Лабораторная работа № 4

Дискреционное разграничение прав в Linux. Расширенные атрибуты.

Алибаева Данагуль НБибд-01-18

Содержание

# Цель работы

Получение практических навыков работы в консоли с расширенными атрибутами файлов.

# Задание

1. От имени пользователя guest определить расширенные атрибуты файла /home/guest/dir1/file1
2. Установить на файл file1 права, разрешающие чтение и запись для владельца файла.
3. Попробовать установить на файл /home/guest/dir1/file1 расширенный атрибут a от имени пользователя guest. В ответ должны получить отказ от выполнения операции.
4. Зайти на третью консоль с правами администратора либо повысить свои права. Попробовать установить расширенный атрибут a на файл /home/guest/dir1/file1 от имени суперпользователя.
5. От пользователя guest проверить правильность установления атрибута.
6. Выполнить дозапись в файл file1 слова «test». После этого выполнить чтение файла file1. Убедиться, что слово test было успешно записано в file1.
7. Попробовать удалить файл file1 либо стереть имеющуюся в нём информацию. Попробовать переименовать файл.
8. Попробовать установить на файл file1 права, например, запрещающие чтение и запись для владельца файла. Проверить удалось ли успешно выполнить указанные команды.
9. Снять расширенный атрибут a с файла /home/guest/dirl/file1 от имени суперпользователя. Повторить операции, которые ранее не удавалось выполнить. Наблюдения занести в отчёт.
10. Повторить действия по шагам, заменив атрибут «a» атрибутом «i». Проверить удалось ли дозаписать информацию в файл. Наблюдения занести в отчёт.

# Теоретическое введение

Атрибуты файла - это набор из девяти основных битов. Определяющих какие из пользователей обладают правами на чтение, запись, а также запуск файлов для выполнения. Данный набор формирует код, называемый режимом доступа к файлу/каталогу. Первые три бита определяют права доступа для владельца. Следующие — для группы пользователей, к которой относится файл и последние три бита — права доступа для всех остальных пользователей в системе. [1]

Существует также ещё четыре дополнительных бита. Которые определяют тип самого файла и задаются непосредственно при создании файла. С помощью команды chmod можно менять основные (и некоторые дополнительные) биты режима доступа. Сделать это может только владелец файла или суперпользователь. Просматривать атрибуты (в том числе и режимы доступа) позволяет команда ls. Таким образом, характер поведения ФС, а также распределение доступа и управление им полностью определяется атрибутами файлов. Которые хранит сама ФС — это самодостаточный и универсальный подход. [1]

Для записи кода режима доступа используется восьмеричная запись чисел. Как уже было отмечено, код доступа содержит три «триады» битов — для пользователя, группы и всех остальных, именно в таком порядке. Битам из первой триады соответствуют значения в восьмеричной записи 400, 200 и 100. Для второй триады (т. е. для группы) — 40, 20 и 10. Наконец, для третьей (все остальные) — 4, 2 и 1. В свою очередь, первому биту в каждой триаде соответствует доступ на чтение (r — «read»). Второму — на запись (w — «write») и третьему — на выполнение, т. е. x — «execute». [1]

Установка бита чтения (r) в одной из триад (или во всех) задаёт право открывать данный файл для чтения соответствующим категориям пользователей. Наличие бита записи (w) позволяет изменять файл. При этом возможно его удаление и/или переименование файла, но только в том случае, если заданы соответствующие биты для его родительского каталога, поскольку именно в его записях хранятся имена файлов. [1]

# Выполнение лабораторной работы

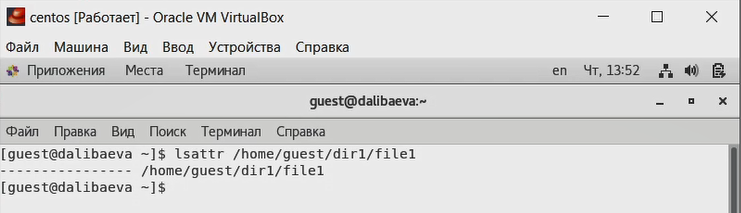
1.Лабораторная работа выполнялась дома со следующими характеристиками техники:

– Intel(R) Core(TM) i5-8300H CPU @ 2.30GHz, 2304 МГц, ядер: 4, логических процессоров: 8

– ОС Майкрософт Windows 10 Pro

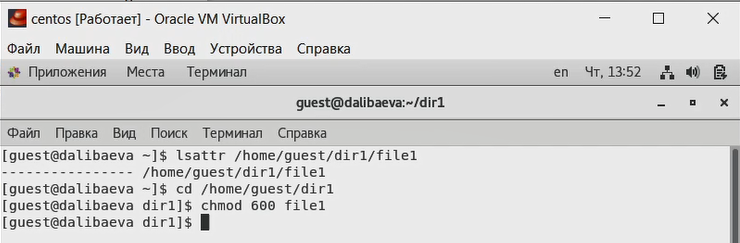
– VirtualBox верс. 6.1.26

1. От имени пользователя guest определила расширенные атрибуты файла /home/guest/dir1/file1 командой lsattr /home/guest/dir1/file1 (рис 1.1). У файла file1 нет ни одного расширенного атрибута.



1.1. Определение расширенного атрибута файла

1. Установила командой chmod 600 file1 на файл file1 права, разрешающие чтение и запись для владельца файла (рис 1.2).



1.2. Установка на файл прав чтения и записи

1. Попробовала установить на файл /home/guest/dir1/file1 расширенный атрибут a от имени пользователя guest: chattr +a /home/guest/dir1/file1 (рис 1.3). В ответ получила отказ от выполнения операции.



1.3. Попытка установления на файл расширенных атрибутов

1. Повысила свои права с помощью команды su. Попробовала установить расширенный атрибут a на файл /home/guest/dir1/file1 от имени суперпользователя: chattr +a /home/guest/dir1/file1 (рис 1.4). Команда выполнилась успешно.



1.4. Установка расширенного атрибута а на файл

1. От пользователя guest проверила правильность установления атрибута: lsattr /home/guest/dir1/file1. (рис 1.5). Команда выполнилась успешно.



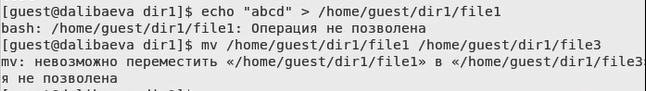
1.5. Успешное выполнение команды chattr +a file1

1. Выполнила дозапись в файл file1 слова «test» командой echo “test” /home/guest/dir1/file1 (рис 1.6). После этого выполнила чтение файла file1 командой cat /home/guest/dir1/file1 (рис 1.6).



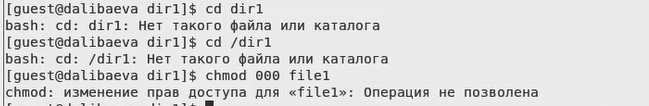
1.6. Чтение и дозапись в файл

1. Попробовала стереть имеющуюся в файле информацию командой echo “abcd” > /home/guest/dirl/file1 (рис 1.7). Операция была не позволена. Попробовала переименовать файл (рис 1.7). Операция также была не позволена.



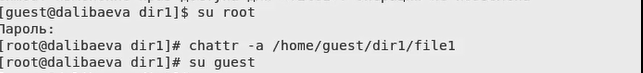
1.7. Попытка прочитать файл и стереть его содержимое

1. Попробовала с помощью команды chmod 000 file1 установить на файл file1 права, например, запрещающие чтение и запись для владельца файла. Успешно выполнить указанную команду не удалось (рис 1.8).



1.8. Попытка установки прав на файл

1. Сняла расширенный атрибут a с файла /home/guest/dirl/file1 от имени суперпользователя командой chattr -a /home/guest/dir1/file1 (рис 1.9). Повторила операции, которые ранее не удавалось выполнить. На этот раз получилось стереть имеющуюся в файле информацию, переименовать файл и выполнить чтение файла. (рис 1.10).

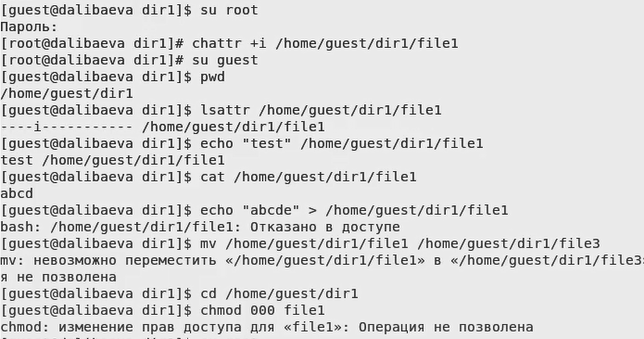


1.9. Снятие расширенного атрибута а с файла



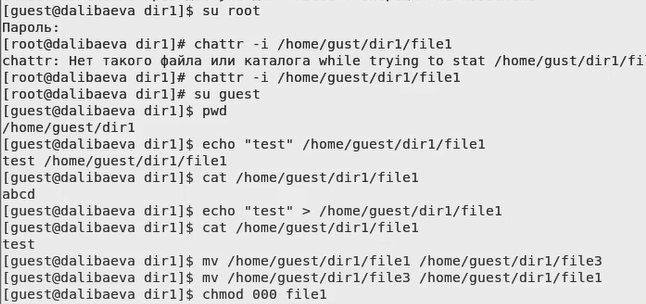
1.10. Успешное стирание, переименование и чтение файла

1. Повторила действия по шагам, заменив атрибут «a» атрибутом «i». Проверила правильность установления атрибута командой lsattr /home/guest/dir1/file1(рис 1.11). Дозапись в файл file1 слова «test» командой echo “test” /home/guest/dir1/file1 сделать удалось (рис 1.11). Выполнить чтение файла file1 командой cat /home/guest/dir1/file1 удалось (рис 1.11). А вот стереть имеющуюся в файле информацию командой echo “abcd” > /home/guest/dirl/file1 не удалось, как и переименование файла (рис 1.11). Установить на файл file1 права с помощью команды chmod 000 file1 также не удалось (рис 1.11).



1.11. Действия с расширенным атрибутом i на файле

1. После снятия расширенного атрибута i с файла /home/guest/dirl/file1 от имени суперпользователя командой chattr -a /home/guest/dir1/file1 все ранее осуществляемые действия успешно выполнились. Дозапись в файл слова «test», чтение файла file1, переименование файла и установка на файл file1 прав стало доступным (рис 1.12).



1.12. Действия без расширенного атрибута i на файле

# Выводы

В результате выполнения работы я повысила свои навыки использования интерфейса командой строки (CLI), познакомилась на примерах с тем, как используются основные и расширенные атрибуты при разграничении доступа. Имела возможность связать теорию дискреционного разделения доступа (дискреционная политика безопасности) с её реализацией на практике в ОС Linux. Опробовала действие на практике расширенных атрибутов «а» и «i».

# Список литературы

1. Атрибуты файлов в Linux // URL: https://itproffi.ru/atributy-fajlov-v-linux/ (дата обращения: 01.10.2021).