Отчёт по лабораторной работе №3

Основы информационной безопасности

Надежда Александровна Рогожина

Содержание

# 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 OC Linux.

# 2 Задание

1. В установленной при выполнении предыдущей лабораторной работы операционной системе создайте учётную запись пользователя guest (использую учётную запись администратора): useradd guest
2. Задайте пароль для пользователя guest (использую учётную запись администратора): passwd guest
3. Аналогично создайте второго пользователя guest2.
4. Добавьте пользователя guest2 в группу guest: gpasswd -a guest2 guest
5. Осуществите вход в систему от двух пользователей на двух разных консолях: guest на первой консоли и guest2 на второй консоли.
6. Для обоих пользователей командой pwd определите директорию, в которой вы находитесь. Сравните её с приглашениями командной строки.
7. Уточните имя вашего пользователя, его группу, кто входит в неё и к каким группам принадлежит он сам. Определите командами groups guest и groups guest2, в какие группы входят пользователи guest и guest2. Сравните вывод команды groups с выводом команд id -Gn и id -G.
8. Сравните полученную информацию с содержимым файла /etc/group. Просмотрите файл командой cat /etc/group.
9. От имени пользователя guest2 выполните регистрацию пользователя guest2 в группе guest командой newgrp guest.
10. От имени пользователя guest измените права директории /home/guest, разрешив все действия для пользователей группы: chmod g+rwx /home/guest
11. От имени пользователя guest снимите с директории /home/guest/dir1 все атрибуты командой chmod 000 dirl и проверьте правильность снятия атрибутов. Меняя атрибуты у директории dir1 и файла file1 от имени пользователя guest и делая проверку от пользователя guest2, заполните табл. 3.1, определив опытным путём, какие операции разрешены, а какие нет. Если операция разрешена, занесите в таблицу знак «+», если не разрешена, знак «-».

Сравните табл. 2.1 (из лабораторной работы № 2) и табл. 3.1. На основании заполненной таблицы определите те или иные минимально необходимые права для выполнения пользователем guest2 операций внутри директории dir1 и заполните табл. 3.2

# 3 Теоретическое введение

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.

**Как узнать разрешения файла?**

Чтобы посмотреть права доступа к файлу, нужно вызвать команду ls с опцией -l. Эта опция отвечает за вывода списка в длинном формате.

ls -l <путь>

user@bash: ls -l /home/karpaff/linuxtutorialwork/chick.png  
-rwxr----x 1 harry users 2.7K Jan 4 07:32 /home/karpaff/linuxtutorialwork/chick.png  
user@bash:

В приведенном выше примере первые 10 символов вывода показывают разрешения файла.

* Первый символ определяет тип файла. Если первый символ — прочерк -, то это обычный файл. Если первый символ d, то это каталог.
* Следующие 3 символа показывают разрешения для владельца. Буква означает наличие разрешения, а прочерк - — его отсутствие. В нашем примере у владельца есть все разрешения (чтение, запись и выполнение).
* Следующие 3 символа показывают разрешения для группы. В этом примере у членов группы есть разрешение на чтение, но нет разрешений на запись и выполнение. Обратите внимание, порядок записи разрешений всегда такой: чтение, запись, выполнение.
* Последние 3 символа показывают разрешения для всех остальных пользователей. В этом примере у них есть только разрешение на выполнение.

**Как изменить права доступа?**

Для изменения прав доступа к файлу или каталога используется команда chmod (от англ. change mode). Эта команда меняет биты режима файла — если совсем просто, это индикатор разрешений.

chmod [разрешение] [путь]

Аргументы команды chmod, отвечающие за разрешение, состоят из 3 компонентов:

* Для кого мы меняем разрешение? Обозначается первыми буквами слов: [ugoa] — user (пользователь, он же владелец), group (группа), others (другие), all (все).
* Мы предоставляет или отзываем разрешения? Обозначается плюсом +, если предоставляем, минусом -, если отзываем.
* Какое разрешение мы хотим изменить? Чтение (r), запись (w), выполнение (x).

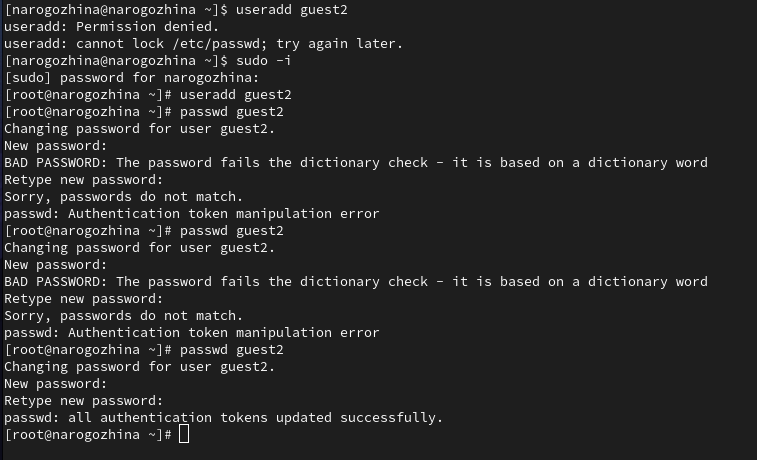
В табл. [[1](#tbl:std-dir)] приведено краткое описание основных команд, использованных в лабораторной работе.

Table 1: Описание некоторых команд Linux

| Имя каталога | Описание каталога |
| --- | --- |
| cd | Смена директории |
| clear | Очищение окна терминала |
| id | Отображение системного идентификатора указанного пользователя |
| whoami | Имя пользователя |
| group | Группа, к которой принадлежит пользователь |
| sudo | Права суперпользователя |
| cat | Отражение содержимого текстового файла в терминале |
| chmod | Изменение прав доступа к файлу |
| ls | Просмотр содержимого директории |
| touch | Создание файла |
| mkdir | Создание директории |
| rm | Удаление файла |
| mv | Перемещение/переименование файла |

# 4 Выполнение лабораторной работы

В первую очередь добавить нового пользователя и установим для него пароль (рис.[??]):



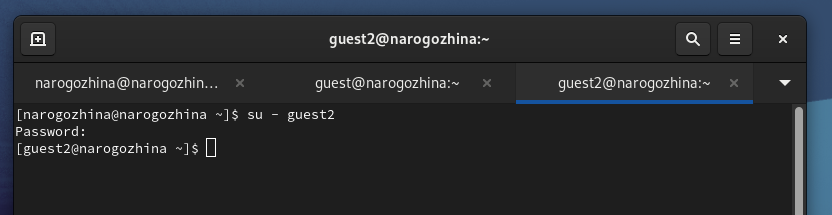
Добавление нового пользователя

Добавляем guest2 в группу guest (рис.[??]):

Добавление нового пользователя в группу

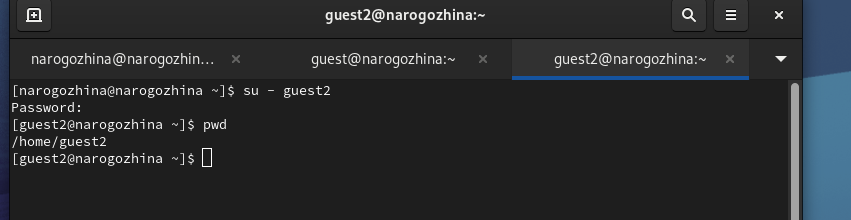
Добавление нового пользователя в группу

С основной консоли войдем в основной аккаунт, а с двух других - в guest и guest2 (рис.[??]):

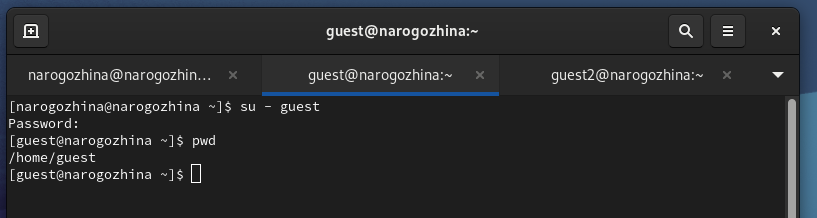


Добавление нового пользователя

Проверка путей (рис.[??], [??]):

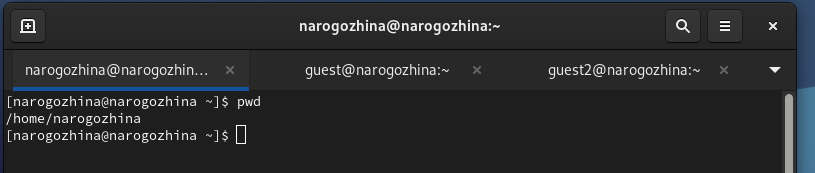


guest2:pwd



guest:pwd

Путь домашнего каталога от основного пользователя (рис.[??]):



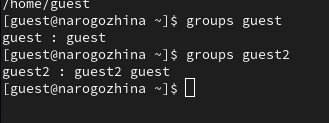
narogozhina:pwd

Groups guest (рис.[??]):

groups guest

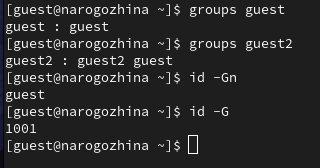
groups guest

Groups guest2 (рис.[??]):



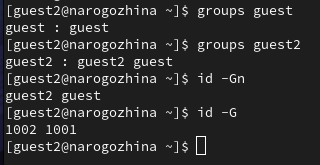
groups guest2

groups VS id -G for guest (рис.[??]):

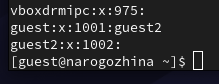


groups|id

groups VS id -G for guest2 (рис.[??], [??]):



groups|id



id

Добавление guest2 в группу guest (рис.[??]):

newgrp guest

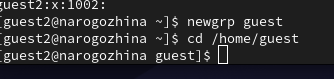
newgrp guest

От имени guest2 изменим права доступа группы (рис.[??]):

chmod

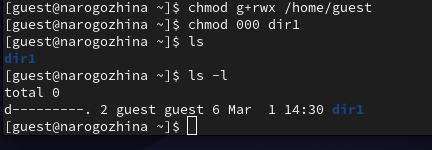
chmod

Проверка (рис.[??]):



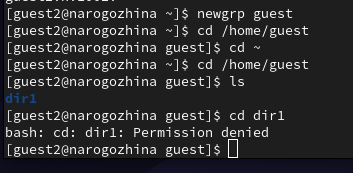
chmod check

Все получилось. Далее убираем все гостевые права с этой директории (рис.[??]):

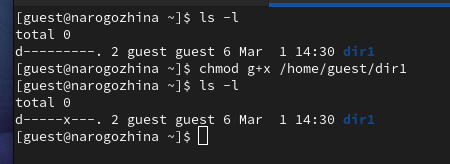


chmod 000

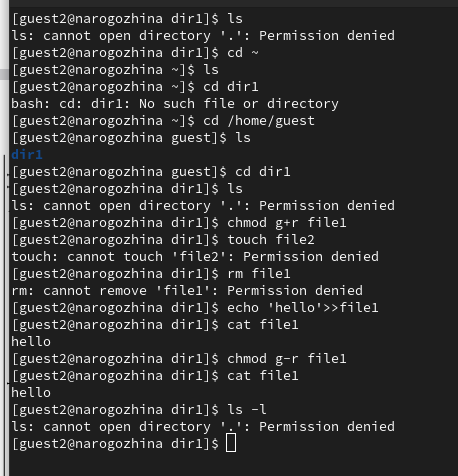
Далее я приведу скриншоты, по которым я заполняла таблицу 2.1 (рис. [??], рис. [??], рис. [??], рис.[??], рис.[??], рис.[??], рис.[??], рис.[??], рис.[??], рис.[??], рис.[??], рис.[??], рис.[??]):



d———



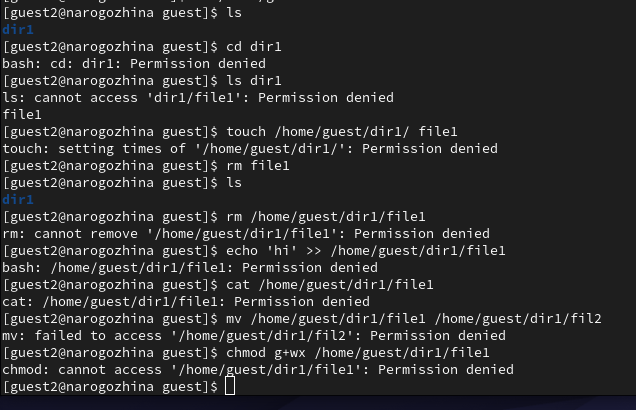
d—–x—



d—–x—

d—r—–

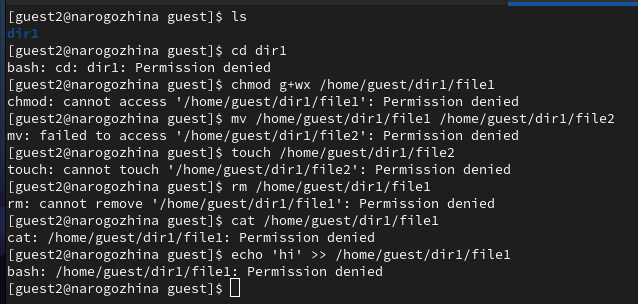
d—r—–



d—r—–

d—-w—-

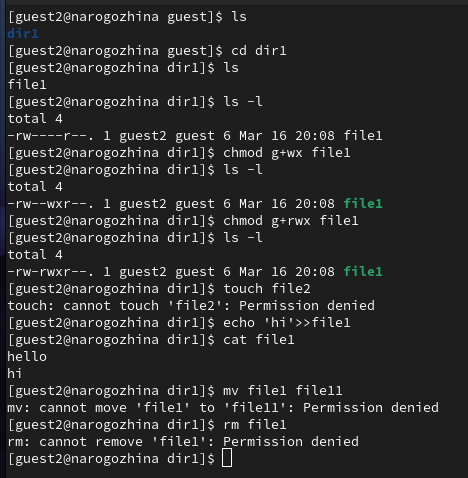
d—-w—-



d—-w—-

d—r-x—

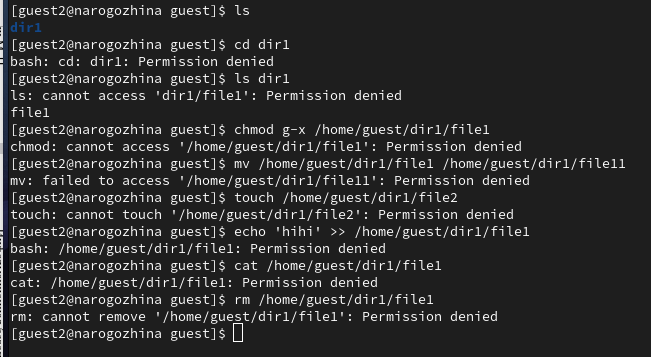
d—r-x—



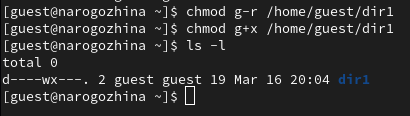
d—r-x—

d—rw—-

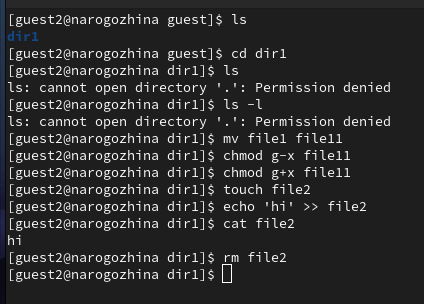
d—rw—-



d—rw—-



d—-wx—



d—-wx—

После проведенных тестов, итоговая таблица[[2](#tbl:std-dir1)] заполнена:

Table 2: 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--x------ | 001 | - | - | - | - | + | - | - | + |
| dr-------- | 100 | - | - | - | - | - | + | - | - |
| d-w------- | 010 | + | - | - | - | - | + | - | - |
| drw------- | 110 | - | - | - | - | - | + | - | - |
| dr-x------ | 101 | - | - | + | + | + | + | - | + |
| d-wx------ | 011 | + | + | + | + | + | - | + | + |
| drwx------ | 111 | + | + | + | + | + | + | + | + |

После создания предыдущей таблицы, можем заполнить минимальные права для совершения операций : |||| | Операция | Минимальные права на директорию | Минимальные права на файл | | Создание файла | x | w | | Удаление файла | x | w | | Чтение файла | r | r | | Запись в файл | wx | w | | Переименование файла | wx | x | | Создание поддиректории | wx | wx | | Удаление поддиректории | wx | wx | ||||

# 5 Выводы

В ходе выполнения работы мы получили практические навыки работы в консоли с атрибутами файлов и закрепили теоретические основы разграничения доступа в современных системах с открытым кодом на базе OC Linux.

# Список литературы