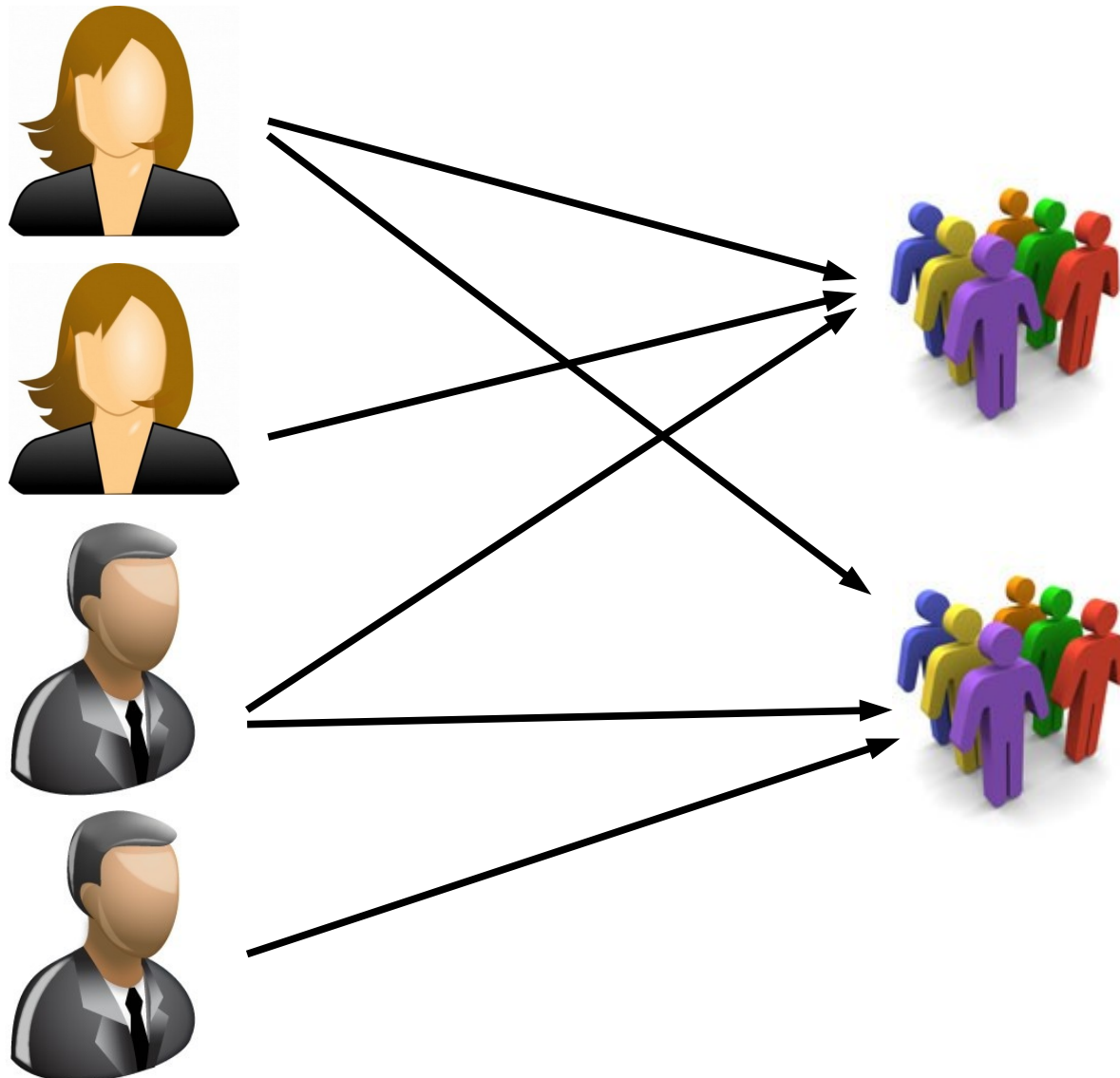


Права доступа

Права доступа



Права доступа

Узнать кто сейчас в системе:

users

Узнать какие вообще есть пользователи:

less /etc/passwd

syslog:x:100:103::...

***bi**:x:**1000**:1000:bi,,,... (UID >= 1000)*

Узнать группы пользователя:

groups <имя_пользователя>

Права доступа

Узнать права доступа:

ls -l <путь>

```
drwxrwxr-x 2 user group ... dir
-rw-rw-r-- 1 user group ... file
```

Права доступа

Узнать права доступа:

ls -l <путь>

```
drwxrwxr-x 2 user group ... dir
-rw-rw-r-- 1 user group ... file
```


Права доступа

Узнать права доступа:

ls -l <путь>

```
drwxrwxr-x 2 user group ... dir
-rw-rw-r-- 1 user group ... file
```

rW-rW-r--



owner group other

r (100, 4) – Read
w (010, 2) – Write
x (001, 1) – eXecute


Права доступа

Узнать права доступа:

ls -l <путь>

```
drwxrwxr-x 2 user group ... dir
-rw-rw-r-- 1 user group ... file
```

rW-rW-r--



owner	group	other
110	110	100

r (100, 4) – Read
w (010, 2) – Write
x (001, 1) – eXecute

4+2+0 4+2+0 4+0+0 = **664**

Права доступа

Права у файлов:

r – просмотр содержимого

w – редактирование

x – запуск

Права доступа

Права у файлов:

r – просмотр содержимого

w – редактирование

x – запуск

Права у директорий:

r – просмотр содержимого

w – создание/удаление файлов/поддиректорий

x – 1) вход в директорию

2) просмотр *inode* файлов/поддиректорий

Права доступа

Изменить права доступа:

chmod [u go a] [+ -] [rwx] <путь>

chmod ug+rw file.txt

chmod [octal code] <путь>

chmod 777 file.txt

Права доступа

Изменить права доступа:

chmod [u go a] [+ -] [rwx] <путь>

chmod ug+rw file.txt

chmod [octal code] <путь>

chmod 777 file.txt

Изменить владельца:

chown new_user:new_group <путь>

chown zoidberg file.txt

chown :futurama file.txt

Права доступа

Суперпользователь:

`root` (группа `root`)

Сделать от имени root:

`sudo` <действие>

`sudo mkdir dir`

`sudo chmod -r dir`

Новые команды

Посчитать что-то в файле:

wc [что-считаем] <путь>

wc -l file.txt

wc -w file.txt

wc -c file.txt

Новые команды

Посчитать что-то в файле:

wc [что-считаем] <путь>

wc -l file.txt

wc -w file.txt

wc -c file.txt

Сравнить файлы/директории:

diff [-q -r] <путь1> <путь2>

diff file1.txt file2.txt | less

diff -qr dir1/ dir2/

Новые команды

Узнать сколько места занимаем на диске:

du [--max-depth <глубина> -h] <путь>

du -h ~/

du --max-depth 1 -h .

df [-h]

узнать сколько места занято/свободно
во всей системе

Новые команды

Вывести часть файла:

head [-n <количество строк>] <путь>

tail [-n <количество строк>] <путь>

```
head -n 10 file.txt
```

```
tail -n 50 file.txt | less
```


Новое о старом

Работа с файлами/директориями:

можно использовать {имя1, имя2, . . . }

```
mkdir ./ {dir1, dir2, dir3}
```

```
touch ~/ {file1, file2}.txt
```

Новое о старом

Работа с файлами/директориями:

можно использовать {имя1, имя2, . . . }

```
mkdir ./ {dir1, dir2, dir3}
```

```
touch ~/ {file1, file2}.txt
```

Вывод с сортировкой:

```
ls --sort=[вид сортировки] -l <путь>
```

```
ls --sort=size -l ~/
```

```
ls --sort=time -l .
```

Новое о старом

Перенаправление в один файл:

&> или **&>>**

```
program &>> both.txt
```

Перенаправление одного потока в другой:

2>&1 stderr в stdout

1>&2 stdout в stderr

Новое о старом

Перенаправление в один файл:

&> или **&>>**

```
program &>> both.txt
```

Перенаправление одного потока в другой:

2>&1 stderr в stdout

1>&2 stdout в stderr

Перенаправление в никуда и из ниоткуда:

/dev/null

```
program 2>/dev/null
```

```
cat /dev/null > file.txt
```