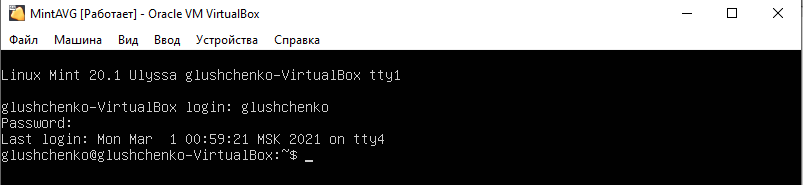
## Управление правами доступа

### Цель работы

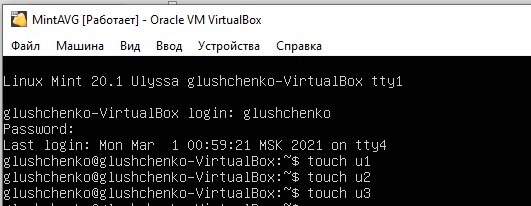
Получить навыки по управлению правами доступа к файловым ресурсам системы, освоить основные команды - *chmod, chown, chgrp*.

### Задания к работе

1. Войти в систему с собственной учетной записью



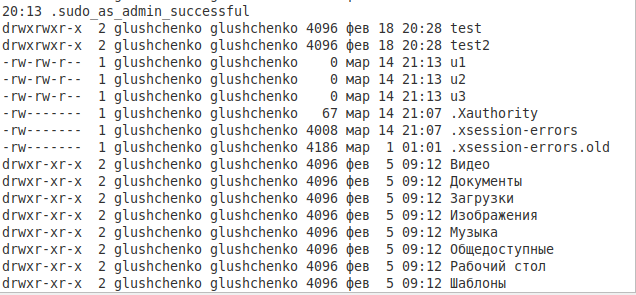
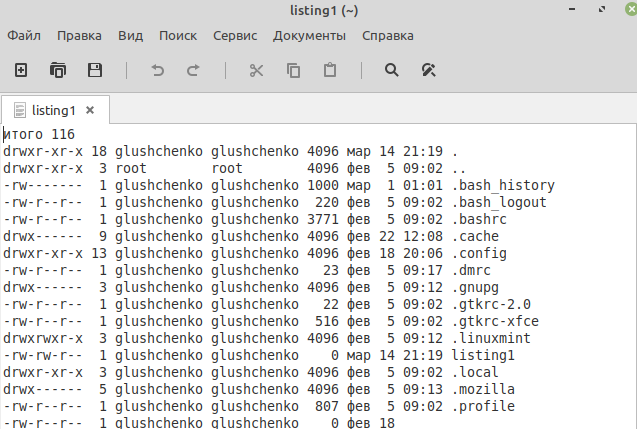
1. Создать в домашнем каталоге 2-3 файла произвольного содержания (имена файлов - u1, u2, u3).



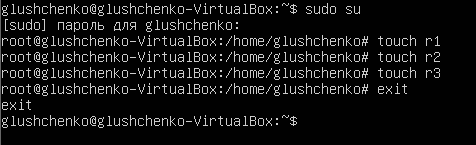
1. Получить развернутый список файлов домашнего каталога и сохранить его в файле listing1



1. Просмотреть файл listing1, обратив внимание на поля прав доступа, владельца и группы



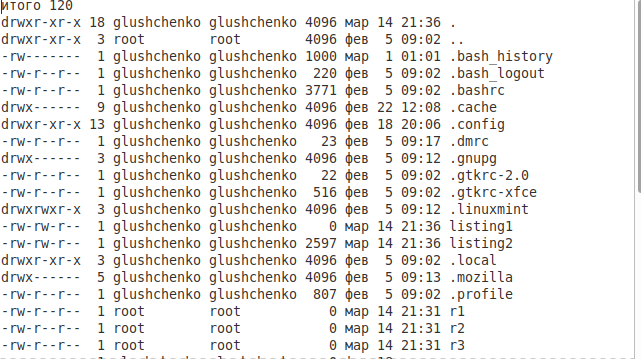
1. Повторить п. 2 от имени пользователя root в новом сеансе или по команде su (имена файлов - r1, r2, r3). Завершить сеанс root



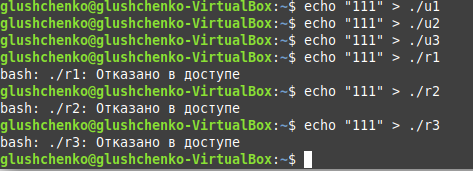
1. Повторить п.3, результат дописать в файл listing1



1. Открыть файл listing1 и сравнить права доступа для файлов, созданных от вашего имени и от имени суперпользователя



1. Изменить содержимое файлов, созданных вами и суперпользователем. Сохранить изменения



1. Открыть сеанс root



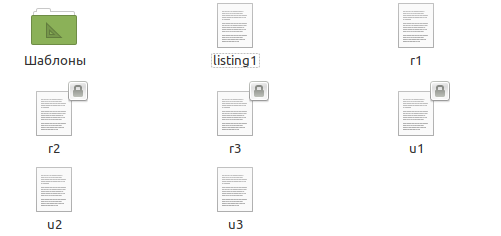
1. Перейти в каталог /home/ваша\_учетная\_запись



1. Изменить права доступа к файлам u1 и r1 следующим образом:
   * u1: запретить запись для владельца и группы
   * r1: разрешить запись для всех



1. Переключиться в сеанс пользователя и изменить содержимое файлов u1 и r1. Сохранить изменения

1. Перейти в сеанс root и изменить владельца файлов u1 и u2 на root, а группу - на stud

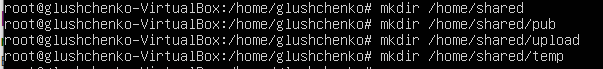
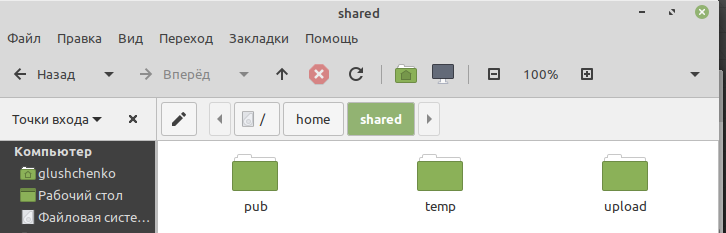




1. Из tty1 попробовать изменить файл u2



1. Из tty2 создать каталоги /home/shared, home/shared/pub, /home/shared/upload, /home/shared/temp. Установить на них следующие права:

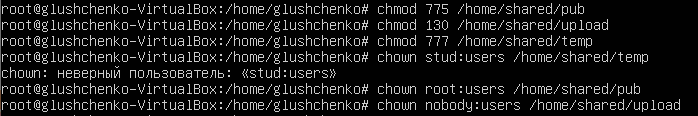
|  |  |  |  |
| --- | --- | --- | --- |
| каталог | владелец | группа | права |
| Pub | root | users | 775 |
| upload | nobody | users | 130 |
| Temp | stud | users | 777 |

777 – чтение, запись, исполнение

775 – чтение, исполнение

Каждая цифра обозначает определенную группу:

* **Первая цифра** - права для владельца,
* **вторая цифра** - права для группы,
* **третья цифра** - права для остальных.

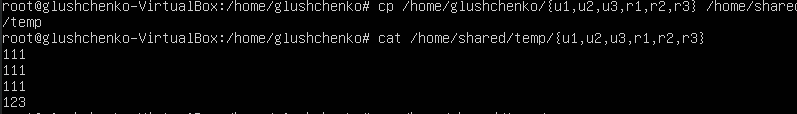
 

1. Выполнить копирование, чтение, удаление файлов u1, u2, u3, r1, r2, r3 в каталоги, созданные в п. 17 из сеансов root, stud и вашего. Сравнить и проанализировать результаты.

Пользователь root pub:

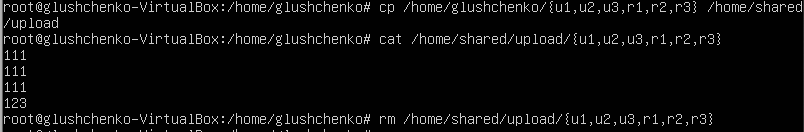


Пользователь root temp:

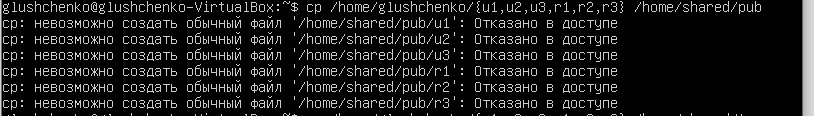




Пользователь root upload:



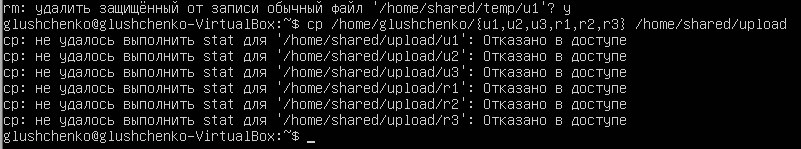
Пользователь glushchenko pub:



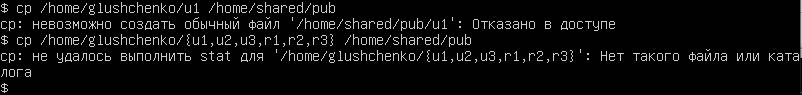
Пользователь glushchenko temp:



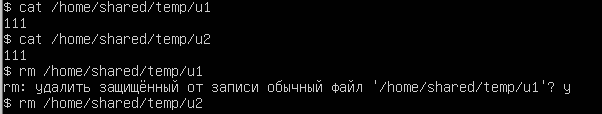
Пользователь glushchenko upload:



Пользователь stud pub:



Пользователь stud temp:

Пользователь stud upload:



Выводы: из суперпользователя копируются, читаются и удаляются все файлы из всех каталогов. Из моего пользователя читается только temp, так как никто не обладает правами к каталогу upload, а к каталогу root обладает правами только суперпользователь. Из stud читается, копируется, удаляется только temp, так как этому пользователю принадлежат права к этому каталогу.

### Контрольные вопросы

1. Зачем у файла нужны атрибуты доступа?

Для того, чтобы различные пользователи или группы пользователей имели определенные права к файлу. Это необходимо для защиты данных от вмешательства.

1. Какие три категории пользователей знают права доступа каждого конкретного файла?

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.

1. Для чего в UNIX-подобных системах используются пользовательские группы?

Для того, чтобы пользователи, находящиеся в разных группах, имели различные права доступа к файлам, тем самым позволяя нескольким людям работать в одной системе и не допустить изменения файлов друг друга или наоборот иметь доступ к файлам в своей группе.

1. Какие действия с файлами регламентируются правами 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.

1. На какой системе счисления основывается числовое представление прав доступа?

На восьмеричной.



1. Чем отличается назначение прав доступа к простому файлу и к директории?

В символах и записи не отличается ничем, однако цифры или символы для chmod будут означать разные действия. Соотношение отображено в вопросе 5.

1. Кто может менять права доступа определенного файла?

Либо владелец, либо суперпользователь. Изменить владельца каталога или файла может только суперпользователь.

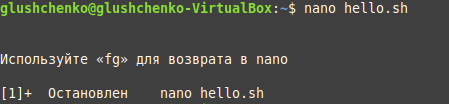
1. Каково назначение специальных битов прав доступа?

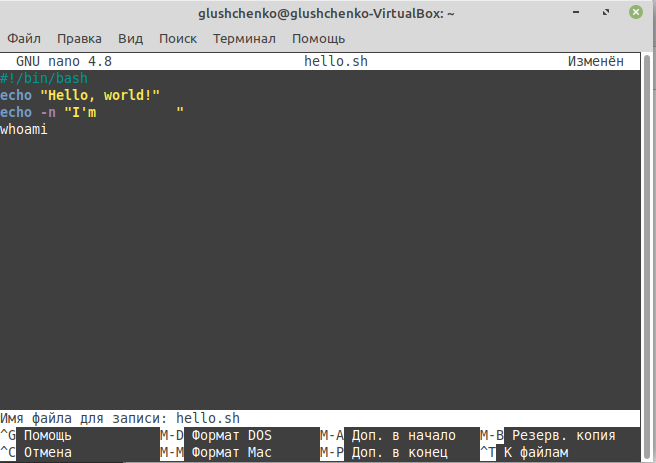
Права на файлы - это правила, у правил есть исключения в виде дополнительных атрибутов. Биты SUID и SGID. Для чего они требуются?  
Если установлены права доступа SUID и файл исполняемый, то при запуске на выполнение файл получает не права запустившего его, а права владельца файла. Если установлены права доступа SGID, то это равнозначно установке бита SUID, только наследуются права не владельца файла, а группы владельца.

### Дополнительные задания

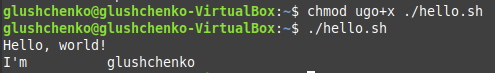
1. В текущей папке создать файл hello следующего содержания

#!/bin/sh  
echo Hello, World!  
echo -n "I'm "  
whoami





1. Выполнить следующие действия и проанализировать результаты:
   * набрать в командной строке имя файла hello и нажать Enter 
   * набрать в командной строке sh hello и нажать Enter 
   * установить для файла hello права на исполнение (x), ввести имя файла в командной строке (./hello) и нажать Enter



Выводы:

С помощью редактора nano создаем скрипт и внутри пишем специальную строку #!/bin/bash и последовательность команд, после чего записываем это в файл, сохраняем и закрываем. При попытке запустить, это сделать невозможно, так как необходимо сначала указать, что данный файл может быть исполняемым, т.е. передать права на исполнение, после чего мы можем запустить наш скрипт, и он сработает.