“Київський фаховий коледж зв’язку”

Циклова комісія Комп’ютерної інженерії

**ЗВІТ ПО ВИКОНАННЮ**

**ЛАБОРАТОРНОЇ РОБОТИ №10**

з дисципліни: «Операційні системи»

**Тема: “Зміна власників і прав доступу до файлів в Linux. Спеціальні каталоги та файли в Linux”**

Виконали студенти

групи КСМ-13а

MathematicainsLivesMatter: Дзизиль Д.Є., Чех І.В.

Перевірив викладач

Сушанова В.С.

Київ 2023

**Мета роботи:**

1. Отримання практичних навиків роботи з командною оболонкою Bash.

2. Знайомство з базовими діями при зміні власників файлів.

3. Знайомство з базовими діями при зміні прав доступу до файлів

4. Знайомство з спеціальними каталогами та файлами в Linux.

**Матеріальне забезпечення занять**

1. ЕОМ типу IBM PC.

2. ОС сімейства Windows (Windows 7).

3. Віртуальна машина – Virtual Box (Oracle).

4. Операційна система GNU/Linux – CentOS.

5. Сайт мережевої академії Cisco netacad.com та його онлайн курси по Linux

**Завдання для попередньої підготовки**

***Готував матеріал студент Чех І.В.***

1. Прочитайте короткі теоретичні відомості до лабораторної роботи та зробіть невеликий словник базових англійських термінів з питань призначення команд та їх параметрів.

|  |  |
| --- | --- |
| Термін англійською | Термін українською |
| File ownership | Власність файлу |
| User owner | Власник користувача |
| File and directory permissions | Права доступу до файлів та каталогів |
| Default permissions | Стандартні права доступу, які надаються файлам та каталогам при їх початковому створенні |
| Group owner | Власник групи |
| GID (Group ID) | Унікальний ідентифікатор групи в операційній системі |
| UID (User ID) | Ідентифікатор користувача |

1. На базі розглянутого матеріалу дайте відповіді на наступні питання:
   1. Яке призначення команди id?

The Linux command "id" displays user IDs (UIDs), group IDs (GIDs), and additional groups a user belongs to.

* 1. Як переглянути які права доступу має власник файлу?

To do this, use the ls command with the –l option.

* 1. Як змінити власника групи?

The chown (change owner) command is used to change the owner of a file group or directory. To change the owner, use the following command:

sudo chown :new\_group file\_or\_directory

* 1. Як можна переглянути у терміналі який тип поточного файлу? Наведіть приклади для різних типів файлів.

If you want to check the type of a file in the terminal, you can use the file command. Here are some examples:

1) file myarchive.zip - ZIP archive;

2) file mydirectory - directory;

3) file example.txt - text file;

4) file mydocument.pdf - PDF document;

5) file myimage.png - PNG image.

* 1. Для чого використовуються дозволи Setuid та Setgid?

The Setuid (suid) permission is used to execute an executable file in the context of its owner, rather than the caller's context. It is typically used to perform tasks that require elevated privileges, such as accessing password files that are only accessible to administrators. For example, the passwd program has the Setuid permission.

The Setgid (sgid) permission is used to execute an executable file in the context of the group of its owner, rather than the caller's group. This permission is useful in a shared environment where group access to resources should be restricted.

* 1. Для чого в системі потрібен так званий “липкий біт” (Sticky Bit). Наведіть приклади коли цей дозвіл доцільно використовувати.

The Sticky Bit is a special permission in UNIX-like operating systems, such as Linux, that provides restrictions for files in a directory with the Sticky Bit set. Its primary function is to limit the ability to delete files from a directory owned by another user. In other words, only the owner of the file, the owner of the directory, and the administrator can delete or rename files in such a directory.

The Sticky Bit is commonly used on temporary directories, like /tmp, to ensure that only the owner of the file, the directory owner, and the system administrator can delete or rename files in that directory. Similarly, if you have a directory that is used for file sharing between users, you may want to set the Sticky Bit to ensure that files can only be deleted by the file owner or administrator.

**Хід роботи.**

***Готував матеріал студент Дзизиль Д.Є.***

1. Початкова робота в CLI-режимі в Linux ОС сімейства Linux:

1.1. Запустіть віртуальну машину VirtualBox, оберіть CentOS та запустіть її. Виконайте вхід в систему під користувачем: CentOS, пароль для входу: reverse (якщо виконуєте ЛР у 401 ауд.) та зпустіть термінал.

1.2. Запустіть віртуальну машину Ubuntu\_PC (якщо виконуєте завдання ЛР через академію netacad)

1.3. Запустіть свою операційну систему сімейства Linux (якщо працюєте на власному ПК та її встановили) та запустіть термінал.

2. Опрацюйте всі приклади команд, що представлені у лабораторних роботах курсу NDG Linux

Essentials:

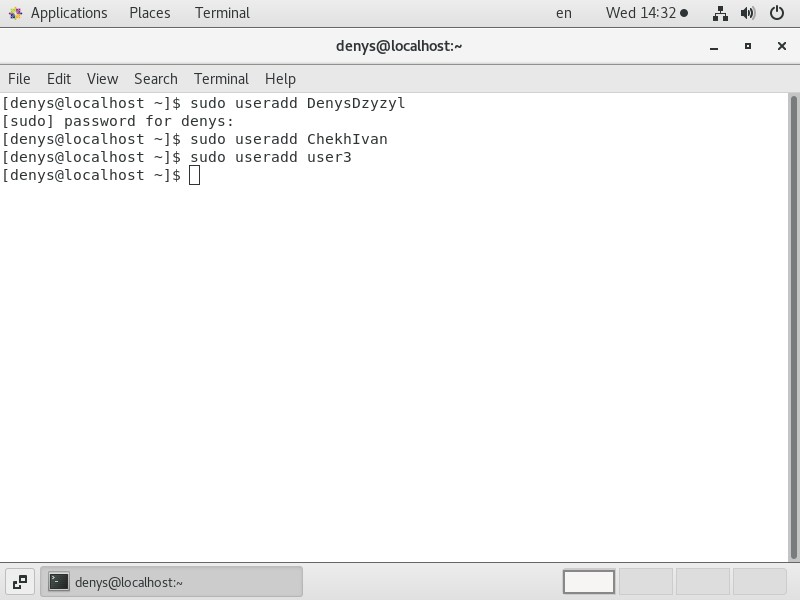
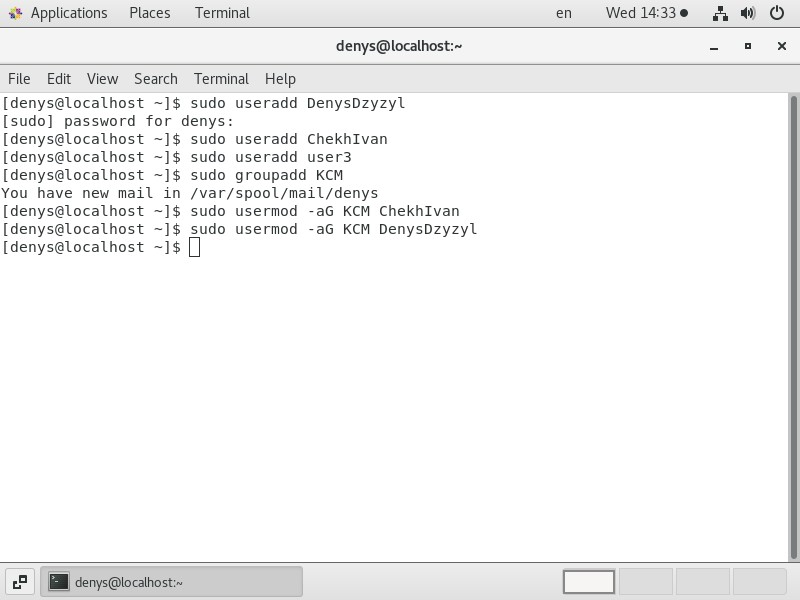
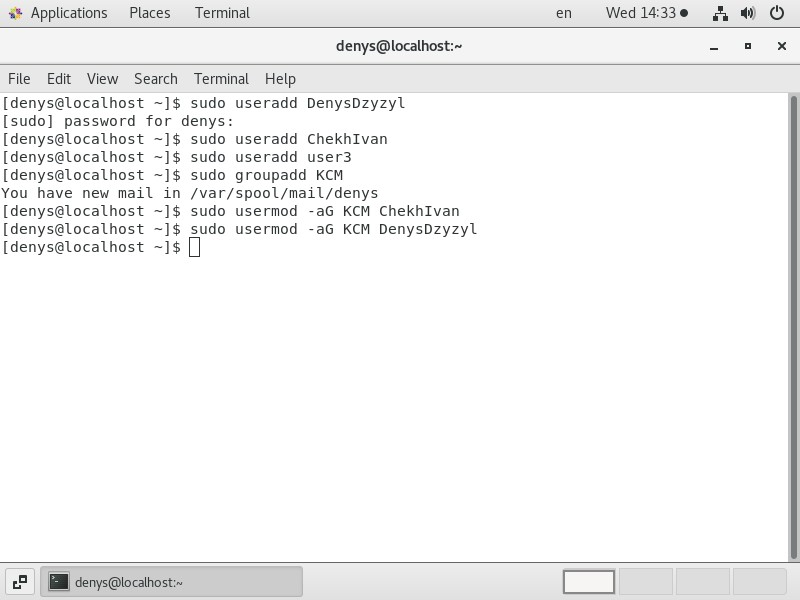
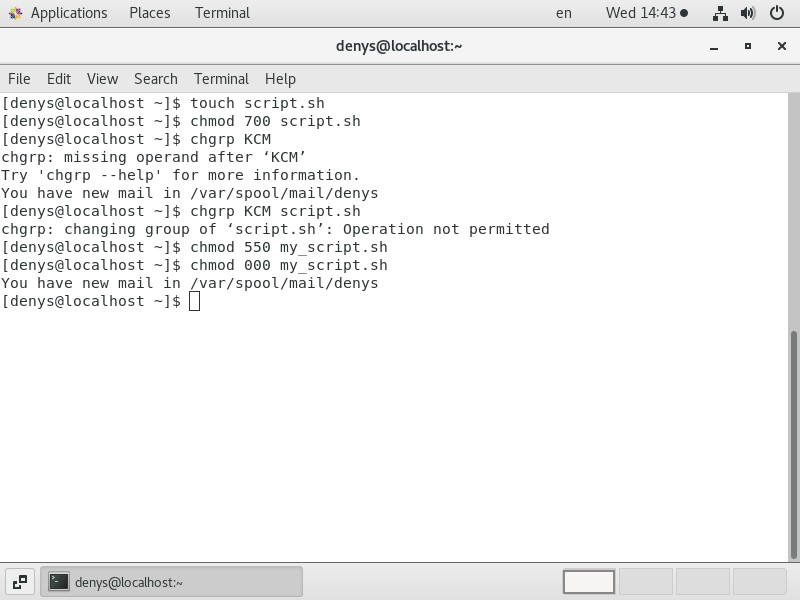
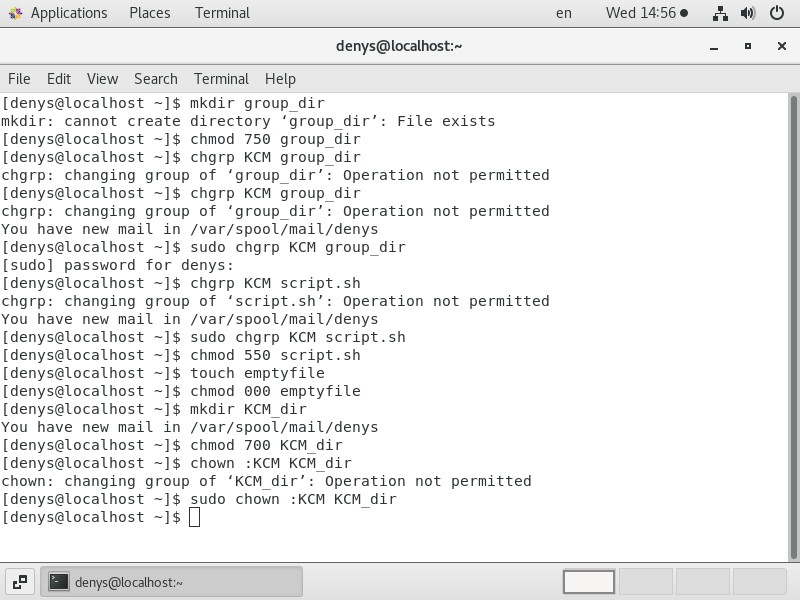
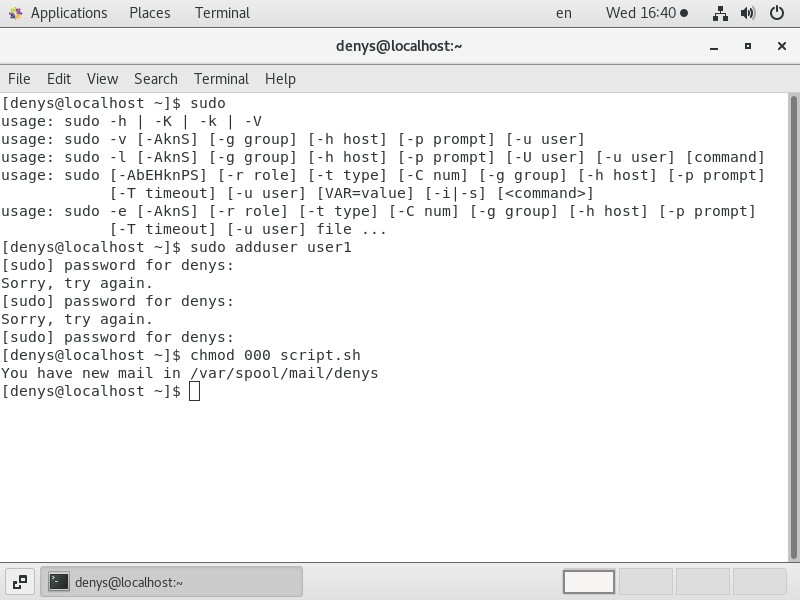
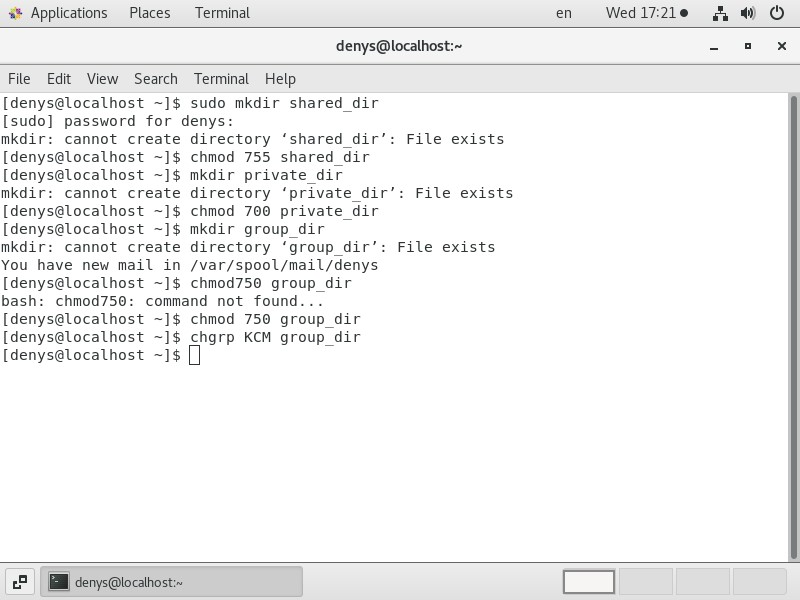
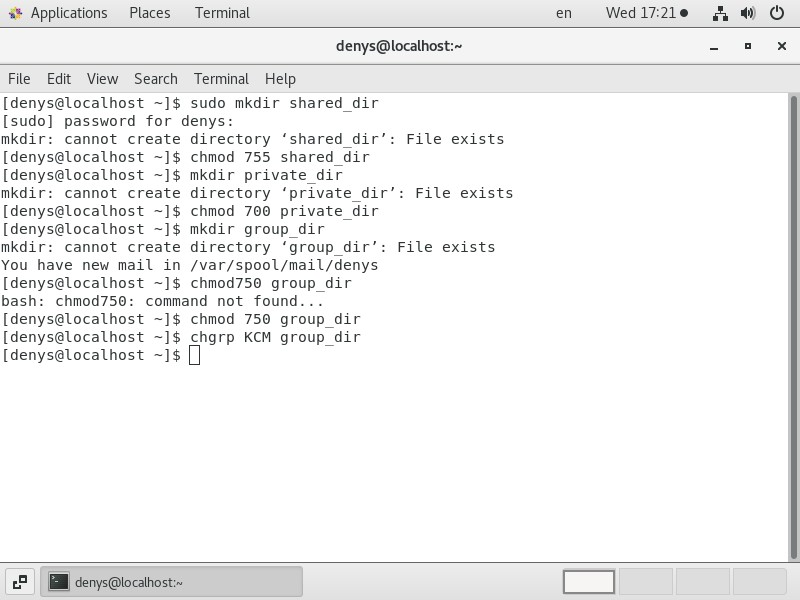
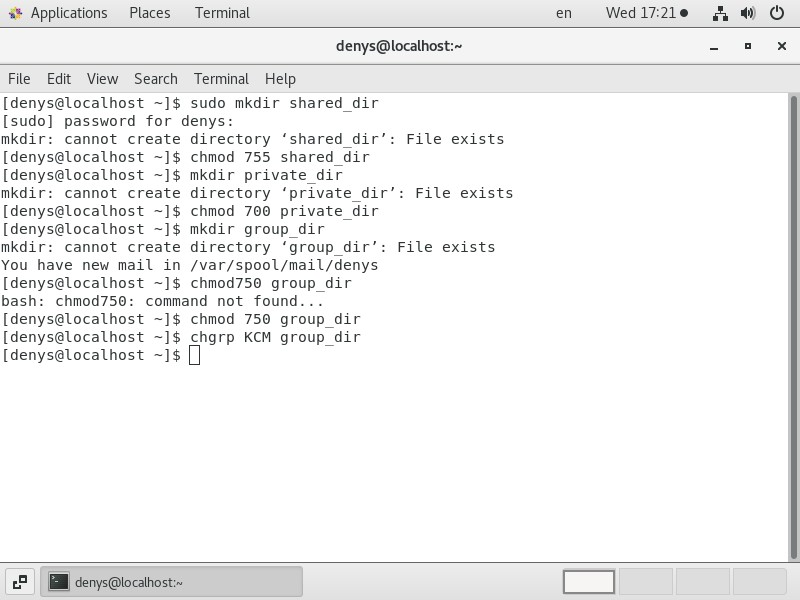
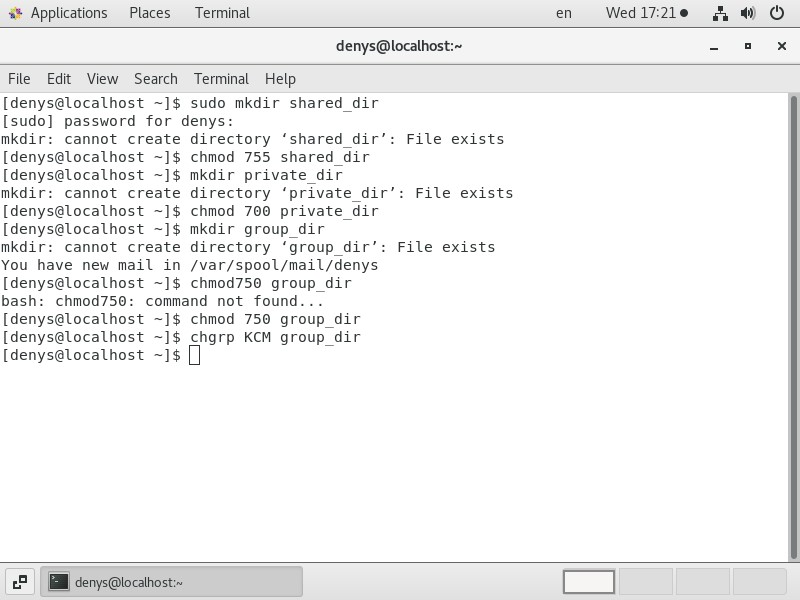
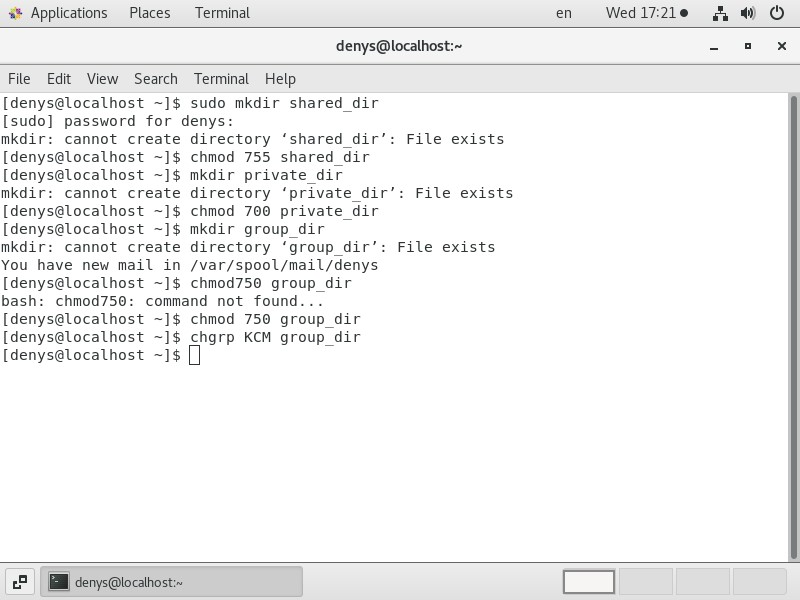
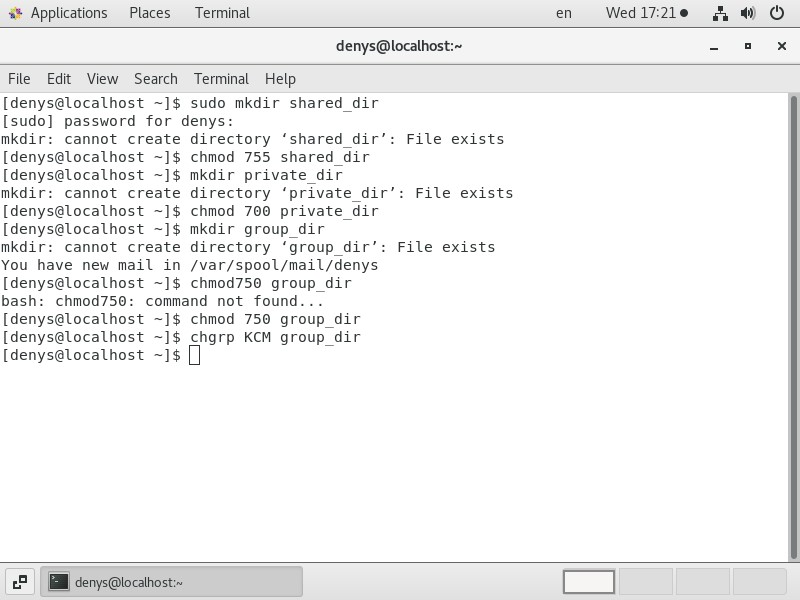
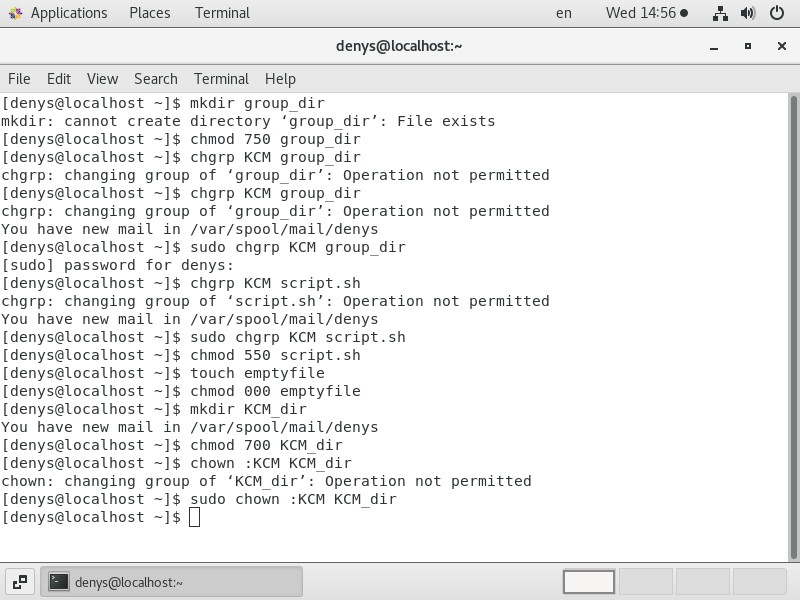
- Lab 17: Ownership and Permissions

- Lab 18: Special Directories and Files

3. Створіть таблицю команд вивчених у п.2 ходу роботи у наступному вигляді:

|  |  |
| --- | --- |
| NDG Linux Essentials - Lab 17 | |
| Command name | Its purpose and functionality |
| cd /tmp | Change to the /tmp directory. |
| mkdir priv-dir pub-dir | Create two directories named priv-dir and pub-dir. |
| touch priv-dir/priv-file<br>touch pub-dir/pub-file | Create two files: priv-file in priv-dir and pub-file in pub-dir. |
| ls -l priv-dir<br>ls -l pub-dir | Display details (long format) of directories priv-dir and pub-dir. |
| ls -ld priv-dir<br>ls -l priv-dir/priv-file | Display details (long format) for priv-dir and priv-file. |
| chmod o-rx priv-dir/ | Deny read and execute access for others to the priv-dir directory. |
| ls -ld priv-dir/ | Check changes in permissions for the priv-dir directory. |
| chmod a+x file<br>chmod g-w file<br>chmod go+r file<br>chmod o=rwx | Change file permissions using the symbolic method. |
| ls -ld pub-dir/<br>chmod o+w pub-dir/<br>ls -ld pub-dir/ | Allow write access for others to the pub-dir directory. |
| ls -l priv-dir/priv-file<br>chmod g-rw,o-r priv-dir/priv-file<br>ls -l priv-dir/priv-file | Deny access for the group and others to the priv-file file. |
| ls -l pub-dir/pub-file<br>chmod a=rw pub-dir/pub-file<br>ls -l pub-dir/pub-file | Grant read and write permission to everyone for the pub-file file. |
| echo "date" > test.sh | Create a file test.sh with the content "date". |
| ./test.sh | Attempt to execute the test.sh file, resulting in an error. |
| chmod u+x test.sh<br>./test.sh | Grant execute permission to the user and execute the test.sh file. |
| NDG Linux Essentials - Lab 18 | |
| ls -ld /tmp | View details of the /tmp directory. |
| ls -ld /var/tmp | View details of the /var/tmp directory. |
| ls -l /etc/shadow | View access rights for the /etc/shadow file. |
| ls -l /usr/bin/passwd | View access rights for the /usr/bin/passwd file. |
| ls -l /usr/bin/wall | View access rights for the /usr/bin/wall file. |
| echo "data" > source | Create a file named source with the content "data". |
| ls -li source | View information about the inode and details of the source file. |
| ln source hardlink | Create a hard link to the source file. |
| ls -li source hardlink | View information about the inode and details of the source and hardlink files. |
| ln hardlink hardlinktwo | Create another hard link to the source file. |
| ls -li hardlink hardlinktwo source | View information about the inode and details of the source, hardlink, and hardlinktwo files. |
| rm hardlinktwo | Remove one of the hard links (hardlinktwo). |
| ls -li source hardlink | View information about the inode and details of the source and hardlink files. |
| rm hardlink | Remove the other hard link (hardlink). |
| ls -li source | View information about the inode and details of the source file. |
| ln -s source softlink | Create a symbolic link (softlink) to the source file. |
| ls -li source softlink | View information about the inode and details of the source and softlink files. |
| ln -s /proc crossdir | Create a symbolic link (crossdir) to the /proc directory. |
| ls -l crossdir | View details of the symbolic link crossdir. |

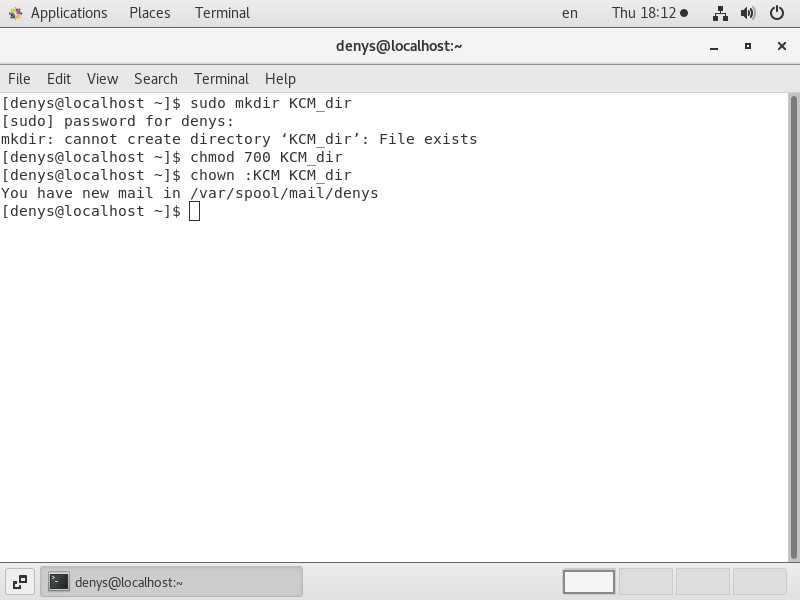
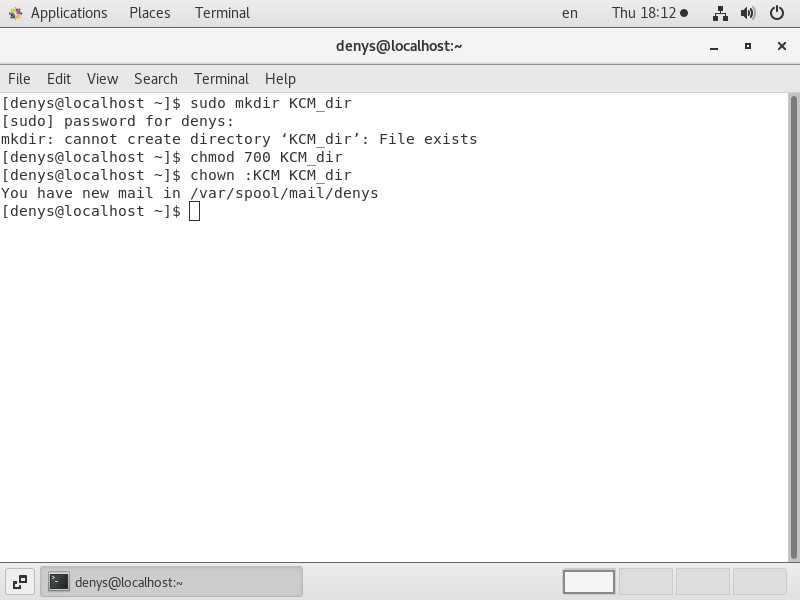
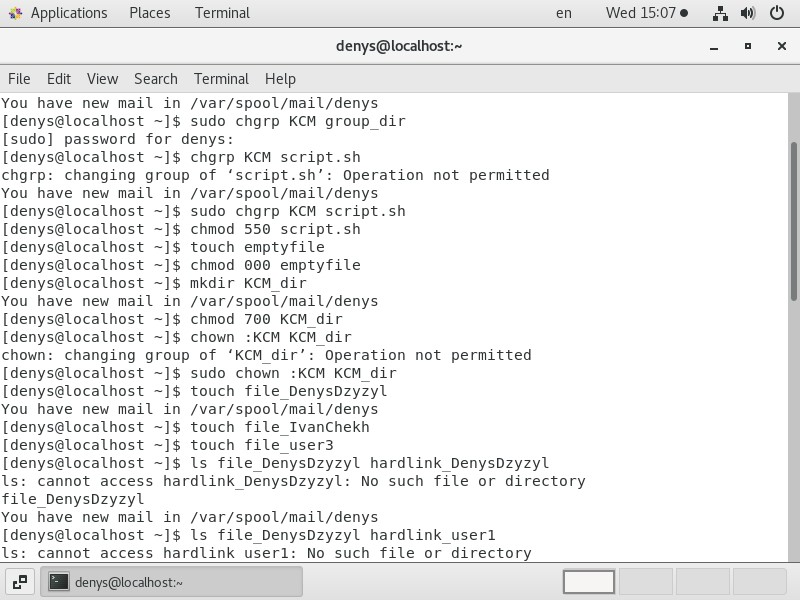
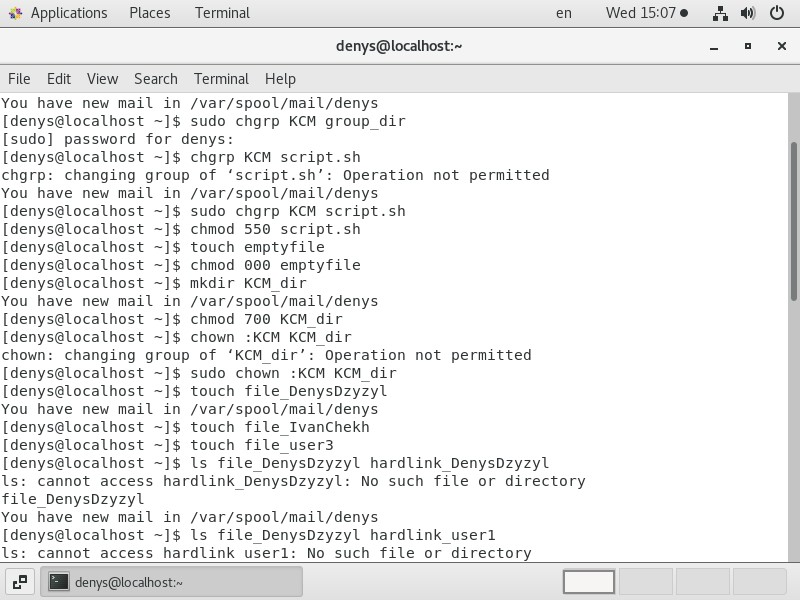
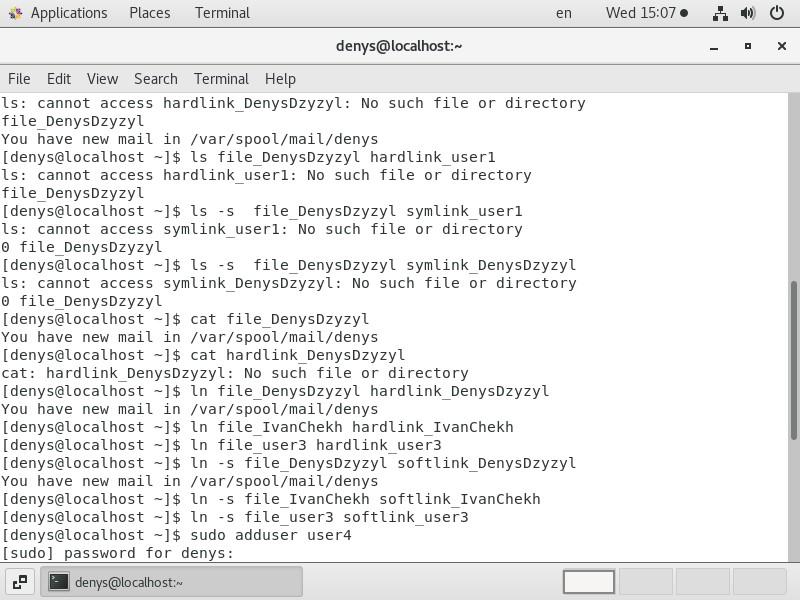
4. Виконайте наступні практичні завдання у терміналі наступні дії (продемонструвати скріншоти):

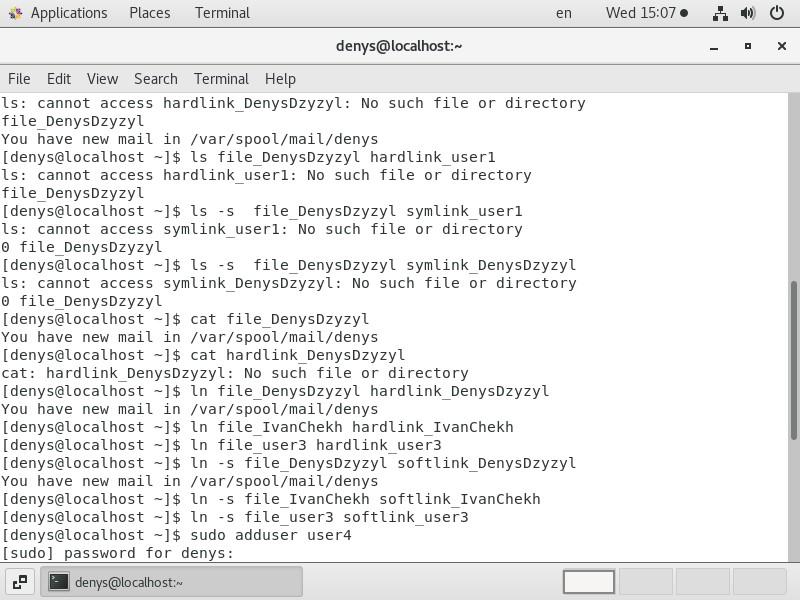
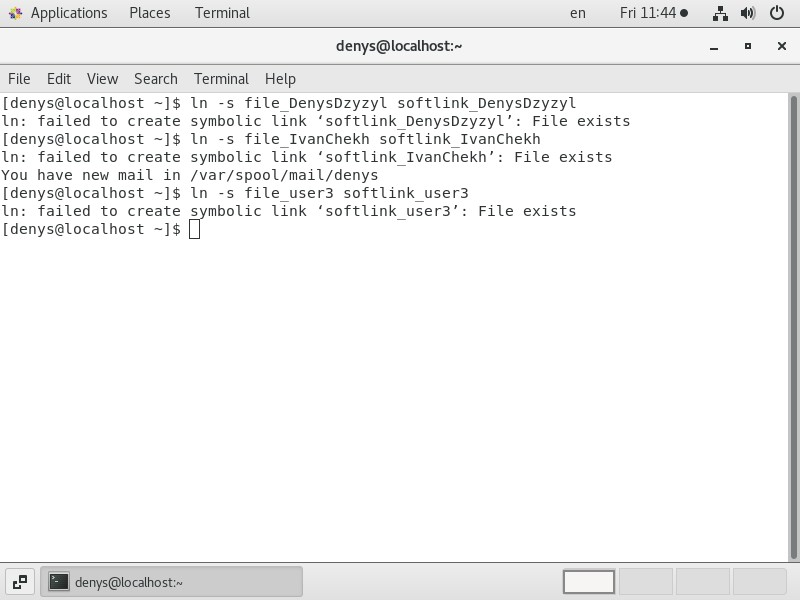
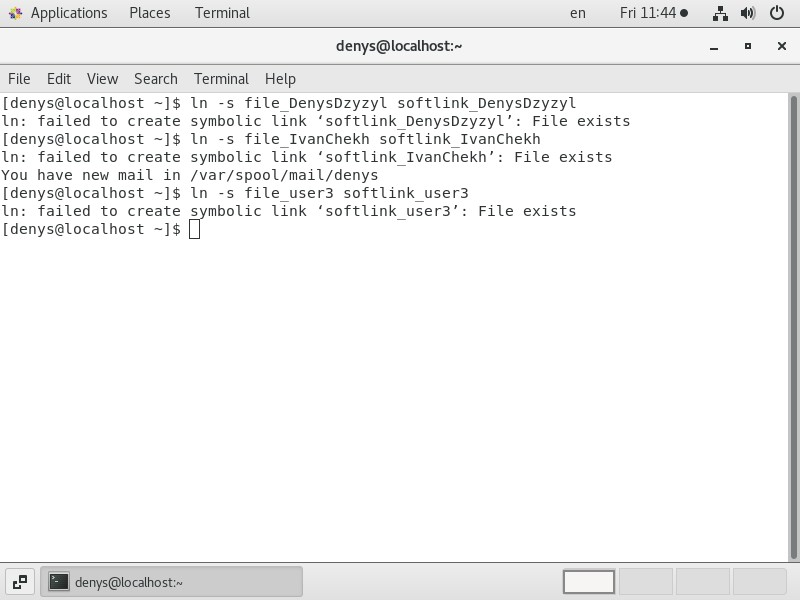
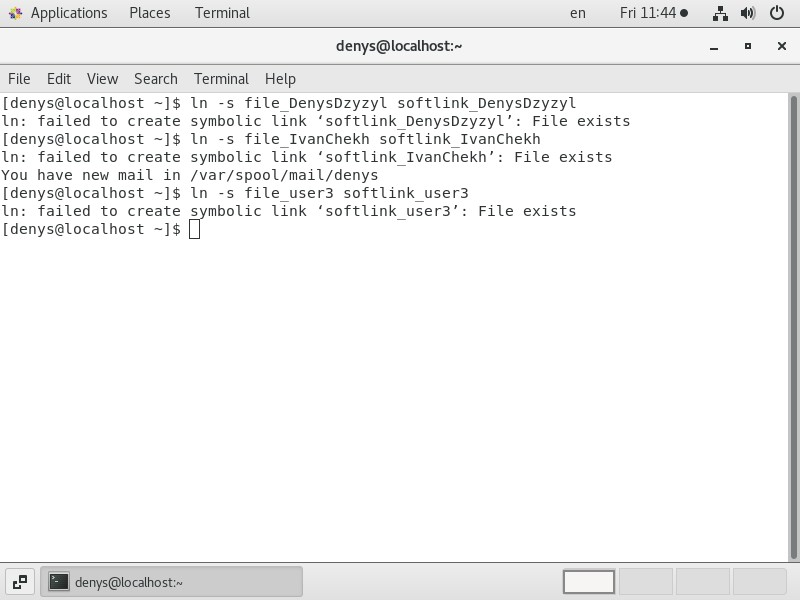
* створіть трьох нових користувачів;
* створіть нову групу користувачів, туди додайте двох, з трьох створених користувачів;
* створіть новий файл, який буде доступний на зчитування, редагування та виконання власником файлу, наприклад найпростіший скриптовий сценарій;
* для користувачів групи власника надайте дозволи на перегляд та виконання (без дозволу на редагування) цього файлу;
* для інших користувачів заборонити доступ до цього файлу;
* подібні дії виконайте для директорій - створіть директорію, яка буде доступна для всіх трьох користувачів, створіть директорію, яку буде доступна тільки для власника, створіть директорію, яку користувачі групи власника зможуть переглядати, але не редагувати;
* створіть порожній файл під назвою emptyfile за допомогою команди touch emptyfile. Тепер “обнуліть” дозволи для файлу з chmod 000 emptyfile. Що станеться, якщо змінити дозволи для emptyfile, передавши лише одне значення для chmod у числовому режимі, наприклад, chmod 4 emptyfile? Що буде, якщо ми використаємо два числа, наприклад chmod 44 emptyfile? Що ми можемо дізнатися про те, як chmod зчитує числове значення?

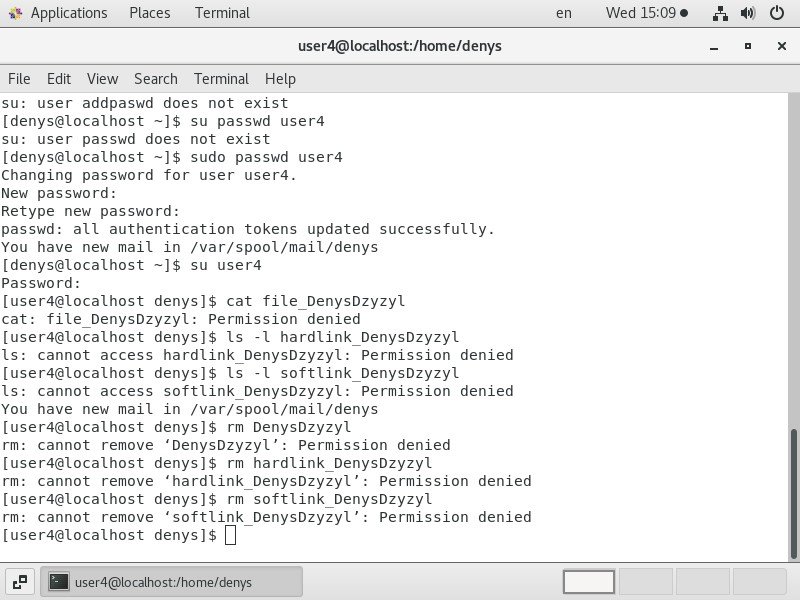
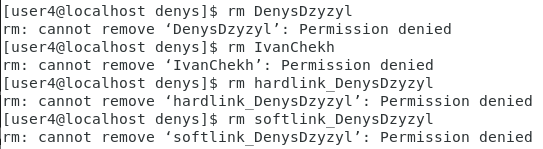
The chmod command in numeric mode uses three numbers to set permissions on a file or directory. Each digit represents access rights for the owner, group, and other users in that order. The value of each digit can be from 0 to 7, indicating a combination of three bits. For example, the number 4 represents the right to read. If all permissions are set to none, any attempt to set the permissions will fail.

When we use the command "chmod 4 emptyfile", the number 4 indicates that the owner of the file has the right to read it. However, if all the permissions of the file were previously reset (chmod 000), then any attempt to set the permissions of the file will fail. Therefore, in such a case, running the command "chmod 4 emptyfile" will not change anything in the file's permissions, as all permissions are currently set to none.

On the other hand, when we use "chmod 44 emptyfile", the first number indicates the read permission for the owner of the file, and the second number indicates the read permission for the group to which the file belongs. But since I have already reset all the permissions by running "chmod 000" before this command, "chmod 44 emptyfile" won't change anything.

* створіть каталог під назвою, де всі файли автоматично будуть належати Вашій групі користувачів і можуть бути видалені лише користувачем, який їх створив?
* під кожним користувачем створіть по одному новому файлу, та жорстке та символічне посилання на нього;



* спробуйте іншими користувачами переглянути ці файли;
* спробуйте іншими користувачами видалити ці файли, зробіть висновки.

Conclusion: Access and the ability to delete files by other users depend on the access rights to those files and the rights of the users attempting to delete them.

**Відповіді на контрольні запитання**

***Готував матеріал студент Чех І.В.***

**Контрольні запитання**

*1. Наведіть приклади зміни прав доступу символічним методом (Symbolic Method)?*

Change file permissions to read/write

chmod a+rw myfile.txt

Change the access rights for the directory to write for all users

chmod o+w mydir

Change file permissions to read for user and group

chmod u+r,g+r myfile.txt

*2. Наведіть приклади зміни прав доступу числовим методом (numeric method, octal method)?*

Change file permissions to read/write

*chmod 666 myfile.txt*

Change the access rights for the directory to write for all users

*chmod 777 mydir*

Change file permissions to read for user and group

*chmod 644 myfile.txt*

*3. Чи можна виконати файл, для якого є права на виконання, але не встановлені права на читання (--x)? Поясніть.*

No, the file cannot be executed unless it has read permissions. A file with executable permissions means that the user can run the file as a program. However, in order to run the application, the user must have access to the contents of the application, and for this they need read permissions.

*4. Яке призначення команди umask?*

The umask command is used to set a mask value that is used to calculate the actual access rights to files that are created by the user. The default mask value is 022, which means that new files are set to 0666 permissions.

*5. Якщо ми змінюємо права доступу та дозволи в поточній сесії чи будуть вони збережені в наступній?*

Yes, the access rights and permissions we change in the current session will be preserved in the next session. However, if we change the mask value using the umask command, this value will be applied to all new files that will be created in the next session.

*6. Чи є якийсь шаблон, яким система користується щодо прав та доступів при створенні нових файлів. Як можна змінити права дозволу за замовчуванням?*

Yes, the system uses the mask value specified by the umask command to calculate the actual access rights to files that are created by the user. The default mask value is 022, which means that new files are set to 0666 permissions.

To change the default permissions, you need to change the mask value using the umask command. For example, to set access rights for new files 0644, you need to run the following command:

umask 022

*7. Уявіть, що програмі потрібно створити одноразовий тимчасовий файл, який більше ніколи не знадобиться після закриття програми. Який правильний каталог для створення цього файлу?*

You need to use the /tmp directory to create a one-time temporary file that will never be needed again after the program is closed. This directory is for temporary files that are created by programs and are deleted when they are finished.

*8. Яким чином можна створити жорстке посилання? В яких ситуаціях їх доцільно використовувати?*

A hard link is created using the ln command with the -s option:

ln -s original reference

A hard link is actually a link to another file. This means that the hard link and the file it links to have the same inode. Sometimes this can be useful, for example, if you need to create multiple links to the same file.

*9. Яким чином можна створити символічне посилання? В яких ситуаціях їх доцільно використовувати?*

Symbolic links can be created using the ln command without the -s option.

Symbolic links are useful for creating aliases for files, creating desktop shortcuts, or moving files or directories between directories without actually moving them.

*10. Порівняйте жорсткі та символічні посилання?*

|  |  |  |
| --- | --- | --- |
| Характеристика | Жорстке посилання | Символічне посилання |
| Тип | Файл | Файл |
| Вміст | Посилання на інший файл | Посилання на шлях до іншого файлу |
| Індексний номер | Ідентичний файлу, на який посилається | Відрізняється від файлу, на який посилається |
| Взаємозв'язок | Файл і посилання є одним і тим же файлом | Файл і посилання є окремими файлами |
| Права доступу | Ідентичні файлу, на який посилається | Ідентичні файлу, на який посилається |
| Розмір | Ідентичний файлу, на який посилається | Ідентичний файлу, на який посилається |
| Переміщення | Неможливо перемістити файл і посилання в різні каталоги | Можна перемістити файл і посилання в різні каталоги |
| Видалення | Якщо видалити файл, на який посилається жорстке посилання, то жорстке посилання також буде видалено | Якщо видалити файл, на який посилається символічне посилання, то символічне посилання залишиться, але буде вказувати на неіснуючий файл |

*11. Є файл оригінал та для нього створено два посилання - символічне та жорстке. Що відбудеться з іншими файлами, якщо видалити:*

*- файл оригінал;*

If you delete the original file, both the hard link and the symbolic link will be deleted.

*- символічне посилання;*

If you delete a symbolic link, it simply ceases to exist. The original file will remain intact.

*- жорстке посилання.*

If you delete a hard link, the original file will also be deleted. This is because the hard link and the original file are the same file.

**Висновки**

During the implementation of the practical task of learning the basics of Linux, the set goals were achieved. Hands-on skills with the Bash shell were acquired, as well as basic knowledge of file permissions and special directories and files in Linux.