Отчёт по практическому заданию:

1. Под своей учеткой создать директорию lab1. В ней создать три файла - file1.txt, file2.txt, file3.txt. Зафиксировать текущие права на файлы и на каталог (скрином или текстом). Установить следующие права на файлы:
   * file1.txt: только владелец может читать и писать.
   * file2.txt: только владелец и группа могут читать и писать, остальные не имеют доступа.
   * file3.txt: все пользователи могут читать, но только владелец может писать.

[Siyovush@localhost lab1]$ ls -l

total 0

-rw-------. 1 Siyovush Siyovush 0 Jun 13 22:40 file1.txt

-rw-rw----. 1 Siyovush Siyovush 0 Jun 13 22:40 file2.txt

-rw-r--r--. 1 Siyovush Siyovush 0 Jun 13 22:40 file3.txt

Создайте новую группу - lab\_group. Добавьте своего пользователя в эту группу. Измените группу для каталога lab1 и всех файлов внутри него на lab\_group. Зафиксируйте текущие права на файлы и каталог (скрином или текстом).

[Siyovush@localhost lab1]$ ls -l

total 0

-rw-------. 1 Siyovush lab\_group 0 Jun 13 22:40 file1.txt

-rw-rw----. 1 Siyovush lab\_group 0 Jun 13 22:40 file2.txt

-rw-r--r--. 1 Siyovush lab\_group 0 Jun 13 22:40 file3.txt

В отчете надо отразить команды, которыми вы осуществили задание 1, описать, что команды делают, и приложить сделанные скрины или текст.

[Siyovush@localhost ~]$ mkdir lab1

// Создание директории

[Siyovush@localhost ~]$ cd lab1/

// Ну, хай понятно)

[Siyovush@localhost lab1]$ touch file1.txt

// создание файла

[Siyovush@localhost lab1]$ touch file2.txt

[Siyovush@localhost lab1]$ touch file3.txt

[Siyovush@localhost lab1]$ chmod 600 file1.txt

//Чтение + Запись + Исполнение = 4 + 2 + 1 поэтому 600

[Siyovush@localhost lab1]$ chmod 660 file2.txt

[Siyovush@localhost lab1]$ chmod 644 file3.txt

[Siyovush@localhost lab1]$ sudo groupadd lab\_group

// Создание группы

[sudo] password for Siyovush:

[Siyovush@localhost lab1]$ sudo usermod -aG lab\_group $USER

// -aG добавляет пользователя в указанную группу G, без удаления его из других групп

// $USER это переменная окружения, содержащая имя текущего пользователя

[Siyovush@localhost lab1]$ sudo chgrp -R lab\_group ~/lab1

// -R нужен для рекурсивного применения chgrp для всех файлов в каталоге

1. Под своей учеткой создать директорию lab2. В ней создать файл input.txt, ввести в него 6-7 разных строк текста. Также, перенаправьте содержимое директории lab2 в файл directory\_list.txt. Добавьте текущее время и дату в конец файла directory\_list.txt. Отобразить содержимое, зафиксировать его скрином или текстом.

[Siyovush@localhost lab2]$ cat directory\_list.txt

// прочитать файл

directory\_list.txt

input.txt

Thu Jun 13 22:54:41 +05 2024

Отсортируйте строки в файле input.txt и выведите результат на экран (для этого используйте команду sort с пайпом).

[Siyovush@localhost lab2]$ sort input.txt

a1

a2

a3

a4

a5

a6

Отфильтруйте строки, содержащие букву 'a', из файла input.txt, используя команду grep и пайп.

[Siyovush@localhost lab2]$ grep 'a' input.txt // фильтр

a1

a2

a3

a4

a5

a6

Подсчитайте количество строк в файле input.txt используя команду wc -l. Зафиксировать результаты скрином или текстом.

[Siyovush@localhost lab2]$ wc -l input.txt // Количество строк

6 input.txt

Отсортируйте строки в файле input.txt, выберите строки, содержащие букву 'e', и подсчитайте их количество, используя комбинацию команд sort, grep и wc -l с пайпами. Зафиксировать результаты скрином или текстом.

[Siyovush@localhost lab2]$ sort input.txt | grep 'e' | wc -l

0

Выполните команду ls non\_existent\_file и перенаправьте сообщение об ошибке в файл errors.txt. Просмотрите содержимое файла errors.txt. Зафиксировать результаты скрином или текстом.

[Siyovush@localhost lab2]$ ls doesnt\_exist 2> errors.txt // перенаправляем ошибки с 2>

// это как stderr в других яп

[Siyovush@localhost lab2]$ cat errors.txt

ls: cannot access doesnt\_exist: No such file or directory

В отчете надо отразить команды, которыми вы осуществили задание 2, описать, что команды делают, и приложить сделанные скрины или текст.

[Siyovush@localhost ~]$ mkdir ~/lab2

[Siyovush@localhost ~]$ cd lab2

[Siyovush@localhost lab2]$ ls // вывод содержимого директории

[Siyovush@localhost lab2]$ echo -e "a1\na2\na3\na4\na5\na6" > input.txt // записать строку в input.txt -e нужен для escape последовательностей(особых символов типа \n \b \t)

[Siyovush@localhost lab2]$ ls > directory\_list.txt // перезапись directory\_list.txt

[Siyovush@localhost lab2]$ date >> directory\_list.txt // Добавить в конец файла

1. BASH-скрипты (все скрипты надо снабдить подробными комментариями). Заметьте, что тут могут быть использованы концепции, которые мы ранее не проходили - это намеренно, надо будет поискать и имплементировать найденное в свои скрипты.

3.1. Напишите скрипт, который запрашивает имя пользователя и выводит приветственное сообщение с введенным именем.

#!/bin/bash

echo "Enter your name:"

read name // как std::cin >> в C++. Нужен для инициализации переменной name и присвоения значения для этой переменной.

echo "Hello, $name!"

3.2. Напишите скрипт, который автоматизирует установку прав доступа и изменение владельца для всех файлов и каталогов внутри заданного каталога. При выполнении скрипта, у всех файлов в каталоге должны устанавливаться владелец user1 и группа lab\_group. Также, все файлы должны получать права, при которых у владельца файла есть все права, у членов группы права на чтение и запись, у остальных - только на чтение.

#!/bin/bash

read dir

sudo chown -R user1:lab\_group $dir // изменить владельца -R нужен для рекурсии(применение ко всем файлам)

sudo chmod -R 764 $dir

3.3. Напишите скрипт, который принимает имя файла в качестве аргумента, отсортировывает его содержимое и выводит результат в новый файл.

#!/bin/bash

read input\_file

output\_file="${input\_file%.\*}\_sorted.txt" // {} нужно для разрганичения переменной, % нужен для отсечения всего что стоит после %

sort $input\_file > $output\_file // перезапись

3.4. Напишите скрипт, который запрашивает у пользователя число и проверяет, является ли оно положительным, отрицательным или нулем. Выведите соответствующее сообщение.

#!/bin/bash

read -p "Enter a number: " number // -p нужен для того, чтобы вывести “Enter …”

if [ "$number" -gt 0 ]; then // -gt это > в скрипте

echo "The number is positive"

elif [ "$number" -lt 0 ]; then // -lt это < в скрипте. Названия видимо идут от greater and // lesser than

echo "The number is negative"

else

echo "The number is 0"

fi // заканчивает поток управления(оператор) if

3.5. Напишите скрипт, который принимает имя файла в качестве аргумента и выводит информацию о файле (существует ли файл, является ли он каталогом или обычным файлом, права доступа и размер).

#!/bin/bash

read file

if [ -e $file ]; then // -e нужен для проверки exists

echo "File exists"

if [ -d $file ]; then // -d это it is a directory?

echo "It is a directory"

else

echo "It is a simple file"

fi

ls -l $file

else

echo "File does not exist"

fi

3.6. Напишите скрипт, который создает резервную копию заданного каталога, сохраняя ее в указанное место и добавляя текущую дату к имени архива. Выполните его, архивировав все скрипты и каталоги, использованные в данном практическом задании.

#!/bin/bash

read dir

read backup\_directory

backup\_name="$(basename $dir)\_backup\_$(date +%Y%m%d).tar.gz" // basename это суффикс в названии после слэша

sudo tar -czf $backup\_directory/$backup\_name $dir // c – create new, z – shrink with gzip,

// f – file name, а $dir тут в качестве пути

Все скрипты выложить на гитхаб, там же приложить архив, сделанный скриптом 3.6. Ссылку добавить в отчет.