“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 тафайли в Linux”**

Виконали студенти

Групи БІКС-13

Когут Ігор

Береза Артем

Перевірив викладач

Сушанова В.С.

**Завдання для попередньої підготовки:**

1. Команда `id` призначена для відображення інформації про ідентифікатори користувача та групи. Вона виводить ідентифікатор користувача (UID), ідентифікатор головної групи (GID) та список додаткових груп, до яких належить користувач.

2. Щоб переглянути права доступу до файлу, можна використати команду ls -l.

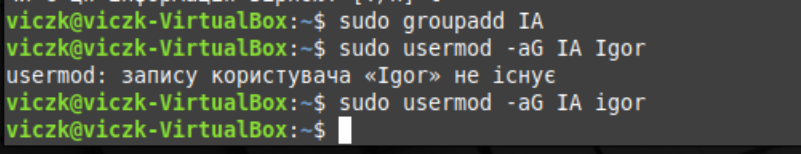
3. Щоб змінити власника групи, можна використати команду chown.

4. Щоб переглянути тип поточного файлу в терміналі, можна використати команду file.

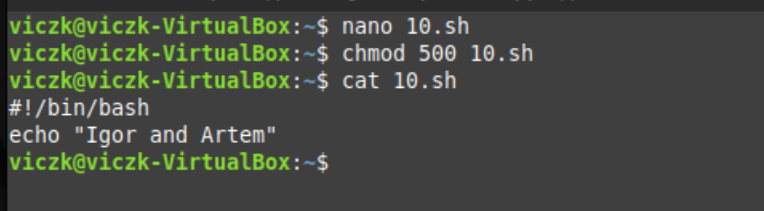
5. Дозволи Setuid та Setgid надають спеціальні привілеї під час виконання виконуваних файлів. Setuid дозволяє виконувати файл з привілеями власника файлу, а Setgid дозволяє виконувати файл з привілеями групи власника файлу. Ці дозволи зазвичай використовуються для виконання виконуваних файлів, які потребують підвищених привілеїв, наприклад, програм, які працюють з системними ресурсами.

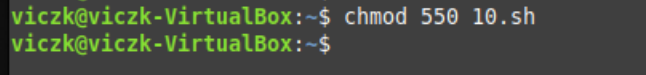
6. Липкий біт використовується для обмеження прав доступу до каталогу. Коли липкий біт встановлено для каталогу, тільки власник файлу, власник каталогу та адміністратор системи можуть видаляти або переміщати файли в цьому каталозі. Це дозволяє захистити важливі файли в спільних каталогах від ненавмисних видалень або замін файлів іншими користувачами. Наприклад, липкий біт часто використовується для каталогу /tmp, щоб забезпечити безпеку тимчасових файлів.

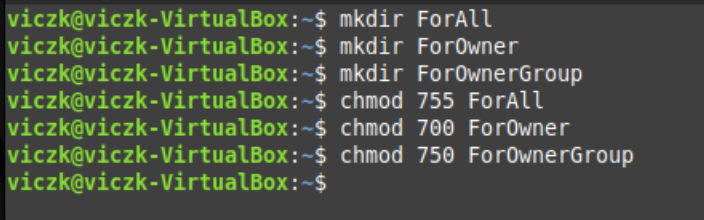
**Хід роботи:**

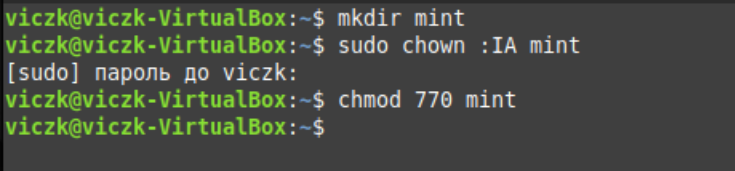












**Контрольні запитання:**

1. Приклади зміни прав доступу символічним методом:

chmod u+r filename

chmod g-w filename

chmod o+x filename

chmod a=rwx filename

2. Приклади зміни прав доступу числовим методом:

chmod 755 filename

chmod 644 filename

chmod 777 filename

3. Команда umask використовується для встановлення маски дозволів за замовчуванням для новостворених файлів та каталогів.

4. Жорстке та символічне посилання є двома типами посилань на файли в файловій системі. Основна відмінність полягає в тому, що жорстке посилання прив'язується безпосередньо до іноду файлу, тоді як символічне посилання просто містить шлях до файлу.

5. Ні, не можна виконати файл, для якого є права на виконання, але не встановлені права на читання (--x). Читання файлу є обов'язковим для виконання його вмісту, тому якщо відсутній доступ на читання, файл не може бути виконаний.

6. Зміни прав доступу та дозволів, виконані в поточній сесії, залишаються активними до закінчення цієї сесії. Якщо ви вийдете з поточної сесії або перезавантажите систему, зміни будуть скинуті на значення, встановлені за замовчуванням.

7. Система Linux використовує шаблони umask для встановлення дозволів за замовчуванням для новостворених файлів та каталогів. Змінити ці значення можна змінивши параметри umask в файлі конфігурації shell (наприклад, ~/.bashrc або /etc/profile).

8. Жорстке посилання створюється за допомогою команди ln з опцією -s. Вони доцільно використовуються, коли потрібно мати два і більше імені для одного файлу в тій же файловій системі.

9. Символічне посилання створюється за допомогою команди ln без будь-яких опцій. Вони доцільно використовуються, коли потрібно створити посилання на файл або каталог в будь-якій частині файлової системи, включаючи інші файлові системи або мережу.

10. Для створення одноразового тимчасового файлу, який більше ніколи не знадобиться після закриття програми, найкраще використовувати каталог /tmp, оскільки він призначений для тимчасових файлів, які можуть бути видалені після використання.

11. Якщо файл оригінал видаляється, то інші файли, які є посиланнями на нього (як жорсткі, так і символічні), залишаться, проте вони стануть некорисними, оскільки оригінал вже відсутній. Якщо видалити символічне посилання, то оригінал файлу залишиться без змін. Але, якщо видалити жорстке посилання, то оригінал файлу також буде видалено, оскільки жорстке посилання прив'язане безпосередньо до іноду файлу.