Лабораторная работа №2

Дискреционное разграничение прав в Linux. Основные атрибуты

Парфенова Елизавета Евгеньевна

Содержание

Список иллюстраций

Список таблиц

1 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.

2 Теоретическое введение

Права доступа в операционной системе Linux представляют собой ключевой элемент безопасности, определяющий, какой доступ имеют пользователи и программы к файлам и каталогам. Чтобы посмотреть права пользователя в Linux, необходимо воспользоваться следующей командой [1]:

ls - l

Изменить права доступа можно с помощью команды *chmod*. Основной синтаксис команды *chmod* выглядит следующим образом [2]:

chmod [опции]

Есть 3 вида разрешений. Они определяют права пользователя на 3 действия: чтение, запись и выполнение. В Linux эти действия обозначаются вот так:

- r — read (чтение) — право просматривать содержимое файла;
- w — write (запись) — право изменять содержимое файла;
- x — execute (выполнение) — право запускать файл, если это программа или скрипт.

У каждого файла есть 3 группы пользователей, для которых можно устанавливать права доступа.

- owner (владелец) — отдельный человек, который владеет файлом. Обычно это тот, кто создал файл, но владельцем можно сделать и кого-то другого.
- group (группа) — пользователи с общими заданными правами.
- others (другие) — все остальные пользователи, не относящиеся к группе и не являющиеся владельцами [3].

Существуют два основных способа записи прав доступа: символьный и числовой формат. Символьный формат использует читаемые буквы и символы для представления прав доступа, в то время как числовой формат использует числа в восьмеричной системе [1].

В табл. 2.1 приведено краткое описание стандартных каталогов Unix.

Таблица 2.1: Формат записи прав доступа системы GNU Linux

Права доступа	Символьный формат	Числовой формат
Чтение	r	4
Запись	w	2
Выполнение	x	1
Нет доступа	-	0

3 Выполнение лабораторной работы

Заходим в виртуальную машину Rocky Linux и с помощью команды *useradd guest* создаем новую учетную запись пользователя guest. Для этого также командой “su” заходим в систему от имени администратора, введя пароль. (рис. 3.1).

Рис. 3.1: Создание новой учетной записи guest

Далее задаем пароль командой *passwd guest*. Он должен быть не менее 8 символов. (рис. 3.2).

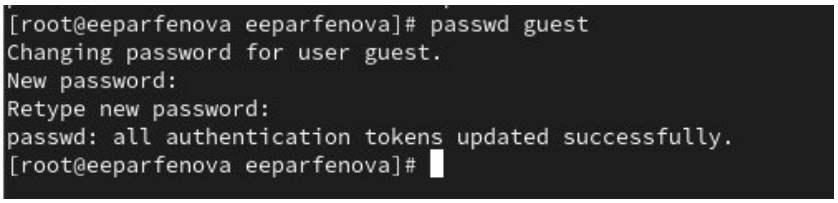

Рис. 3.2: Задание пароля для новой учетной записи

Затем выходим из системы с помощью Log out и заходим в новую учетную запись, вводя пароль. В консоли определяем директорию, в которой находимся, командой *pwd*. (рис. 3.3) Видим, что эта директория является нашей домашней директорией. Также проверяем это с помощью команды *cd ~* (переход в домашню. директорию) и видим, что наше местоположение не меняется.

Рис. 3.3: Вывод команды pwd

Командой *whoami* уточняем имя пользователя. Видим, что оно guest, как мы и задавали. (рис. 3.4)

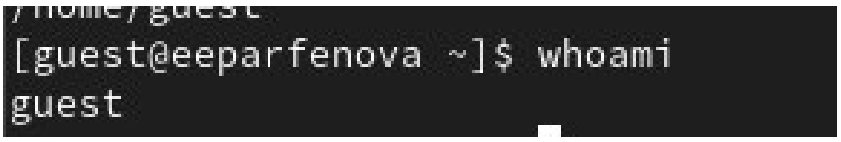

Рис. 3.4: Вывод команды whoami

В выводе команды *id* видим, что имя нашего пользователя guest, его id = 1001 (uid), а также он входит в группу guest с таким же id (group). Далее введем команду *groups* и увидим наше имя пользователя - значит наш пользователь входит только в 1 группу, как и было указано выше. (рис. 3.5). Также видим, что имя пользователя в выводе *id* совпадает с приглашением командной строки.

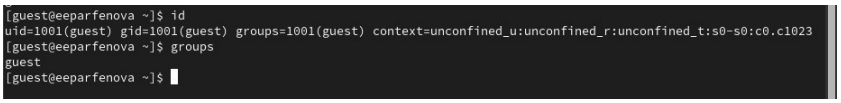

Рис. 3.5: Вывод команды id и groups

Командой *cat /etc/passwd* просмотрим файл /etc/passwd и найдем там свою учетную запись самой последней строкой. (рис. 3.6). Видим, что имя пользователя, а также uid и gid, равные 1001, совпадают с нашими прошлыми данными. Проверим, правильно ли мы все нашли, с помощью команды *cat /etc/passwd | grep guest* (рис. 3.7). Все оказалось верно

Рис. 3.6: Содержимое файла /etc/passwd

Рис. 3.7: Вывод команды cat /etc/passwd | grep guest

Командой *ls -l /home/* определим все существующие в домашнем каталоге директории. Поддиректориями являются guest и eeparfenova (рис. 3.8). Владелец директорий имеет право на чтение, запись и выполнение (изменение), остальные не могут ничего.

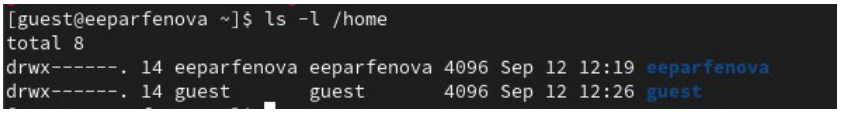

Рис. 3.8: Директории домашнего каталога

Командой *lsattr /home* проверим расширенные атрибуты, установленные на поддиректориях /home. Видим, что на поддиректории guest не установлено никакхи расширенных атрибутов, а информацию об этом на директории другоих пользователей (eeparfenova) мы увидеть не можем (рис. 3.9).

Рис. 3.9: Расширенные атрибуты поддиректорий

Командой *mkdir dir1* создаем новую директорию dir1. Командами *ls -l u lssttr* видим, что владелец имеет право на чтение, запись и изменения (то есть все права), а для остальных доступно только чтение и не доступно внесение изменений. А по выводу второй команды видим, что никаких расширенных атрибутов не установлено (рис. 3.10).

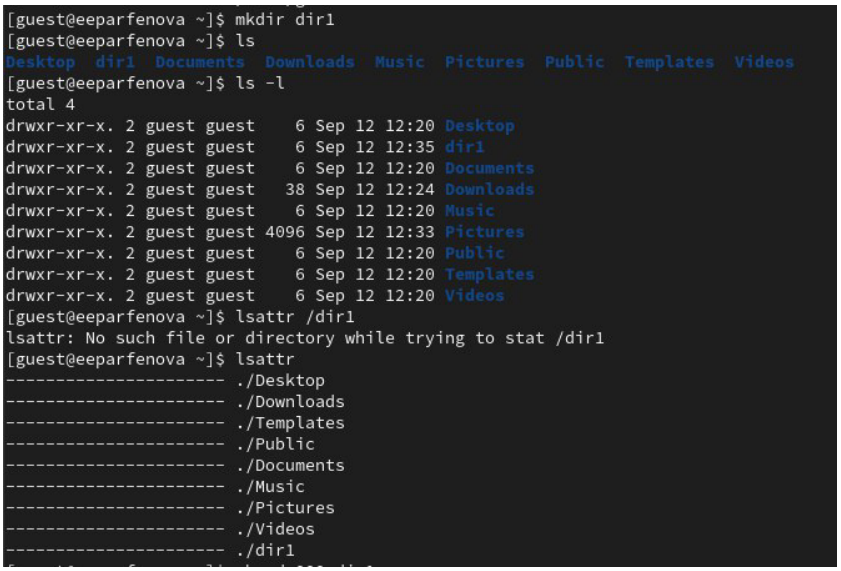

Рис. 3.10: Информация о правах и расширенных атрибутах dir1

Командой *chmod 000 dir1* снимаем все атрибуты с директории dir1 и проверяем, получилось ли это командой *ls -l* (рис. 3.11). Все прошло корреткно

Рис. 3.11: Снятие всех атрибутов с команды dir1

Командой *echo “test” > /home/guest/dir1/file1* создаем в директории dir1 файл file1, однако получаем отказ в операции, так как с директории dir1 сняты абсолютно все атрибуты и мы не имеем никаких прав в ней. Отказ в операции не создал файл внутри дирректории (так как и на это у нас нет парва), однако проверить это с помощью *ls -l /home/guest/dir1* мы также не можем, так как атрибут на просмотр дирректории также снят (у нас нет на это прав) (рис. 3.12)

Рис. 3.12: Попытка создания файла

Далее заполним таблицу «Установленные права и разрешённые действия» оптыным путем. Получившаяся таблица 3.1

Таблица 3.1: Установленные права и разрешённые действия

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

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

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

Права директории	Права файла	Создание файла	Удаление файла	Запись в файл	Чтение файла	Смена директории	Просмотр файлов в директории	Переименование файла	Смена атрибутов файла
d(500)	(700)	-	-	-	-	+	+	-	+
d(600)	(700)	-	-	-	-	-	+	-	-
d(700)	(700)	+	+	+	+	+	+	+	+

На основе анализа уже заполненной таблицы заполним следующую таблицу, которая указывает на миниммальные права для файла и директории для того или иного действия. Получившаяся таблица 3.2

Таблица 3.2: Минимальные права для совершения операций

Операция	Минимальные права на директорию	Минимальные права на файл
Создание файла	d(300)	(000)
Удаление файла	d(300)	(000)
Чтение файла	d(100)	(400)
Запись в файл	d(100)	(200)

Операция	Минимальные права на директорию	Минимальные права на файл
Переименование файла	d(300)	(000)
Создание поддиректории	d(300)	(000)
Удаление поддиректории	d(300)	(000)

4 Выводы

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. Как дать права пользователю Linux: инструкция [Электронный ресурс]. ООО «ТАЙМВЭБ.КЛАУД»., 2024. URL: https://timeweb.cloud/tutorials/linux/kak-dat-prava-polzovatelyu-linux.
2. Что делает команда chmod и как ее использовать в Linux [Электронный ресурс]. ОО «Селектел», 2024. URL: https://selectel.ru/blog/tutorials/what-the-chmod-command-does-and-how-to-use-it-in-linux/.
3. Права доступа в Linux [Электронный ресурс]. CodeChick.io, 2024. URL: https://codechick.io/tutorials/unix-linux/unix-linux-permissions.