**Лабораторна робота № 3**

***Тема:*** Структура файлової системи UNIX, основні команди, команди роботи з файлами (man, passwd, date, cat, more, wc, who, ls, cd, cal, cp, mv, mkdir, rm, rmdir).

***Мета роботи:*** вивчити основні команди роботи з файлами в UNIX-подібних ОС.

### Вивчення файлової системи і функцій по обробці і управлінню даними.

# Мета роботи:

Метою роботи є вивчення структури файлової системи ОС Linux, вивчення команд створення, видалення, модифікації файлів і каталогів, функцій маніпулювання даними.

# Теоретична частина

## Файлова структура системи Linux

В операційній системі Linux файлами вважаються звичайні файли, каталоги, а також спеціальні файли, що зв’язані з периферійними пристроями (кожний пристрій представляється у вигляді файлу). Доступ до всіх файлів однотипний, в тому числі, і до файлів периферійних пристроїв. Такий підхід забезпечує незалежність програми користувача від особливостей вводу/виводу на конкретний зовнішній пристрій.

Файлова структура Linux має ієрархічну деревоподібну структуру. В кореневому каталозі розміщуються інші каталоги і файли, включаючи 5 основних каталогів:

* bin – більшість виконувальних командних програм і *shell* - процедур;
* tmp – тимчасові файли;
* usr – каталоги користувачів (умовне позначення);
* etc – переважно адміністративні утиліти і файли;
* dev – спеціальні файли, які представляють периферійні пристрої; при додванні периферійного пристрою в каталог /dev повинен бути доданий відповідний файл (коса / означає приналежність кореневому каталогу).

Поточний каталог – це каталог, в якому на даний момент знаходиться користувач. При наявності прав доступу користувач може перейти після входу в систему в інший каталог. Поточний каталог позначається крапкою (**.**); батьківський каталог, до якого належить поточний, позначається двома крапками (**..**).

Повне ім’я може включати імена каталогів, включаючи кореневий, розділений косою лінією, наприклад: /home/student/file.txt. Перша коса лінія позначає кореневий каталог, і пошук файлу буде починатися з нього, а потім в каталозі home, потім в каталозі student.

Один файл можна зробити приналежним декільком каталогам. Для цього використовується команда **ln (link)**:

# ln <ім’я файла 1> ім’я файла 2>

Ім’я 1-го файла – це повне ім’я файлу, на який встановлюється зсилка; ім’я 2-го файла – це повне ім’я файлу в новому каталозі, де буде використовуватися ця зсилка. Нове ім’я може не відрізнятися від старого. Кожний файл може мати декілька зсилок, тобто він може використовуватися в різних каталогах під різними іменами.

Команда **ln** з аргументом -s створює символічну зсилку:

# ln -s <ім’я файла 1> <ім’я файла 2>

Тут ім’я 2-го файлу є символічною зсилкою. Символічна зсилка є особим видом файлу, в якому зберігається ім’я файлу, на який символічна зсилка зсилається. Linux працює з символічною зсипкою не так, як із звичайним файлом – наприклад, при виводі на екран вмісту символічної зсилки з’являться дані файлу, на який ця символічна зсилка зсилається:

В Linux розрізняють 3 рівні доступу до файлів і каталогів:

1. доступ власника файлу;
2. доступ групи користувачів, до якої належить власник файлу;
3. решта користувачів.

Для кожної групи існуються свої байти атрибутів, значення яких розшифровується наступним чином:

r – дозвіл на читання;

w – дозвіл на запис;

x – дозвіл на виконання;

- – відсутність дозволу.

Перший символ байту атрибутів визначає тип файлу і може інтерпретуватися з наступними значеннями:

- – звичайний файл;

d – каталог;

l – символічна зсилка;

в – блок-орієнтований спеціальний файл, який відповідає таким периферійним пристроям, як накопичувачі на дисках;

с – байт-орієнтований спеціальний файл, який може бути пов’язаний з такими периферійними пристроями як принтер, термінал.

В домашньому каталозі користувач має повний доступ до файлів (READ, WRITE, EXECUTE; r, w, x).

Атрибути файлу можна переглянути командою **ls -l** і вони представляються в наступному форматі:

d rwx rwx rwx

| | | |

| | | | Доступ для решти користувачів

| | | Доступ до файлу для членів групи

| | Доступ до файлу власника

| Тип файлу (директорія)

Приклад. Командою **ls -l** отримаємо лістинг вміст поточної директорії student:

* rwx --- --- 2 student 100 Mar 10 10:30 file\_1
* rwx --- r-- 1 adm 200 May 20 11:15 file\_2
* rwx --- r-- 1 student 100 May 20 12:50 file\_3

Після байтів атрибутів на екран виводиться наступна інформація про файл:

* число зсилок файлу;
* ім’я власника файлу;
* розмір файлу в байтах;
* дата створення файлу (або модифікації);
* час;
* ім’я файлу.

Атрибути файлу і доступ до нього, можна змінити командою:

# chmod <коди захисту> <імя файлу>

Коди захисту можуть бути задані в числовому або символьному вигляді. Для символьного коду використовуються:

знак плюс (+) – додати права доступу;

знак мінус (-) – відмінити права доступу;

r,w,x - доступ на читання, запис, виконання;

u,g,o – власника, групи, решти.

Коди захисту в числовому вигляді можуть бути задані у вісімковій формі. Для контролю встановленого доступу до свого файлу після кожної зміни коду захисту потрібно перевіряти свої дії за допомогою команди **ls -l.**

Приклади:

**chmod g+rw,o+r file.1** – встановлення атрибутів читання і запису для групи і читання для всіх решти користувачів;

**ls -l file.1** – читання атрибутів файлу;

**chmod o-w file.1** – відмінна атрибуту запису для решти користувачів;

**>letter** – створення файлу letter. Символ > використовується як для для пере адресації, так і для створення файлу;

**cat** – вивід вміст файлу;

**cat file.1 file.2 > file.12** – конкатенація файлів (об’єднання);

**mv file.1 file.2** – перейменування файлу file.1 в file.2;

**mv file.1 file.2 file.3 directory** – переміщення файлів file.1, file.2, file.3 у вказану директорію;

**rm file.1 file.2 file.3** – видалення файлів file.1, file.2, file.3;.

**cp file.1 file.2** – копіювання файлу з перейменуванням;

**mkdir namedir** – створення каталогу;

**rm dir\_1 dir\_2** – видалення каталогів dir\_1 dir\_2;

**ls [acdfgilqrstv CFR] namedir** – вивід вмісту каталогу; якщо в якості namedir вказано ім’я файлу, то виводиться вся інформація про цей файл. Значення аргументів:

* l список включає всю інформацію про файли;
* t сортування по часу модифікації файлів;
* a в список включаються всі файли, в тому числі і ті, які починаються з крапки;
* s розміри файлів вказуються в блоках;
* d вивести ім’я самого каталогу, але не вміст;
* r сортування рядків виводу;
* i вказати ідентифікаційний номер кожного файлу;
* v сортування файлів по часу останнього доступу;
* q символи, що не виводяться замінити на знак ?;
* с використовувати час створення файлу при сортуванні;
* g те саме, що -l, але з вказуванням імені групи користувачів;
* f вивід вмісту всіх вказаних каталогів, відміняє прапорці -l, -t, -s, -r і активізує прапорець -а;
* С вивід елементів каталогу в декілька стовбців;
* F додавання до імені каталогу символу / і символу \* до імені файлу, для яких дозволено виконання;
* R рекурсивний вивід вмісту підкаталогів заданого каталогу.

**cd <namedir>** - перехід інший каталог. Якщо параметри не вказані, то

відбувається перехід в домашній каталог користувача.

**pwd** – вивід імені поточного каталогу;

# grep [-vcilns] [шаблон пошуку] <ім’я файла> - пошук файлів з вказуванням

або без вказування контексту (шаблону пошуку).

Значення ключів:

* v – виводяться рядки, які не містять шаблон пошуку;
* c – виводиться тільки число строк, які містять або не містять шаблон;
* i – при пошуку не відрізняються великі і малі букви;
* l – виводяться тільки імена файлів, які містять вказаний шаблон; - n – перенумерувати вивідні рядки;
* s – формується тільки код завершення.

Приклади.

1. Вивести імена всіх файлів поточного каталогу, які містять послідовність "student" і мають розширення .txt:

**grep -l student \*.txt**

1. Визначити ім’я користувача, який входить в ОС Linux з терміналу tty23:

**who | grep tty23**

# Порядок виконання роботи

1. Ознайомитися з файловою структурою ОС Linux. Вивчити команди роботи з файлами.
2. Використовуючи команди ОС Linux, створити два текстових файли.
3. Отримані файли об’єднати в один файл і його вміст переглянути на екрані.
4. Створити новий каталог і перемістити до нього отримані файли.
5. Вивести повну інформацію про всі файли і проаналізувати рівні доступу.
6. Додати для всіх трьох файлів право виконання членам групи і решти користувачів.
7. Переглянути атрибути файлів.
8. Створити ще один каталог.
9. Встановити додатковий зв'язок об’єднаного файлу з новим каталогом, але під іншим іменем.
10. Створити символічний зв'язок.
11. Зробити поточним новий каталог і вивести на екран розширений список інформації про його файли.
12. Виконати пошук заданої послідовності символів в файлах поточної директорії і отримати перелік відповідних файлів.
13. Отримати інформацію про активні процеси та імена інших користувачів.
14. Зробити звіт про виконану роботу і видалити свої файли і каталоги.
15. Вийти із системи.

# Індивідуальні завдання для виконання

* 1. Завантажтеся в систему під вашим користувацьким ім'ям.
  2. Поміняйте ваш пароль. Ваш новий пароль повинен включати в себе як частину номер Вашої залікової книжки.
  3. Виведіть системну дату.
  4. Підрахуйте кількість рядків у файлі:

|  |  |
| --- | --- |
| Варіант | Файл |
| 1, 2, 4 | /etc/passwd |
| 3, 10 | /etc/group |
| 6, 9 | /etc/profile |
| 5, 7, 8 | /etc/fstab |

* 1. Виведіть на екран вміст відповідного файлу.
  2. Виведіть календар на <1995+№варіанту> рік.
  3. Виведіть календар на 1752 рік. Чи не помічаєте що-небудь цікаве у вересні?

Поясніть.

* 1. Визначте, хто ще завантажений у систему.
  2. Скопіюйте *(скопіюйте, а не перемістіть, бо система перестане працювати коректно!)* файли

|  |  |  |
| --- | --- | --- |
| варіант | файл 1[[1]](#footnote-1) | файл 2 |
| 1 | /bin/cat | /bin/at |
| 2 | /bin/cal | /bin/chmod |
| 3 | /bin/ls | /bin/chown |
| 4 | /bin/tee | /bin/file |
| 5 | /bin/more | /bin/gzip |
| 6 | /bin/date | /bin/gunzip |
| 7 | /bin/cp | /bin/ps |
| 8 | /bin/mv | /bin/csh |
| 9 | /bin/lpr | /bin/sh |
| 10 | /bin/find | /bin/ksh |

у ваш домашній каталог різними способами.

* 1. Створіть каталог lab\_1.
  2. *Скопіюйте* в нього з вашого домашнього каталогу копію файлу 1, яку ви отримали в п.9, під ім'ям my\_<ім’я файлу 1>. *Перемістіть* в цей каталог з вашого домашнього каталогу копію файлу 2, яку ви отримали в п.9, перейменувавши його при цьому в my\_<ім’я вихідного файлу 2>. За ім’я вихідного файлу слід брати саме ім’я файлу, без імен каталогів і шляху до файлу (інакше символ "/" буде проінтерпретований операційною системою зовсім не так, як Ви очікуєте).
  3. Перейдіть у свій домашній каталог і переконайтеся в тому, що все зроблено правильно.
  4. Створіть каталог lab\_1\_<№варіанту> і перейдіть в нього.
  5. Скопіюйте в каталог lab\_1\_<№варіанту> файл з п.4 під ім'ям n<ім’я вихідного файлу>.
  6. За допомогою команд cat і more перегляньте його вміст.
  7. Перейдіть у свій домашній каталог.
  8. Видаліть каталог lab\_1\_<№варіанту>.

1. Якщо файл 1 або 2 не знайдено в каталозі /bin, шукайте його в каталогах /usr/bin, /sbin або /usr/sbin [↑](#footnote-ref-1)