“Київський фаховий коледж зв’язку”

Циклова комісія комп’ютерної та програмної 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-23А

Команда КГК:

Корольов Є.Ю.,

Горохов Д.С. та

Коваленко С.О.

Перевірила викладач

Сушанова В.С.

Київ 2024

**Мета роботи:**

1. Отримання практичних навиків роботи з командною оболонкою Bash.
2. Знайомство з базовими діями при зміні власників файлів, .прав доступу до файлів
3. Знайомство з спеціальними каталогами та файлами в Linux.

**Матеріальне забезпечення занять**

1. ЕОМ типу IBM PC.

2. ОС сімейства Windows та віртуальна машина Virtual Box (Oracle).

3. ОС GNU/Linux (будь-який дистрибутив).

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

**Завдання для попередньої підготовки:**

***Готував матеріал студент Корольов Є. та Коваленко С.***

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

|  |  |
| --- | --- |
| Термін | Опис та Команди |
| Owner | Власник файлу чи каталогу.  Команда: chown user file.txt |
| Group | Група користувачів, які мають доступ до файлу.  Команда: chown :group file.txt |
| Other | Інші користувачі, які не є власниками чи членами групи. |
| Read (r) | Дозвіл на читання файлу чи каталогу.  Команда: chmod u+r file.txt |
| Write (w) | Дозвіл на зміну вмісту файлу.  Команда: chmod g+w file.txt |
| Execute (x) | Дозвіл на виконання файлу або доступ до каталогу.  Команда: chmod o+x script.sh |
| Permission string | Послідовність символів, що відображає права доступу (rwxr-xr--). |
| chown | Зміна власника файлу чи групи.  Приклад: chown user:group file.txt |
| chmod | Зміна прав доступу до файлу чи каталогу.  Приклад: chmod 755 file.txt |
| Special directories | Спеціальні каталоги, як /etc, /home, /var. |

1. Вивчіть матеріали онлайн-курсу “NDG Linux Essentials” (netacad.com):

* Chapter 17 - Ownership and Permissions
* Chapter 18 - Special Directories and Files

1. Пройдіть тестування у курсі NDG Linux Essentials за такими темами:

* Chapter 17 Exam
* Chapter 18 Exam

1. На базі розглянутого матеріалу дайте відповіді на наступні питання:
   1. Яке призначення команди id?
   2. Як переглянути які права доступу має власник файлу?
   3. \*Як змінити власника групи?
   4. \*Як можна переглянути у терміналі який тип поточного файлу? Наведіть приклади для різних типів файлів
   5. \*\*Для чого використовуються дозволи Setuid та Setgid?
   6. \*\*Для чого в системі потрібен так званий “липкий біт” (Sticky Bit). Наведіть приклади коли цей дозвіл доцільно використовувати.
2. Підготувати в електронному вигляді початковий варіант звіту:

* Титульний аркуш, тема та мета роботи
* Словник термінів
* Відповіді на п.4.1-4.6 з завдань для попередньої підготовки

**Хід роботи:**

***Готував матеріал студент Корольов Є. та Горохов Д.(401 ауд.)***

1. Початкова робота в CLI-режимі в Linux ОС сімейства Linux:
2. Запустіть операційну систему Linux Ubuntu. Виконайте вхід в систему та запустіть термінал ***(якщо виконуєте ЛР у 401 ауд.)***.
3. Запустіть віртуальну машину Ubuntu\_PC ***(якщо виконуєте завдання ЛР через академію netacad)***
4. Запустіть свою операційну систему сімейства Linux ***(якщо працюєте на власному ПК та її встановили)*** та запустіть термінал.
5. Опрацюйте всі приклади команд, що представлені у лабораторних роботах курсу NDG Linux Essentials - Lab 15: System and User Security та Lab 16: Creating Users and Groups. Створіть таблицю для опису цих команд

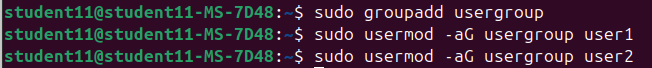
|  |  |
| --- | --- |
| Назва команди | Її призначення та функціональність |
| id | The id command can be useful for verifying which user account you are using and which groups you have available to use. By viewing the output of this command, you can see the user's identity information expressed both as a number and as a name. |
| groups | The output of the groups command may not be as detailed as the output of the id command, but if all you need to know is what groups you can switch to by using the newgrp command, then the groups command provides the information that you need. The id command output does show your current primary group, so it is useful for verifying that the newgrp command succeeded. |
| chgrp | To change the group owner of an existing file the chgrp command can be used. |
| chown | The chown command allows the root user to change the user ownership of files and directories. A regular user cannot use this command to change the user owner of a file, even to give the ownership of one of their own files to another user. However, the chown command also permits changing group ownership, which can be accomplished by either root or the owner of the file. |
| umask | The umask command is a feature that is used to determine default permissions that are set when a file or directory is created. Default permissions are determined when the *umask value* is subtracted from the maximum allowable default permissions |
| passwd | Now consider the passwd command. When this command runs, it modifies the /etc/shadow file, which seems impossible because other commands that the sysadmin user runs that try to access this file fail. |

* 1. Виконайте наступні практичні завдання у терміналі наступні дії (продемонструвати скріншоти):
* створіть трьох нових користувачів;

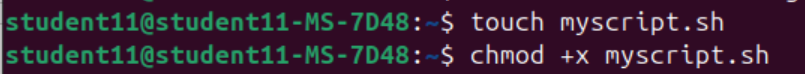




* створіть нову групу користувачів, туди додайте двох, з трьох створених користувачів;

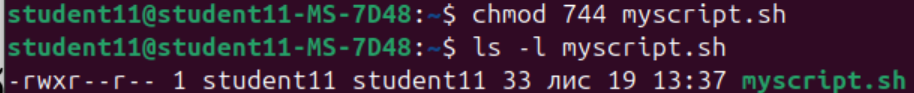


* створіть новий файл, який буде доступний на зчитування, редагування та виконання власником файлу, наприклад найпростіший скриптовий сценарій;



* 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;
* директорія, доступна для всіх трьох користувачів:







* директорія, доступна лише для власника:



* директорія, для перегляду групи, без редагування:





* \*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?

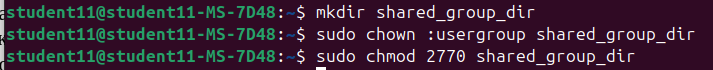


000: усі доступи(читання, запис, виконання) заборонені.

chmod 4 emptyfile - власник отримує читання, група і інші залишаються без доступу.

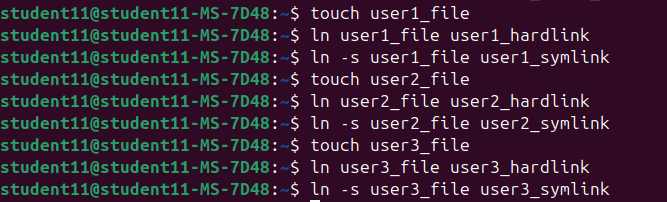
chmod 44 emptyfile - (040) група отримує читання, (004) інші отримують читання.

* \*\*створіть каталог під назвою, де всі файли автоматично будуть належати Вашій групі користувачів і можуть бути видалені лише користувачем, який їх створив?



2770 - встановлює setgrid змушує всі файли успадкувати групу каталогу. Тобто усі файли належать групі usergroup.

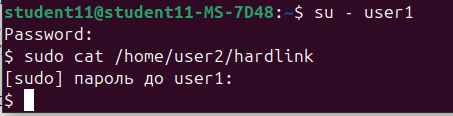
* \*\*під кожним користувачем створіть по одному новому файлу, та жорстке та символічне посилання на нього;



* \*\*спробуйте іншими користувачами переглянути ці файли;

Перед виконання користувачам необхідно надати права доступу:





файл з жорстким посиланням пустий, але переглянути іншими користувачами можливо.

* \*\*спробуйте іншими користувачами видалити ці файли, зробіть висновки.



chmod works with three types of permissions: owner, group, and other.

setgid makes files inherit a directory group.

Hard links allow you to work with files even after deleting the original.

Only the owner or a user with the appropriate rights can delete a file.

**Контрольні запитання:**

***Готувала матеріали студентка Коваленко С.***

1. Наведіть приклади зміни прав доступу символічним методом (Symbolic Method)?
2. Наведіть приклади зміни прав доступу числовим методом (numeric method, octal method)?
3. Яке призначення команди umask?
4. Порівняйте жорсткі та символічні посилання?
5. \*Чи можна виконати файл, для якого є права на виконання, але не встановлені права на читання (--x)? Поясніть.
6. \*Якщо ми змінюємо права доступу та дозволи в поточній сесії чи будуть вони збережені в наступній?.
7. \*Чи є якийсь шаблон, яким система користується щодо прав та доступів при створенні нових файлів. Як можна змінити права дозволу за замовчуванням?
8. \*Яким чином можна створити жорстке посилання? В яких ситуаціях їх доцільно використовувати?
9. \*Яким чином можна створити символічне посилання? В яких ситуаціях їх доцільно використовувати?
10. \*\*Уявіть, що програмі потрібно створити одноразовий тимчасовий файл, який більше ніколи не знадобиться після закриття програми. Який правильний каталог для створення цього файлу?
11. \*\*Є файл оригінал та для нього створено два посилання - символічне та жорстке. Що відбудеться з іншими файлами, якщо видалити:

* файл оригінал;
* символічне посилання;
* жорстке посилання.

Виснови: