лабораторной работы я получила практические навыки работы в консоли с расширенными атрибутами файлов.

# Список литературы

1. Дискреционное разграничение доступа Linux [Электронный ресурс]. 2023. URL: <https://debianinstall.ru/diskretsionnoe-razgranichenie-dostupa-linux/>.