Управление пользователями

Основа распределения прав доступа в операционной системе Linux лежит на понятии пользователь. Пользователю-владельцу файла выдаются определенные полномочия для работы с ним, а именно на чтение, запись и выполнение. Также отдельно устанавливаются полномочия на чтение, запись и выполнение для всех остальных пользователей. Поскольку в Linux все есть файл, то такая система позволяет регулировать доступ к любому действию в этой операционной системе с помощью установки прав доступа на файлы.

Добавление пользователя (`useradd`)

Пример УП_01:

Создать пользователя **student01**

`useradd student01`

Эта команда создаст в системе нового пользователя **student01**. Чтобы изменить настройки создаваемого пользователя, вы можете использовать следующие ключи:

Таблица УП_01

Ключ	Описание
-b	Базовый каталог. Это каталог, в котором будет создана домашняя папка пользователя. По умолчанию /home
-c	Комментарий. В нем вы можете напечатать любой текст.
-d	Название домашнего каталога. По умолчанию название совпадает с именем создаваемого пользователя.
-e	Дата, после которой пользователь будет отключен. Задается в формате ГГГГ-ММ-ДД. По умолчанию отключено.
-f	Количество дней, которые должны пройти после устаревания пароля до блокировки пользователя, если пароль не будет изменен (период неактивности). Если значение равно 0, то запись блокируется сразу после устаревания пароля, при -1 - не блокируется. По умолчанию -1.
-g	Первичная группа пользователя. Можно указывать как GID, так и имя группы. Если параметр не задан будет создана новая группа название которой совпадает с именем пользователя.
-G	Список вторичных групп в которых будет находится создаваемый пользователь
-k	Каталог шаблонов. Файлы и папки из этого каталога будут помещены в домашнюю папку пользователя. По умолчанию /etc/skel.
-m	Ключ, указывающий, что необходимо создать домашнюю папку. По умолчанию домашняя папка не создается.
-p	Зашифрованный пароль пользователя. По умолчанию пароль не задается, но учетная пользователь будет заблокирован до установки пароля
-s	Оболочка, используемая пользователем. По умолчанию /bin/sh.
-u	Вручную задать UID пользователю.

Изменение пароля (passwd)

Изменить пароль пользователю можно при помощи утилиты **`passwd`**.

`Пример УП_02:`

Создать для пользователя **student01** пароль **test**

Ввод команды на создание пароля

`passwd student01`

Ввод пароля (на экране не отображается) (рис. УП_02)

`test`

Будет выведено сообщение о нессответсвии пароля политике безопасности (BAD PASSWORD).

Повтор ввода пароля (на экране не отображается)

test

Рис. УП_02 Создание пароля

Основные ключи passwd:

Таблица УП_01

Ключ	Описание
-d	Удалить пароль пользователю. После этого пароль будет пустым, и пользователь сможет входить в систему без предъявления пароля.
-e	Сделать пароль устаревшим. Это заставит пользователя изменить пароль при следующем входе в систему.
-i	Заблокировать учетную запись пользователя по прошествии указанного количества дней после устаревания пароля.
-n	Минимальное количество дней между сменами пароля.
-x	Максимальное количество дней, после которого необходимо обязательно сменить пароль.
-l	Заблокировать учетную запись пользователя.
-u	Разблокировать учетную запись пользователя.

Получение информации о пользователях

w – вывод информации (имя пользователя, рабочий терминал, время входа в систему, информацию о потребленных ресурсах CPU и имя запущенной программы) о всех вошедших в систему пользователях.

who – вывод информации (имя пользователя, рабочий терминал, время входа в систему) о всех вошедших в систему пользователях.

who am i или **whoami** или **id** – вывод вашего имени пользователя.

users – вывод имен пользователей, работающих в системе.

id ***имя_пользователя*** – вывод о идентификаторах пользователя: его uid, имя_пользователя, gid и имя первичной группы и список групп в которых состоит пользователь

groups ***имя_пользователя*** – вывод списка групп в которых состоит пользователь.

Пример УП_03:

Войти в систему под именеем student01 и вывести информацию о пользователях, вошедших в систему.

Открытие второй терминальной линии: **ALT+F2**

Выполняется ввод имени и пароля

Вывод информации о пользователях

users

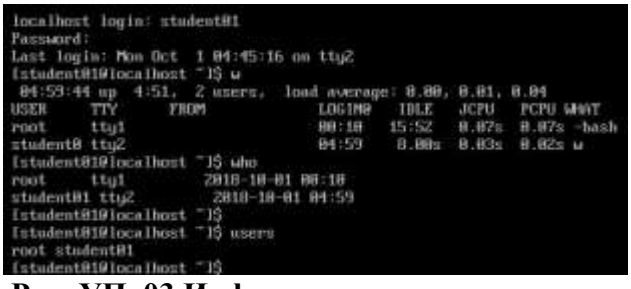


Рис. УП_03 Информация о пользователях

Для хранения сведений об учетных записях пользователей существует файл **/etc/passwd**, доступный всем пользователям для чтения.

Пример УП_04:

Вывести список учетных записей системы

```
cat /etc/passwd
```

Результат (рис. УП_04):

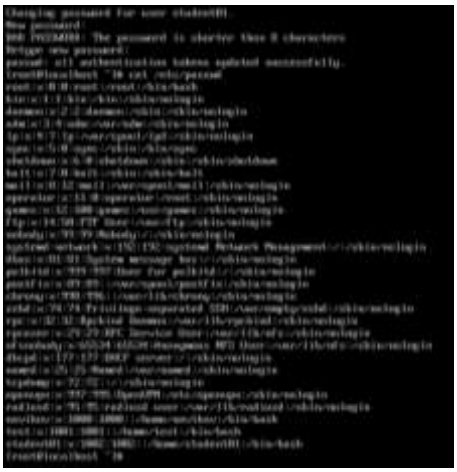


Рис. УП_04 Учетные записи

Поля информации разделяются двоеточием (:). Каждая строка типичного Linux-файла «/etc/passwd» содержит 7 полей:

Root: имя пользователя;

x: место для информации о паролях; пароль можно найти в файле «/etc/shadow». Если в данном поле будет указано значение, отличное от **x**, то во входе в систему будет отказано

0: ID пользователя. Каждый пользователь имеет уникальный идентификатор, благодаря которому система распознает его. ID root-пользователя всегда 0;

0: ID группы. Каждая группа имеет уникальный идентификатор. По умолчанию у каждого пользователя есть главная группа. Опять же, ID root-группы всегда 0;

root: поле для примечаний. Данное поле можно использовать для описания пользователя или его функций. Оно может содержать что угодно, начиная от контактной информации пользователя и заканчивая описанием сервисов, для которых была создана учетная запись;

/root: домашний каталог. Для обычных пользователей домашним каталогом является «/home/username», для root-пользователя это «/root»;

/bin/bash: оболочка пользователя. Данное поле содержит оболочку, которая будет создана, или команды, которые будут выполняться при входе пользователя в систему.

Пример УП_05:

Создать пользователя test в сеансе пользователя student01.

```
useradd test
```

В данном случае возникнет ошибка **Permission denied** (рис. УП_05). Связано это с отсутствием достаточных прав на выполнение данной команды

```
[student01@localhost ~]$ useradd test
-bash: /usr/sbin/useradd: Permission denied
[student01@localhost ~]$ _
```

Рис. УП_05 Отказ в выполнении команды

Получение прав суперпользователя

Для выполнения те тех или иных команд требуются права суперпользователя, которых у обычного пользователя нет. Для выхода из этой ситуации можно использовать команду **su** получения привилегий **root** (переключение в сеанс **root**)

```
su root
Ввести пароль
```

Выполните пример **УП_05** после выполнения команды **su**

Удаление пользователя (userdel)

Пример УП_06:

Удалить пользователя test вместе с домашним каталогом

userdel -r test

userdel имеет всего два основных ключа:

Ключ	Описание
-f	Принудительно удалить пользователя, даже если он сейчас работает в системе.
-r	Удалить домашний каталог пользователя.

Для выхода из сеанса **root** необходимо выполнить команду **exit**

Установка пустого пароля пользователя

Суперпользователь с помощью утилит командной строки **passwd** и **usermod** или путем редактирования файла **/etc/shadow** может удалить пароль пользователь, дав возможность входить в систему без указания пароля.

passwd -d student01

или

usermod -p "" student01

Изменение пользователя (usermod)

Синтаксис

usermode [-ключ] пользователь

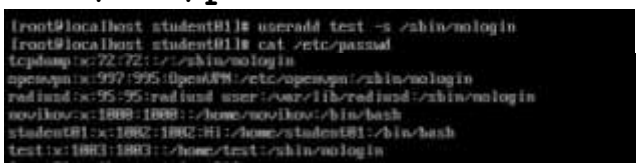
usermod использует те же опции, что и useradd.

Пример УП_07:

Создать пользователя **test** без права локального входа и вручную установить UID на 10 единиц больше, чем ему назначит система.

useradd test –s /sbin/nologin

cat /etc/passwd



UID пользователя **test** равен **1003**. Меняем на **1013**

usermod –u 1013 test

cat /etc/passwd

Группы Linux

Группы разработаны для того, чтобы расширить возможности управления правами. Разберем небольшой пример, возьмем организацию в которой есть только один компьютер, у нас есть администраторы и пользователи. У каждого человека свой аккаунт на компьютере. Администраторы могут настраивать систему, пользователям же лучше не давать воли, чтобы что-то не сломали. Поэтому администраторы объединяются в группу **admin**, и ей дается доступ ко всему оборудованию, реально же ко всем файлам в каталоге **dev**, а пользователи, объеденные в группу **users**, и этой группе дается возможность читать и записывать файлы в общий каталог, с помощью которого они могут обмениваться результатами своей работы. Можно назначать права для каждого пользователя отдельно разрешая ему доступ к тому или иному файлу, но это слишком неудобно. Поэтому и были придуманы группы. А если пользователи — это процессы. Группы используются не столько для обеспечения доступа для пользователей, сколько для управления правами программ, особенно их доступом к оборудованию. Для сервисов создаются отдельные группы и пользователь, от имени которого запущен он, сервис может состоять в нескольких группах, что обеспечивает ему доступ к определенным ресурсам.

Все группы, созданные в системе, находятся в файле **/etc/group**

```
cat /etc/group
```

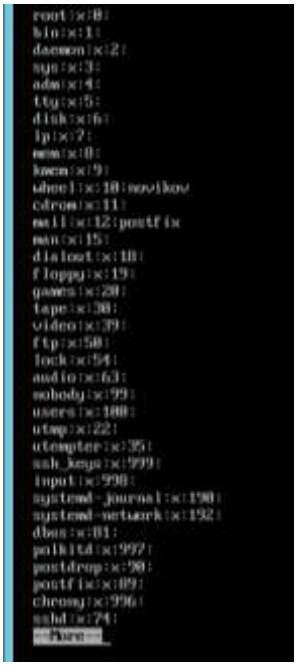


Рис. УП_06 Список групп

daemon — от имени этой группы и пользователя daemon запускаются сервисы, которым необходима возможность записи файлов на диск.

sys — группа открывает доступ к исходникам ядра и файлам include сохраненным в системе

sync — позволяет выполнять команду /bin/sync

games — разрешает играм записывать свои файлы настроек и историю в определенную папку

man — позволяет добавлять страницы в директорию /var/cache/man

lp — позволяет использовать устройства параллельных портов

mail — позволяет записывать данные в почтовые ящики /var/mail/

proxy — используется прокси серверами, нет доступа записи файлов на диск

www-data — с этой группой запускается веб-сервер, она дает доступ на запись /var/www, где находятся файлы веб-документов

list — позволяет просматривать сообщения в /var/mail

nogroup — используется для процессов, которые не могут создавать файлов на жестком диске, а только читать, обычно применяется вместе с пользователем nobody.

adm — позволяет читать логи из директории /var/log

tty — все устройства /dev/vca разрешают доступ на чтение и запись пользователям из этой группы

disk — открывает доступ к жестким дискам /dev/sd* /dev/hd*, можно сказать, что это аналог рут доступа.

dialout — полный доступ к серийному порту

cdrom — доступ к CD-ROM

wheel — позволяет запускать утилиту sudo для повышения привилегий

audio — управление аудиодрайвером

src — полный доступ к исходникам в каталоге /usr/src/

shadow — разрешает чтение файла /etc/shadow

utmp — разрешает запись в файлы /var/log/utmp /var/log/wtmp

video — позволяет работать с видеодрайвером

plugdev — позволяет монтировать внешние устройства USB, CD и т д

staff — разрешает запись в папку /usr/local