

Домашнее задание 1. Shell и ssh

Все задачи можно протестить подключившись к 194.5.78.240 под пользователем homework1_user

Процесс тестирования

1. В каждой задаче вам нужно написать shell-скрипт. Для запуска задачи вы положить его на тест-машину (AllCups это сделает сам). Далее я буду называть этот скрипт `$SCRIPT` и описывать ваши шаги на сервере 194.5.78.240 .
2. Запустите проверку вашего скрипта с помощью команды

```
# TASK_NUM - число, номер задачи
echo $TASK_NUM | /opt/hw1/control/run_control.sh $SCRIPT
```

Сверьте результат с эталонным ответом для тестового стенда

Задания

1. **1 балл.** Дан файл `/opt/hw1/datasets.tar.gz` , содержащий файлы `queries.txt` и `features.txt`

```
# queries.txt
# <query>\t<qid>. Обратите внимание, в запросе могут быть пробелы
PekgC VfwfS8rp4gTAqQPNkSf4jL      406e5b5603
UGEcfvVUQzoQZzE2 LOfs5094xFU7L8VUVNaK33 7 SR      30ded66905
Edk91x 6          980837b81d
wkd      4f12643284

# features.txt
# <qid>\t<f1>\t<f2>\t<f3>\t<f4>
ec56644b1c      25690.26773      8562.29618      7798.9810      11043.31131
357e85d700      20691.17109      2188.29939      21218.4209      66.23120
```

Требуется вывести на stdout объединение этих файлов со следующими колонками, **отсортированное по qid** (исправлено!):

```
<qid>\t<query>\t<f2>\t<f3>
```

Обратите внимание, что не по всякому qid могут найтись фичи, такие писать не нужно.

Ответ проверяется по контрольной сумме вывода. Эталонный ответ на стенде (через `run_control.sh`):

```
a90421945f57707123c859ab7ef966707b0396353a69c584c2fe3a8bd5940c0c
```

Требуемые команды/конструкции:

```
tar, mktmp, cat, cut, sort, join ; VAR=$(CMD) ;
```

2. 1 балл. Напишите скрипт, который скопирует `/opt/hw1/datasets.tar.gz` в путь `/opt/user2_for_user3/datasets.tar.gz` и разрешите пользователю `user3` его прочитать. Владелец конечного файла должен быть `user2`. Учите права на директории/файлы в системе.

Сведения для задачи:

```
$ ls -la /opt
total 24
drwxr-xr-x 1 root  root  4096 Sep 20 22:32 .
drwxr-xr-x 1 root  root  4096 Sep 20 22:34 ..
drwxr-xr-x 1 root  root  4096 Sep 20 22:34 hw1
drwx----- 1 user2 user2 4096 Sep 20 22:32 user2_for_user3

$ ls -la /opt/hw1/ | grep datasets
-rw-r--r-- 1 root      root      44238 Sep 20 22:32 datasets.tar.gz

root$ ls -la /opt/user2_for_user3
total 12
drwx----- 1 user2 user2 4096 Sep 20 22:32 .
drwxr-xr-x 1 root  root  4096 Sep 20 22:32 ..

root$ cat /etc/sudoers | tail -n 5
hw1_docker  ALL=(user2) NOPASSWD: ALL

# Для проверки
hw1_docker  ALL=(user3) NOPASSWD: /bin/ls -la /opt/user2_for_user3/datasets.tar.gz
hw1_docker  ALL=(user3) NOPASSWD: /bin/sha256sum /opt/user2_for_user3/datasets.tar.gz
```

Ответ проверяется по контрольной сумме файла. Эталонный ответ на стенде (через `run_control.sh`):

6dc47d8bf9f7dd77d8c1c69493ed8d75939401b4e8d087916e3c34172aeca330

Требуемые команды: `sudo`, `cp`, `chmod`, `chown`

3. 2 балла. Передайте на сервер `$MASTER_SERVER` (переменная окружения), на порт 12345 строку

```
HELLO_i_AM_a_WORKER,$HOSTNAME
```

Где вместо `$HOSTNAME` нужно подставить имя вашей машины. После этого начните слушать порт 6789, куда мастер через 1 секунду пришлет строку. Ее нужно распечатать в `stdout`.

Обратите внимание, что у вас 1 попытка - потом мастер закрывает порт (нужно перезайти на сервер).

Эталонный ответ на стенде: mS8H2bcewD2xtWyglj2UVGuL3MJxbEQQ

Требуемые команды: nc (обратите внимание на флаг, который делает shutdown в конце input ; флаг для прослушивания порта), hostname

4. 2 балла, 1 добавляется к экзамену В /opt/hw1/keys/ лежат 150 частных ssh-ключей. Один из них позволит подключиться к серверу \$MASTER_SERVER (порт 22) под пользователем user4. Нужно найти его и распечатать в stdout:

- Его номер
- Его md5 отпечаток (в формате как в эталонном ответе)

Обратите внимание:

- Нужно игнорировать host key, см. <https://askubuntu.com/a/87452> (<https://askubuntu.com/a/87452>). Доп. опции в ssh можно передавать через -o
- <https://serverfault.com/a/61936> (<https://serverfault.com/a/61936>) - не спрашивать пароль, использовать только ssh-ключи
- Сам ключ можно передать через -i
- Лог ssh можно сохранить через флаг -E logfile
- Простой перебор ключей не уложится по времени - нужен способ предложить серверу сразу все. Используйте ssh-agent

Эталонный ответ на стенде:

```
107
MD5:c8:48:81:66:01:50:25:a0:db:18:23:09:bb:d2:d1:2f
```

Требуемые команды/конструкции: ssh, ssh-agent, ssh-keygen, ssh-add, mktemp, grep, cut, sed, awk, /bin/true ; циклы bash ; перенаправление в /dev/null

Домашнее задание 1. Компиляция и отладка C-программы

Задачи решаются, подключившись к 194.5.78.240 под пользователем homework2_user из VS Code!

Процесс тестирования

1. В каждой задаче вам нужно написать shell-скрипт. Для запуска задачи вы положить его на тест-машину (AllCups это сделает сам). Далее я буду называть этот скрипт \$SCRIPT и описывать ваши шаги на сервере 194.5.78.240 .
2. Запустите проверку вашего скрипта с помощью команды

```
# TASK_NUM - число, номер задачи
echo $TASK_NUM | /opt/hw2/control/run_control.sh $SCRIPT
```

Сверьте результат с эталонным ответом для тестового стенда

Задания

1. 1 балл.

Напишите ключи компиляции (\$ARGS), которые нужны, чтобы команда

```
# headers - in /opt/hw2/task1_prog/include
gcc -Wall -Wextra -Werror $ARGS -o a.out /opt/hw2/task1_prog/src/main.cpp
```

выполнилась и полученный исполняемый файл вывел "123".

Копия папки /opt/hw2/task1_prog/ есть у вас в homedir.

2. 1 балл.

Напишите скрипт, который позволит команде

```
# ...copy /opt/hw2/task2_prog/* to tmpdir, cd to tmpdir...
gcc -Wall -Wextra -Werror -o a.out main.cpp
```

выполниться и полученный файл должен распечатать "123". Обратите внимание, ваш скрипт будет запускаться во временной директории, куда скопируется содержимое /opt/hw2/task2_prog/

Требуемые команды: sed

3. 2 балла.

Рассмотрим директорию /opt/hw2/task3_prog . В ней лежат 27 бинарных файлов. Определите по 3 файла:

- a. Вывод которых начинается с 'a', а собраны они с ключом -O2, без -g
- b. Вывод которых начинается с 'b', а собраны они с ключами -O2 -g
- c. Вывод которых начинается с 'c', а собраны они с ключами -O0 -g

Скрипт должен анализировать файлы и выводить по 3 указанных номера на строку (отсортированно по возрастанию)

Эталонный ответ на стенде:

```
11 15 16
13 19 25
3 4 22
```

Требуемые команды: objdump -d (изучить ассемблер), grep, for, printf (padding номеров), массивы в shell

4. 2 балла, 1 добавляется к экзамену

Попытка запустить /opt/hw2/task4_prog/main завершается ошибкой

```
$ echo "123" | gdb -batch -ex "run" --args /opt/hw2/task4_prog/main
```

```
Program received signal SIGFPE, Arithmetic exception.
```

```
0x000055e27a3981ce in main () at /opt/hw2/sources/task4_prog/main.cpp:9
```

Напишите, что нужно поставить вместо \$ARGS в команду gdb:

```
echo "123" | gdb $ARGS --args /opt/hw2/task4_prog/main
```

чтобы программа вывела ">>123<<" и завершилась без ошибки