"Kyiv Professional College of Communications" Computer Engineering Cycle Commission

**Laboratory work №3**

**Theme:** “Familiarity with basic CLI-mode commands in Linux”

in the discipline: "Operating Systems"

Performed by students

prepared by students:

KSM-33 group

Team:

Koval Ivan Volodymyrovych,

Kuzmenko Denys Maksymovych

The teacher checked:

Sushanova V.S.

**Objective:**

1. Introduction to basic CLI commands in Linux.
2. Introduction to basic text commands in terminal mode in different OSes.

**Material support for classes:**

1. IBM PC type computer.

2. Windows family OS and Virtual Box virtual machine (Oracle).

3. GNU/Linux OS (any distribution).

4. Cisco Network Academy website netacad.com and its online Linux courses

**Словник базових англійських термінів**

CLI  - Інтерфейс командного рядка. Спосіб взаємодії з операційною системою шляхом введення текстових команд.

ShellОболонка - Програма, яка перекладає команди, введені користувачем, у дії, що виконує операційна система.

Bourne Again SHell -Найпоширеніша оболонка для дистрибутивів Linux.

Prompt - Рядок запрошення важлива частина інтерфейсу терміналу, що відображається перед введенням команди.

Command - Команда. Програмне забезпечення, яке при виконанні в CLI виконує певну дію на комп'ютері.

Option - Параметр. Використовується для розширення або зміни основної поведінки команди.

Argument - Використовується для надання додаткової інформації команді .

Variable - Функція, що дозволяє користувачу або оболонці зберігати дані.

Command line - Можливість оболонки Bash зберігати попередньо введені команди.

**Завдання для попередньої підготовки:**

1. \*Прочитайте короткі теоретичні відомості до лабораторної роботи та зробіть невеликий словник базових англійських термінів з питань призначення команд та їх параметрів.
2. Вивчіть матеріали онлайн-курсу академії Cisco “NDG Linux Essentials”:

* Chapter 5 - Command Line Skills
* Chapter 6 - Getting Help

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

* Chapter 05 Exam
* Chapter 06 Exam

1. \*Дайте визначення наступним поняттям:

* Командний інтерпретатор – це програма, яка дозволяє користувачу взаємодіяти з операційною системою через текстовий інтерфейс командного рядка. Він приймає текстові команди від користувача, інтерпретує їх і виконує відповідні дії, наприклад, запуск інших програм або керування файлами та завданням
* Оболонка – програмний посередник між користувачем та ядром операційної системи, який надає інтерфейс для взаємодії з комп'ютером. Вона може бути як текстовою (командною), де користувач вводить команди, так і графічною, з вікнами та мишкою, що робить роботу з ОС простішою та інтуїтивнішою
* Команда – це інструкція, яку користувач вводить у командний рядок або через інтерфейс, щоб виконати певну дію або запустити програму. Команди надають користувачам можливість взаємодіяти з операційною системою, управляти файлами, запускати додатки та налаштовувати систему

1. \*\*Дайте відповіді на наступні питання:

* **Яку базову інформацію надає рядок запрошення prompt?**

Рядок запрошення промт надає таку базову інформацію:

* ім’я користувача;
* назву комп’ютера (хоста);
* поточний каталог;
* символ $ або #.
* **Для чого команді потрібні параметри та аргументи?**

Аргументи – це дані, над якими виконується команда (наприклад, ім’я файлу чи каталогу).

Параметри – змінюють поведінку команди (наприклад, показати докладно, відсортувати, відобразити приховані файли)

* **Яке призначення команд ls, які параметри та аргументи вона може мати? Наведіть 3 приклади.**

**ls** – виводить список файлів та каталогів у поточному або вказаному каталозі.

Поширені параметри:

* ls -l — детальний список (права доступу, власник, розмір, дата);
* ls -a — показати приховані файли (починаються з .);
* ls -h — розміри у зручному форматі (K, M, G).
* **Яким чином можна використати історію команд, які переваги це надає?**

Як можна використати:

* Перегляд історії
* Виконання команди за номером
* Пошук у історії: **Ctrl+R** і введення частини команди
* Використання стрілок **↑/↓** для прокрутки

Переваги:

* економія часу;
* зменшення кількості помилок при повторному наборі;
* зручний пошук і повторення складних команд.
* **Яке призначення команди echo?**

Команда echo виводить текст або значення змінної у термінал.

* **Охарактеризуйте поняття змінної в оболонці Bash, які типи змінних вона підтримує?**

**Змінна** – це іменоване місце у пам’яті, де зберігається значення (рядок, число, шлях тощо).

**Типи змінних у Bash:**

* **Локальні** – діють лише в межах поточної оболонки або скрипта.
* **Глобальні (середовища)** – доступні для всіх процесів, що запускаються з оболонки.
* **Позиційні параметри** (аргументи скрипта).
* **Спеціальні змінні** (службова інформація).
* **Яке призначення команд env, export та unset?**

**env** – показує всі змінні середовища або запускає програму з новим набором змінних.  
**export** – робить змінну локальної оболонки глобальною (видимою іншим процесам).

**unset** – видаляє змінну.

* **Які команди для отримання довідки по командам в терміналі ви знаєте?**
* **info <команда>** — розширена документація.
* **help <команда>** — для вбудованих команд оболонки.

1. Підготувати в електронному вигляді початковий варіант звіту:

* Титульний аркуш, тема та мета роботи
* Словник термінів
* Відповіді на п.4 та п.5 з завдань для попередньої підготовки

**Хід роботи:**

1. Опрацюйте всі приклади команд, що представлені у лабораторній роботі курсу ***NDG Linux Essentials - Lab 5: Command Line Skills*** та ***Lab 6: Getting Help.*** Створіть таблицю для опису цих команд

|  |  |
| --- | --- |
| Назва команди | Її призначення та функціональність |
| **ls** | Виводить список файлів і каталогів у поточному каталозі |
| **ls -l** | Детальний список файлів (права, власник, розмір, дата) |
| **ls -l /tmp** | Показує детальну інформацію про файли у каталозі /tmp |
| **ls -l /home** | Відображає детальну інформацію про файли у каталозі /home |
| **whoami** | Відображає ім’я поточного користувача |
| **uname** | Відображає інформацію про ядро операційної системи |
| **uname -n / uname --nodename** | Показує ім’я хоста мережевого вузла |
| **pwd** | Відображає поточний робочий каталог |
| **history** | Відображає нумерований список введених команд |
| **!number** | Виконує команду з історії за номером |
| **echo** | Використовується для виведення тексту або значення змінної |
| **echo $PATH** | Виводить вміст змінної PATH (шляхи пошуку програм) |
| **``echo Today is `date```** | Виконує команду date і виводить результат |
| **echo D\*** | Використовує метасимволи для відображення файлів, що починаються з D |
| **echo "D\*"** | Виводить рядок D\* буквально, без розширення |
| **echo Hello; echo Linux; echo Student** | Виконує кілька команд послідовно |
| **false; echo Not; echo Conditional** | Виконує команди послідовно навіть після помилки |
| **echo Start && echo Going && echo Gone** | Виконує наступну команду тільки якщо попередня завершилась успішно |
| **echo Success && false && echo Bye** | При збої виконання зупиняє наступні команди, пов’язані && |
| **which date** | Показує шлях до виконуваного файлу date |
| **which ls** | Показує шлях до виконуваного файлу ls |
| **type command** | Визначає тип команди (вбудована, зовнішня, псевдонім) |
| **type -a ls** | Показує всі варіанти команди ls |
| **type vi** | Вказує, де знаходиться команда vi |
| **cd /bin** | Змінює поточний каталог на /bin |
| **cd** | Повертає користувача у домашній каталог |
| **type vlc** | Перевіряє наявність команди vlc у PATH |
|  |  |
|  |  |

1. Робота в в терміналі (закріплення практичних навичок) **обов'язково представити свої скріншоти:**

2.1. Робота зі змінними (Variables) та псевдонімами (Aliases) в терміналі:

* Створіть змінні, що будуть містити Ваші імена та прізвища $var\_name1, $var\_name2, $var\_name3
* За допомогою команди echo виведіть імена студентів вашої команди
* Створіть псевдоніми mycal1, mycal2, mycal3 для команди cal для автоматичного виведення календарю вашого року народження

2.2. \*Робота з функціями (Functions) в терміналі:

* Створіть функцію students\_report, що порядково буде виводити спочатку імена студентів Вашої команди, а потім роки їх народження

2.3. \*Робота з лапками (Quoting) в терміналі. Виведіть в командному рядку наступні речення:

* “We create such variables as $var\_name1, $var\_name2, $var\_name3, which stored our names Name1, Name2, Name3” (у реченні спочатку виводимо назви змінних, а потім їх вміст)
* “We create such Aliases as mycal1, mycal2, mycal3, which can show our calendars: Calendar1, Calendar2, Calendar3” (у реченні спочатку виводимо назву команди-псевдонімів, потім вивід цих команд).

2.4. \*\*Робота з інструкціями керування (Control Statements) в терміналі:.

* Чи можна завдання 2.1 та 2.2 ходу роботи виконати через інструкції керування без написання окремої функції, як це буде виглядати?

2.5. Робота з командами довідки (Man Pages) в терміналі:

* На прикладі команди uname продемонструйте як отримати довідку. На основі отриманої додаткової інформації наведіть 5 різних варіантів виводу результату інформації по даній команді з використанням 5 різних параметрів (Options)

**Контрольні запитання:**

**1. Які типи команд існують в оболонці Bash?**

В оболонці Bash існує чотири основні типи команд: вбудовані команди, функції оболонки, аліаси та зовнішні команди (окремі виконувані файли, які шукаються у $PATH, як-от /bin/ls).

**2. Що таке змінні оточення? Як їх можна переглянути в терміналі?**

Змінні оточення – це іменовані значення, що зберігаються в пам'яті оболонки і автоматично доступні всім дочірнім процесам та командам, що виконуються в цьому середовищі. Їх використовують для конфігурації. Переглянути їх можна командами env або printenv, а значення конкретної змінної – за допомогою echo $ЗМІННА.

**3. Опишіть змінну $PS1. Як в терміналі переглянути її вміст?**

$PS1 — це змінна оточення, яка визначає формат і вміст рядка запрошення (prompt) у терміналі. Вона керує тим, яка інформація (користувач, хост, поточний каталог) відображається перед введенням кожної команди. Переглянути її вміст можна командою: echo $PS1.

**4. Як можна змінити значення змінної $PS1? Як змінити її значення не на поточний сеанс, а за замовчуванням?**

**З**мінити $PS1 на поточний сеанс можна простим присвоєнням нового значення, наприклад: PS1="[MyPrompt> ]". Рядок запрошення миттєво зміниться. Щоб змінити значення за замовчуванням (зберегти його після перезапуску), потрібно додати команду присвоєння до конфігураційного файлу оболонки, наприклад, до ~/.bashrc.

**5. Для чого використовують лапки в оболонці Bash?**

Лапки використовуються для екранування та контролю інтерпретації спеціальних символів оболонкою. Одинарні лапки забезпечують повне екранування — увесь текст сприймається буквально, змінні не інтерпретуються. Подвійні лапки забезпечують часткове екранування — зберігається інтерпретація змінних ($VAR) та командних підстановок, але інші спеціальні символи ігноруються.

**6. Для чого використовують інструкції керування, які їх види Ви знаєте?**

Інструкції керування використовуються в Bash-сценаріях для керування порядком виконання команд, дозволяючи програмі приймати рішення та повторювати дії. Основні види: умовні структури (if/elif/else) для виконання коду за певною умовою та циклічні структури (for, while, until) для багаторазового виконання блоку команд.

**7. В чому різниця якщо в кінці рядку запрошення bash стоїть символ $ чи #?**

Символ в кінці рядка запрошення вказує на права користувача:

* $  - означає, що ви працюєте як звичайний користувач з обмеженими правами.
* #  - означає, що ви працюєте як суперкористувач root (адміністратор) з повними правами доступу до системи.

**8. Яке призначення команд whereis та locate? Яка між ними відмінність?**

* whereis: Шукає виконувані файли та документацію (man) для вказаної команди в стандартних бінарних каталогах. Шукає "наживо" і дуже швидко.
* locate: Шукає будь-який файл чи каталог у всій системі. Шукає надзвичайно швидко, використовуючи попередньо створену базу даних (індекс), тому може не відображати файли, створені після останнього оновлення індексу.

**Conclusions**

During the work, the goal was successfully achieved - basic knowledge and practical skills of interacting with the operating system through the command line were obtained. It was learned that the Bash shell is a key interpreter that uses the prompt line to display system information. Four types of commands were studied (built-in, functions, aliases, external) and their execution was mastered using parameters and arguments for precise control of actions. Skills were acquired in working with the ls and echo commands for viewing and outputting data. In addition, mechanisms for managing environment variables using env, export and unset were mastered, which is critically important for setting up the working environment. Mastering the help systems (man, --help) and search tools (locate, whereis) provided the basis for independent study of new commands. Thus, this work laid the necessary foundation for further administration and effective use of Linux systems.