

1. Создать файл crontab, который ежедневно регистрирует занятое каждым пользователем дисковое пространство.

Скрипт. Проверял для папки /home/phoenix т.к. для других папок скрипт долго выполняется..

```
#!/bin/bash
data=$(date +"%d-%m-%y")
sed 's/.*//' /etc/passwd >> file; #считываем имена всех пользователей
while read User;
do
    sudo find /home/phoenix/ -type f -user $User >> 1 #находим все файлы
                                                    #принадлежащие каждому пользователю
                                                    #и записываем их в файл

    while read file_n;
    do
        sudo ls -sk $file_n >> 2                    #смотрим размер файлов в килобайтах
                                                    #и записываем это в файл
    done < ./1
    rm 1;
    if [ -f ./2 ]
    then
        sed -i 's/.*//'. ./2                        #убираем имя
        sed -i 's/0.*//'. ./2                       #убираем пустые значения
        sed -i 's/итого.*//'. ./2                   #убираем из вывода 'итого'
        sum=0;
        while read Line;
        do
            let sum+=Line;
        done < ./2
        echo "user: $User = $sum К" >> $data.txt
        rm 2
    else
        echo "user: $User = 0 К" >> $data.txt
    fi
    continue
done < ./file
rm file
```

Запускаем. Видим, что появился новый файл 16-11-18.txt

```
phoenix@phoenix:~/script_test$ ls
6.2.sh 6.2.sh.save home.sh task01 task02 task03 task2 task3 test.sh test.sh.save time
phoenix@phoenix:~/script_test$ ./home.sh
[sudo] пароль для phoenix:
phoenix@phoenix:~/script_test$ ls
16-11-18.txt 6.2.sh.save task01 task03 task3 test.sh.save
6.2.sh home.sh task02 task2 test.sh time
phoenix@phoenix:~/script_test$
```

Выводим содержимое файла на экран

```
phoenix@phoenix:~/script_test$ cat 16-11-18.txt
User: root = 0 K
User: daemon = 0 K
User: bin = 0 K
User: sys = 0 K
User: sync = 0 K
User: games = 0 K
User: man = 0 K
User: lp = 0 K
User: mail = 0 K
User: news = 0 K
User: uucp = 0 K
User: proxy = 0 K
User: www-data = 0 K
User: backup = 0 K
User: list = 0 K
User: irc = 0 K
User: gnats = 0 K
User: nobody = 0 K
User: systemd-timesync = 0 K
User: systemd-network = 0 K
User: systemd-resolve = 0 K
User: systemd-bus-proxy = 0 K
User: syslog = 0 K
User: _apt = 0 K
User: lxd = 0 K
User: messagebus = 0 K
User: uidd = 0 K
User: dnsmasq = 0 K
User: phoenix = 8028 K
User: sshd = 0 K
User: lightdm = 0 K
User: whoopsie = 0 K
User: avahi-autoipd = 0 K
User: avahi = 0 K
User: colord = 0 K
User: speech-dispatcher = 0 K
User: hplip = 0 K
User: kernoops = 0 K
User: pulse = 0 K
User: rtkit = 0 K
User: saned = 0 K
User: usbmux = 0 K
User: new_root = 0 K
User: new_root1 = 0 K
User: new_root2 = 0 K
User: user1 = 0 K
User: user2 = 0 K
User: user3 = 0 K
phoenix@phoenix:~/script_test$
```

Ставим скрипт в ежедневное расписание.

```
GNU nano 2.5.3          Файл: /tmp/crontab.АРВ3Х/crontab
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command
@daily /home/phoenix/scripts_test/home.sh
```

2. Создать скрипт ownersort.sh, который в заданной папке копирует файлы в директорию, названные по имени владельца каждого файла. Учтите, что файл должен остаться принадлежать соответствующему владельцу

```
GNU nano 2.5.3          Файл: ownersort.sh
#!/bin/bash
sudo find $1 -type f >> 1 #поиск всех файлов в папке
while read file;
do
    user=$(stat -c%U $file); #определение пользователя
    mkdir $user             #создание папки с именем пользователя
    # echo "name of file" ${file##*/} - для отладки было
    cp ${1}/${file##*/} ${1}/${user}/${file##*/} #копируем в папку пользователя
    chown --reference=${1}/${file##*/} ${1}/${user}/${file##*/} #копируем владельца файла
    chmod --reference=${1}/${file##*/} ${1}/${user}/${file##*/} #копируем права файлов
done < 1
rm 1 #удаляем файл со списка всех файлов в папке
```

Запускаем скрипт

```
phoenix@phoenix:~/test$ ls
2 file1 file2 head ownersort.sh
phoenix@phoenix:~/test$ ll
итого 20
drwxrwxr-x 2 phoenix phoenix 4096 ноя 16 00:34 ./
drwxr-xr-x 21 phoenix phoenix 4096 ноя 15 23:26 ../
-rw-rw-r-- 1 phoenix phoenix 202 ноя 15 23:29 2
-rw-rw-r-- 1 phoenix phoenix 0 ноя 15 14:33 file1
-rw-rw-r-- 1 phoenix phoenix 0 ноя 15 14:33 file2
-rwxrwxr-x 1 phoenix phoenix 132 ноя 15 19:56 head*
-rwxrwxr-x 1 phoenix phoenix 747 ноя 16 00:32 ownersort.sh*
phoenix@phoenix:~/test$ ./ownersort.sh /home/phoenix/test
```

```
phoenix@phoenix:~/test$ ls
2 file1 file2 head ownersort.sh phoenix
phoenix@phoenix:~/test$ cd phoenix/
phoenix@phoenix:~/test/phoenix$ ll
итого 24
drwxrwxr-x 2 phoenix phoenix 4096 ноя 16 00:34 ./
drwxrwxr-x 3 phoenix phoenix 4096 ноя 16 00:34 ../
-rw-rw-r-- 1 phoenix phoenix 148 ноя 16 00:34 1
-rw-rw-r-- 1 phoenix phoenix 202 ноя 16 00:34 2
-rw-rw-r-- 1 phoenix phoenix 0 ноя 16 00:34 file1
-rw-rw-r-- 1 phoenix phoenix 0 ноя 16 00:34 file2
-rwxrwxr-x 1 phoenix phoenix 132 ноя 16 00:34 head*
-rwxrwxr-x 1 phoenix phoenix 747 ноя 16 00:34 ownersort.sh*
phoenix@phoenix:~/test/phoenix$
```