**ОТЧЕТ**

о выполнении долгосрочного домашнего задания

по дисциплине:

«Безопасность операционных систем»

Выполнил:

Лимасов С.В.\_\_\_\_\_\_\_\_

(подпись)

Проверил:

Бульдяев С.А.\_\_\_\_\_\_\_

(подпись)

Дата: \_\_\_\_\_\_\_\_\_\_\_\_\_\_

2019 г.

Оглавление

[1. Задание 1 3](#__RefHeading___Toc1308_2743461330)

[2. Задание 2 (Вариант 14) 4](#__RefHeading___Toc1310_2743461330)

[3. Задание 3 (Вариант 2) 6](#__RefHeading___Toc1312_2743461330)

[4. Задание 4 (Вариант 6) 8](#__RefHeading___Toc1314_2743461330)

[5. Задание 5 (Вариант 6) 12](#__RefHeading___Toc1316_2743461330)

[Заключение: 13](#__RefHeading___Toc1318_2743461330)

## Задание 1

Написать командный файл, формирующий структуру подкаталогов в домашнем каталоге необходимую для сдачи дальнейших заданий. Содержимое текстового файла task1.sh представлено ниже (см. листинг 1.1).

Листнинг1.1

#!/bin/bash

path\_result=$1

path\_input=$2

fullname=$3

numberOnList=11

numberOfGroup=3

cd $path\_input

for((i=2;i<=5;i++))

do

count=$(wc -l < $i.txt)

indNum=$(((($numberOnList+($i\*$numberOfGroup))%$count)+1))

touch $path\_result/task$i-$indNum.sh

echo "#!/bin/bash">$path\_result/task$i-$indNum.sh

echo "#$fullname">>$path\_result/task$i-$indNum.sh

task=$(cat $i.txt | sed -n $indNum,${indNum}p)

echo "#$task" >> $path\_result/task$i-$indNum.sh

chmod +x $path\_result/task$i-$indNum.sh

done

Результат выполнения работы скрипта task1.sh с параметрами “bos-ddz/”, “bos-tasks/”, “3-11-lim” представлен на скриншоте (см. рисунок 1.1).

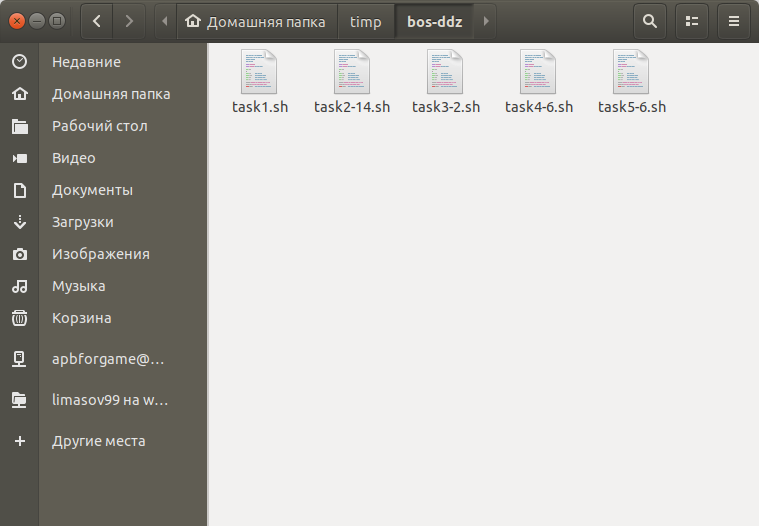


Рисунок 1.1 – Результат работы скрипта task1.sh.

## Задание 2 (Вариант 14)

Найти в файловой системе файл string.h и определить его владельца. Содержимое текстового файла task2-14.sh представлено ниже (см. листинг 2.1).

Листинг 2.1

#!/bin/bash

#2017-3-11-limasov

#14. Найти в файловой системе файл string.h и определить его владельца.

countOfLines=`find / -name "string.h" 2>/dev/null | wc -l`

for (( i=1; i<=countOfLines; i++ ))

do

file=`find / -name "string.h" 2>/dev/null | sed -n $i,${i}p`

info=`ls -l $file`

#echo "info: $info"

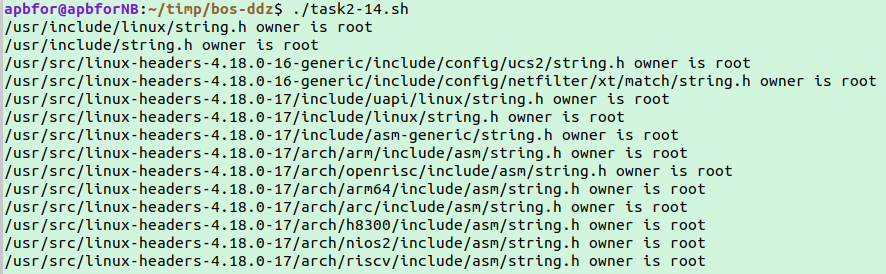
owner=`echo "$info" | cut --fields=3 --delimiter=' '`

echo -n "$file"

echo " owner is $owner"

done

Результат выполнения работы скрипта task2-14.sh представлен ниже (см. рисунок 2.1).

Рисунок 2.1 – Результат работы скрипта task2-14.sh.

## Задание 3 (Вариант 2)

Сравнивать имя пользователя, переданное в качестве, параметра с именем текущего пользователя. Если имена совпадают выводить подтверждение и проверять на равенство имени текущего пользователя значению root. Если равенство подтвердилось – выводить предупреждающее сообщение. В случае если введённое в качестве параметра имя пользователя не совпадает с именем текущего пользователя – выводить сообщение о несовпадении имен. Содержимое текстового файла task3-2.sh представлено ниже (см. листинг 3.1).

Листинг 3.1

#!/bin/bash

#2017-3-11-limasov

#2. Сравнивать имя пользователя, переданное в качестве, параметра с именем текущего пользователя. Если имена совпадают выводить подтверждение и проверять на равенство имени текущего пользователя значению root. Если равенство подтвердилось – выводить предупреждающее сообщение. В случае если введённое в качестве параметра имя пользователя не совпадает с именем текущего пользователя – выводить сообщение о несовпадении имен.

getName=$1

username=`whoami`

if [[ "$getName" == "$username" ]]

then

echo "is eq"

if [[ "$username" == "root" ]]

then

echo "you are root, be carefull"

fi

exit 0

fi

echo "isn't eq"

Результат выполнения работы скрипта проверим на трех именах пользователя - "apbfor" (текущий пользователь), "ivanov" (несуществующий) и "root" (вызов происходит через sudo, поэтому скрипт отрабатывает корректно) представлен на скриншоте (см. рисунок 3.1).

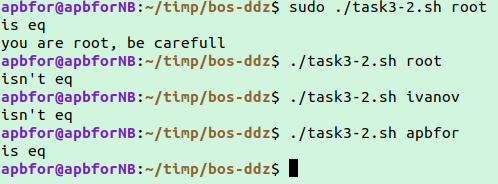


Рисунок 3.1 – Работа скрипта task3-2.sh.

## Задание 4 (Вариант 6)

В заданном каталоге произвести сортировку файлов по размеру: файлы с размером меньше 1кБ переместить в директорию sort.1, файлы с размером больше 1кБ и меньше 1MБ переместить в директорию sort.1000, остальные файлы переместить в директорию sort.mega. Содержимое текстового файла task3-6.sh представлено ниже (см. листинг 4.1).

Листинг 4.1

#!/bin/bash

#2017-3-11-limasov

#6. В заданном каталоге произвести сортировку файлов по размеру: файлы с размером меньше 1кБ переместить в директорию sort.1, файлы с размером больше 1кБ и меньше 1MБ переместить в директорию sort.1000, остальные файлы переместить в директорию sort.mega.

If [[ $# == 0 ]]

then

path="./"

else

path=$1

fi

mkdir "sort.1"

mkdir "sort.1000"

mkdir "sort.mega"

for file in `ls ${path}`

do

if [[ !(-f "$file") ]]

then

echo "$file is not a file!"

continue

fi

echo "Processing $file file"

size=$(du $file | cut --fields=1)

echo "File size is $size"

if [[ size > 1 || size < 1024 ]]

then

mv -v $file ${path}/"sort.1000"

elif [[ size < 1 ]]

then

mv -v $file ${path}/"sort.1"

else

mv -v $file ${path}/"sort.mega"

fi

done

Содержимое текущей директории предоставлено на скриншоте (см. рисунок 4.1)

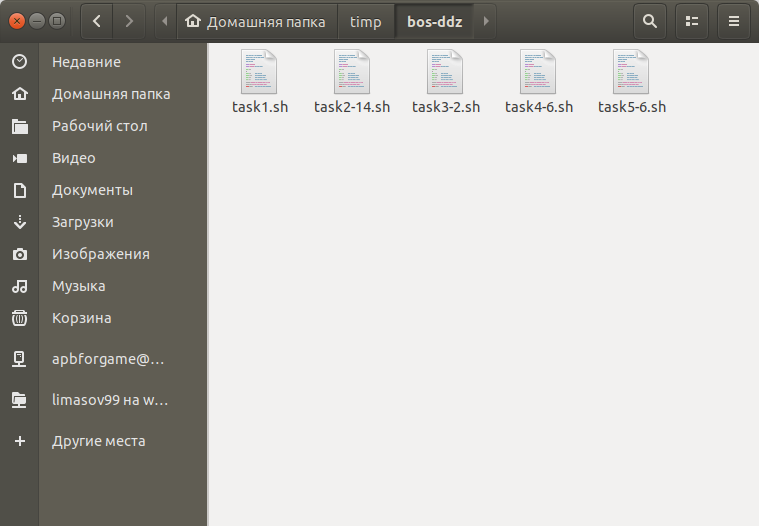


Рисунок 4.1 — Содержимое директории, используемой при запуске скрипта

Результат работы скрипта представлен на двух скриншотах (см. рисунок 4.2 и рисунок 4.3).

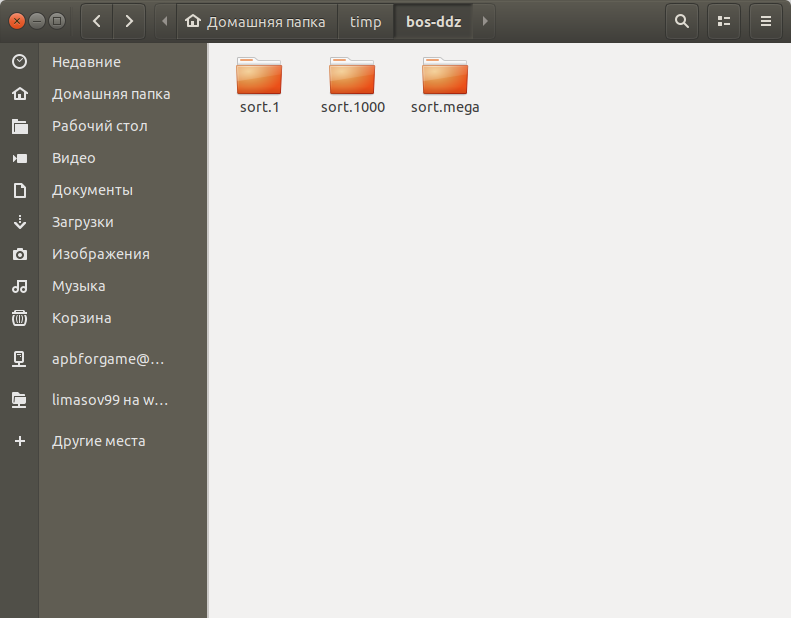


Рисунок 4.2 – Общий вид текущей директории.

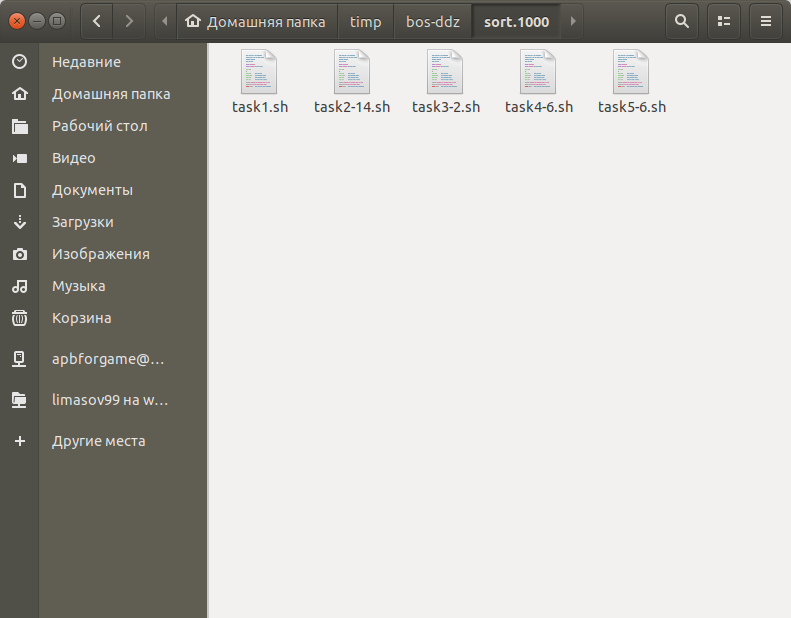


Рисунок 4.3 — Содержимое каталога sort.1000

## Задание 5 (Вариант 6)

Подсчитать количество слов в текстовом файле, содержащих букву 'q'. Текст скрипта представлен ниже (см. листинг 4.1).

Листинг 4.1

#!/bin/bash

#2017-3-11-limasov

#6. Подсчитать количество слов в текстовом файле, содержащих букву 'q'.

if [[ $# == 0 ]]

then

echo "path to file and his name as arg in terminal"

exit 1

fi

file=$1

cat "$file" | tr ' ' '\n' | sed '/[[:punct:] [:digit:]]/d' | grep -o '[^[:space:]]\*q[^[:space:]]\*' | wc -l

exit 0

Содержимое файла file.txt представлено на скриншоте (см. рисунок 4.1).

Демонстрация работы скрипта представлена на скриншоте (см. рисунок 4.2).

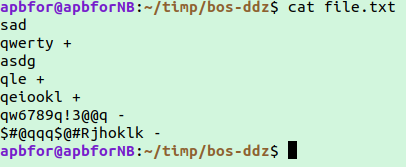


Рисунок 4.1 – Содержимое файла file.txt.



Рисунок 4.2 - Результат выполнения скрипта на файле file.txt

## Заключение:

В ходе выполнения долгосрочного домашнего задания я ознакомился с работой в консоли Ubuntu, а также с синтаксисом Bash и основными функциями.