“Київський фаховий коледж зв’язку”

Циклова комісія Комп’ютерної інженерії

**ЗВІТ ПО ВИКОНАННЮ**

**ЛАБОРАТОРНОЇ РОБОТИ №3**

з дисципліни: «Операційні системи»

**Тема: «Знайомство з базовими командами CLI-режиму в Linux»**

Виконали

студенти

групи БІКС-13

Команда «chipichapa»:

Ємець В.О.,

Савустьян М.В.

Перевірив викладач

Сушанова В.С.

Робота студентів групи БІКС-13 Команда «chipichapa»: Ємець В., Савустьян М.

Київ 2024

**Мета роботи:**

1. Знайомство з базовими командами CLI-режиму в Linux.

2. Знайомство з базовими текстовими командами в термінальному режимі роботи в різних ОС.

**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**

***Готувала матеріал студентка*** *Ємець 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 |
| shell | оболонка |
| terminal | термінал |
| commands | команди |
| arguments | аргументи |
| options | параметри |
| variables | змінні |
| quoting | цитування |
| control statements | керуючі оператори |

4. \*Дайте визначення наступним поняттям:

- Командний інтерпретатор – is an application that provides user interaction with the operating system.

- Оболонка – is the command line interpreter that translates commands entered by a user into actions to be performed by the operating system.

- Команда – is a software program that when executed on the CLI, performs an action on the computer.

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

- Яку базову інформацію надає рядок запрошення prompt?

Typically the prompt contains information about the user and the system. The prompt shown contains the following information:

● Username (sysadmin)

● System name (localhost)

● Current Directory (~)

- Для чого команді потрібні параметри та аргументи?

Options are used to modify the core behavior of a command while arguments are used to provide additional information (such as a filename or a username).

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

The ls command in Linux is used to list directory contents of files and directories.

Examples: ls -l, ls -lr, ls -rl.

- Яким чином можна використати історію команд, які переваги це надає?

There are several ways to view the history of teams:

1) Use the up and down arrow keys to scroll through the previous commands, and then press enter to repeat the command

2) Use the history command

3) Use the grep command for advanced searching in the history

There are advantages to this: it saves time, so you don't need to memorise long commands, and you can use history to view a list of commands that have been used before. You can also fix errors by tracking down the commands that caused problems.

- Яке призначення команди echo?

The echo command in Linux is a built-in command that allows users to display lines of text or strings that are passed as arguments. It is commonly used in shell scripts and batch files to output status text to the screen or a file.

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

A variable is a feature that allows the user or the shell to store data. There are two types of variables used in the Bash shell: local and environment.

- Яке призначення команд env, export та unset?

The env command displays the current environment variables and their values.

The export command makes shell variables into environment variables. The unset command removes a variable or function value.

- Які команди для отримання довідки по командам в терміналі ви знаєте?

There are two main commands to get help on the usage of a Linux command:

1) help: For shell builtin commands

2) man: For other Linux commands

**Хід роботи**

***Готував матеріал студент***

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 інформації про каталоги та файли. За замовчуванням без аргументів відображає інформацію для поточного каталогу |
| ls -l | Використанні параметру **-l** в команді **ls** дозволяє відобразити інформацію про файли, розташовані в поточному робочому каталозі, у довгому форматі, який надає більш розширену додаткову інформацію |
| ls -l /tmp | Використання аргументу **/tmp** в поєднанні з параметром **-l** в команді **ls** дозволяє відобразити детальну інформацію про файли в каталозі /tmp. |
|  |  |
|  |  |
|  |  |

2. Робота 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**

***Готувала матеріал студентка Ємець В.***

1. Які типи команд існують в оболонці Bash?

The Bash shell has many popular features, a few of which are listed below:

● Command line history

● Inline editing

● Scripting: The ability to place commands in a file and then interpret (effectively use Bash to execute the contents of) the file, resulting in all of the commands being executed.

● Aliases: The ability to create short nicknames for longer commands.

● Variables: Used to store information for the Bash shell and for the user.

2. Що таке змінні оточення? Які вони бувають. Як їх можна переглянути в терміналі?

Environment variables are used to store app secrets and configuration data, which are retrieved by your running app when needed. There are three types:

1) System Environment Variables

System environment variables reside on the topmost root of the system and are the same for all processes running in a system under all the user profiles of the system. Your operating system/system administrator usually sets these, and you’re rarely required to fiddle with them.

2) User Environment Variables

User environment variables are those that are local to a user profile in Windows systems. These variables are used to store user-specific information such as the path to a local installation of libraries that are not to be used by all users, values specific to programs installed only for specific users, etc.

You do not need the system administrator to make changes to these variables; you can do that yourself as a user. These variables are helpful when implementing local changes in your system without affecting other users.

3) Runtime/Process Environment Variables

Runtime environment variables are further narrowed down only to the runtime or the process that they are associated with. These are usually set by the parent process that creates the new process and is accompanied by the system and user environment variables as well.

You can use terminal scripting to create and store these variables on the fly. Runtime variables are usually not permanent unless scripted, and you need to define them whenever you start a new process.

To view environment variables in the terminal:

1) Use the printenv command to display all environment variables

2) To search for specific variables, pipe the output into less for easier navigation

3) Alternatively, use grep to check specific environment variables

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

PS1 variable is a type of environment variable that defines the generation and customization of the content of the primary prompt that appears while interacting with the Bash shell. Actually, PS1 stands for Prompt String 1. It specifies how the prompt appears before you type a new command in your terminal. Moreover, you can modify this PS1 variable and customize the prompt’s appearance.

To view the current value of $PS1 in your terminal, simply type command echo $PS1.

4. \*Як можна змінити значення змінної $PS1? Що при цьому відбудеться в рядку запрошенні в bash (рядок запрошення перед початком кожної команди). Як змінити значення цієї змінної не на поточний сеанс, а за замовчуванням?

To customize the bash prompt (PS1) in Linux, you can use the following steps:

* Open the `~/.bashrc` file in a text editor. This file is located in your home directory and contains configuration settings for the bash shell.
* Find the line that sets the value of `PS1`.

When you change the value of the $PS1 variable in your Bash shell, it affects the appearance of the prompt line displayed before each command.

To change the value of $PS1 by default (for all future sessions), you need to modify your shell configuration file (usually ~/.bashrc or ~/.bash\_profile). Then open the appropriate file in a text editor (e.g., nano ~/.bashrc). Next, you need to add or modify the line that sets $PS1 according to your desired format. And finally save the file and either restart your terminal or run source ~/.bashrc to apply the changes.

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

Quotation marks are used throughout Linux administration and most computer programming languages to let the system know that the information contained within the quotation marks should either be ignored or treated in a way that is very different than it would normally be treated. There are three types of quotes that have special significance to the Bash shell: double quotes ", single quotes ', and back quotes `. Each set of quotes alerts the shell not to treat the text within the quotes in the normal way.

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

Program Control Instructions are the machine code that are used by machine or in assembly language by user to command the processor act accordingly. These instructions are of various types. These are used in assembly language by user also. But in level language, user code is translated into machine code and thus instructions are passed to instruct the processor do the task.

There are the following types: Compare Instruction, Unconditional Branch Instruction, Conditional Branch Instruction, Subroutines, Halting Instructions and Interrupt Instructions.

7. \*\*В чому різниця якщо в кінці рядку запрошення bash стоїть символ $ чи #? Наприклад на екрані ми бачимо наступні записи



$ indicates that you are a normal user. As a regular user, you have limited privileges and can perform regular tasks within your user account. # signifies that you are the system administrator (root). Root users have elevated privileges and can perform administrative tasks, such as installing software, modifying system files, and managing user accounts.

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

The basic difference is that locate locates all the related file names in the entire filesystem, whereas whereis command only give the location (system/local address of file) of installed application.

**Висновки:** we got acquainted with the basic commands of the CLI mode in Linux and with the basic text commands in the terminal mode in different operating systems.