# Лабораторная работа №1. Избирательная модель управления доступом на примере ОС Linux.

## 1. Создание пользователей

### 1.1. Создать 3 новых пользвателя (user1, user2, user3). Создать для них домашнии директории и установить shell-оболочку по умолчанию (/bin/sh)

- adduser -h /home/tedmeadow -s /bin/sh tedmeadow
- adduser -h /home/ted -s /bin/sh ted
- adduser -h /home/meadow -s /bin/sh meadow

### 1.2. Проверить, что пользователи и домашние директории были созданы. Проверить группы, которым принадлежат пользователи.

- cd /home

  > meadow ted tedmeadow

- cat /etc/passwd

  > tedmeadow:x:1000:1000:Linux User,,,:/home/tedmeadow:/bin/sh ted:x:1001:1001:Linux User,,,:/home/ted:/bin/sh meadow:x:1002:1002:Linux User,,,:/home/meadow:/bin/sh

- cat /etc/group

  > tedmeadow:x:1000: ted:x:1001: meadow:x:1002:

- Что значат цифры в passwd?
  - **username:x:uid:gid:pd:/home/folder:/bin/sh**
    - username - имя пользователя
    - x - указатель, где хранится пароль(В нашем случае указатель на то, что пароль хранится в зашифрованном виде в файле /etc/shadow)
    - uid - Идентификатор пользователя
    - gid - Идентификатор группы по умолчанию
    - pd - Описания пользователя(Всякого рода персональные данные)
    - /home/folder - Домашняя директория
    - /bin/sh - shell - оболочка, с которой грузится пользователь по умолчанию

### 1.3. Просмотреть файл /etc/shadow, объяснить его структуру.

- cat /etc/shadow

> tedmeadow:$6$RGq0/rrLIvPZpGq1$QWXfoM/.ChoeTskxKeNvGEgjidMPSoj4sp0XrVmSkwunqleKedsCb/S0cV.fZSa/AzMXYmV7r4XJFmU0ieyGi/:19040:0:99999:7:::
> ted:$6$QC2RVaiOjxGrgjl0$P7pvBTG4BVGoxkUt7gb0D/Atv.6BsiXPJ08LUO3ruS.QCwKla300yx5PpHIWK4IEpIypmqcletPeRBJr3jsX0/:19040:0:99999:7:::
> meadow:$6$UWdysXIfDVr87wDv$3Vx3E0QNf3Q4js55uGY6Pq40p47WBKZ7EKxmlDGhgtQKHfP2U7sqBZlvblukN0D9nWDN/1EB6Y7g4ZJsjqnwb1:19040:0:99999:7:::

- Какой алгоритм хэширования используется?

  По $6$ можно понять, что используется SHA-512

- Измените пароль какого-либо пользователя, посмотрите, что изменилось? passwd meadow

  Было:

  > meadow:$6$UWdysXIfDVr87wDv$3Vx3E0QNf3Q4js55uGY6Pq40p47WBKZ7EKxmlDGhgtQKHfP2U7sqBZlvblukN0D9nWDN/1EB6Y7g4ZJsjqnwb1:19040:0:99999:7:::

  Стало:

  > meadow:$6$iVSyHkUcKZVRcdLl$cmHHW9eSbLekOwCVy9IYF.feY/Iy/xAFvParY/zWoQhCPzg/xStTQbYxy.8EbNW0N8Xl09I6y5VpR1FRosvj..:19040:0:99999:7:::

  Изменился только зашифрованный пароль

### 1.4. Создайте новую группу, которая будет включать двух из трех пользователей (user1, user2). Удалите группы тех пользователей, которых вы объединили (user1, user2).

- addgroup meadows
- adduser ted meadows
- adduser meadow meadows
- delgroup ted

  > delgroup: 'ted' still has 'ted' as their primary group!

- delgroup meadow

  > delgroup: 'meadow' still has 'meadow' as their primary group!

## 2. Разрешения

### 2.1. Авторизуйтесь под первым пользователем (user1). Создайте файл в домашней директории пользователя, проверьте разрешения по умолчанию.

- su - ted
- touch test_file
- ls -l

> -rw-r--r-- 1 ted ted 0 Feb 17 20:57 test_file

2.2. Измените права таким образом, чтобы пользователь той же группы (user2) мог редактировать файл, а отдельный (user3) - не мог видеть и редактировать.

- chown ted:meadows test_file
- chmod g+w test_file
- chmod o-r test_file
- ls -l

> -rw-rw---- 1 ted meadows 0 Feb 17 20:57 test_file

2.3. Измените права на файл таким образом, чтобы владелец (user1) и отдельный пользватель (user3) могли читать и редактировать файл, а второй пользователь (user2) - нет.

- chmod g-rw test_file
- chmod o+rw test_file
- ls -l

> -rw----rw- 1 ted meadows 0 Feb 17 20:57 test_file

2.4. Создайте директорию, поместите в нее несколько файлов. Повторите шаг 2.2 и 2.3 для директории и файлов внутри нее.

- mkdir test_dir
- cd test_dir/
- touch file1
- touch file2
- touch file3
- 2.2
    - chown ted:meadows test_dir/ -R
    - chmod g+w test_dir/ -R
    - chmod o-r test_dir/ -R
    - ls -l test_dir/

      > -rw-rw---- 1 ted meadows 0 Feb 17 21:13 file1 -rw-rw---- 1 ted meadows 0 Feb 17 21:13 file2 -rw-rw---- 1 ted meadows 0 Feb 17 21:13 file3
- 2.3

    - chmod g-rw test_dir -R
    - chmod o+rw test_dir -R
    - ls -l test_dir/

> -rw----rw- 1 ted meadows 0 Feb 17 21:13 file1 -rw----rw- 1 ted meadows 0 Feb 17 21:13 file2 -rw----rw- 1 ted meadows 0 Feb 17 21:13 file3

### 2.5. Создайте скрипт на ЯП Python с расширением *.py. Задайте ему шебанг. Попробуйте запустить скрипт самостоятельно, без вызова интерпретатора.

- vim test.py

```
#!/usr/bin/env python3

print('hello world!')
```

- ./test.py

> -sh: ./test.py: Permission denied

- Чтобы исправить эту ошибку добавляем права для запуска: chmod +x test.py
- ./test.py

> hello world!

## 3. Специальные разрешения

### 3.1. Что делает следующая команда?

- chmod 4762 filename
    - rwsrw--w-
    - Специальное разрешение: SUID
    - user: read & write & execute
    - group: read & write
    - other: write

### 3.2. Зайдите под первым пользователем (user1) и создайте директорию (например, dir1), внутрь поместите несколько исполняемых скриптов с расщирением *.py. Для этой директории рекурсивно установите UID, GID и Sticky Bit. Дайте скриптам права на исполнение.

- mkdir dir1
- cp test.py dir1/py1.py

- cp test.py dir1/py2.py
- cp test.py dir1/py3.py
- chmod u+s dir1/ -R
- chmod g+s dir1/ -R
- chmod +t dir1/ -R
- chmod +x dir1/ -R
- ls -l dir1/

```
-rwsr-sr-t 1 ted ted 46 Feb 17 21:52 py1.py
-rwsr-sr-t 1 ted ted 46 Feb 17 21:52 py2.py
-rwsr-sr-t 1 ted ted 46 Feb 17 21:52 py3.py
```

Пропали все флаги "x", для пользователя и группы появились флаги s, которые означают SUID и SGID соответственно, а така же появился флаг t, который означает Sticky Bit

### 3.3. Удалите права на исполнения для всего в домашней директории.

- chmod -x /home/ted/dir1/*

```
-rwSr-Sr-T 1 ted ted 46 Feb 17 21:52 py1.py
-rwSr-Sr-T 1 ted ted 46 Feb 17 21:52 py2.py
-rwSr-Sr-T 1 ted ted 46 Feb 17 21:52 py3.py
```

Так как флаги s и t находятся в позициях флагов x, то они также показывают флаг x при помощи регистра. Когда s и t = +x, когда же S и T = -x.

### 3.4. Установите для созданной директории (dir1) права таким образом, чтобы пользователи из других групп могли осуществлять навигацию и читать файлы внутри.

- chmod o+x dir1/

```
drwsr-sr-t 2 ted ted 4096 Feb 17 21:52 dir1
```

- Затем измените владельца этой директории на группу отдельного пользователя (user3).
  - chown :tedmeadow dir1

    ```
    drwsr-sr-t 2 ted tedmeado 4096 Feb 17 21:52 dir1
    ```

  - Изменилась группа, которая владеет папкой -> tedmeadow теперь может то, что мог meadow и ted и наоборот. Но при этом ted имеет еще и права юзера.

### 3.5. Авторизуйтесь в системе под вторым пользователем (user2) и создайте директорию {user2}_share. Установите разрешения таким образом, чтобы второй пользователь мог делиться файлами с первым (user1) и отдельным (user3).

- su - meadow

- mkdir meadow_share
- chmod u+s meadow_share/ -R
- chmod +t meadow_share/ -R
- chmod o+w meadow_share/ -R

```
drwsr-srwt 2 meadow meadow 4096 Feb 17 22:31 meadow_share
-rwSr--rwT 1 meadow meadow 0 Feb 17 22:30 test1
-rwSr--rwT 1 meadow meadow 0 Feb 17 22:30 test2
-rwSr--rwT 1 meadow meadow 0 Feb 17 22:31 test3
```

### 3.6. Создайте файл sensetive_data от имени второго пользователя (user2) и запишите в него какую-либо текстовую информацию. Уберите разрешения для группы и всех остальных.

- Авторизуйтесь под отдельным пользователем (user3), попытайтесь прочитать файл sensetive_data с помощью утилиты cat.

  ```
  cat: can't open 'sensetive_data': Permission denied
  ```

  Так как не хватает прав доступа, нам не дают прочитать файл

- Попытайтесь прочитать файл с помощью утилиты vi.

  Появляется пустое окно, как будто мы создали новый файл и снизу написано Permission denied, так как у нас нет прав для чтения, но если мы что то напишем и попытаемся сохранить - нам не хватит прав доступа

- Установите SUID bit на vi (от имени суперпользователя). Пропала надпись снизу Permission denied, но прочитать или сохранить файл все равно нельзя.