“Київський фаховий коледж зв’язку”

Циклова комісія Комп’ютерної та програмної інженерії

**ЗВІТ ПО ВИКОНАННЮ**

**ЛАБОРАТОРНОЇ РОБОТИ №**2

з дисципліни: «Операційні системи»

**Тема: «“Знайомство з інтерфейсом та можливостями ОС Linux”»**

Виконали

груп**и КСМ 13А**

**Команда : ВВС**

**Панчук О. Петрик С.**

Перевірив викладач

Сушанова В.С.

Київ 2023

**Мета роботи:**

1. Отримання практичних навиків роботи з середовищами віртуальних машин та операційними системами різних типів та сімейств – їх графічною оболонкою, входом і виходом з системи, ознайомлення зі структурою робочого столу, вивчення основних дій та налаштувань при роботі в системі.

**Матеріальне забезпечення занять**

1. ЕОМ типу IBM PC.

2. ОС сімейства Windows (Windows 7).

3. Віртуальна машина – Virtual Box (Oracle).

4. Операційна система GNU/Linux – CentOS.

**Завдання для попередньої підготовки**

***Готував матеріал студент Панчук О.***

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 |
| **Operating System** | Операційна система |
| **GUI terminal** | термінал з графічним інтерфейсом користувача |
| Server Applications | Серверні додатки |
| Desktop Applications | Додатки для робочого столу |
| Line interface | інтерфейс командного рядка |
| Linux community | спільнота користувачів Linux |
| Network service | мережева служба |
| Linux administrator. | адміністратор Linux |
| BIOS | Базова система введення-виведення |
| virtualbox | virtualbox |

Вивчіть матеріали онлайн-курсу академії Cisco “NDG Linux Essentials”:

- Chapter 3 - Working in Linux

- Chapter 4 - Open Source Software and Licensing

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

- Chapter 03 Exam

- Chapter 04 Exam

4. Дайте визначення наступним поняттям:

- CLI-режим-or Command Line Interface mode, is a mode of operation for a computer program or operating system in which the user interacts with the program or system by entering text commands from the keyboard at a command prompt, rather than using a graphical user interface. In this mode, the user can execute commands to control the system, launch programs, manage files, and more, using text-based input and receiving text-based output from the system.

- Термінал на основі графічного інтерфейсу користувача - is a type of interface that allows users to interact with electronic devices through graphical images and visual cues, as opposed to text-based interfaces that rely on text, command-line input, and textual navigation.

-A virtual terminal is a software environment within an operating system that allows a user to interact with the system through a text-based interface or command-line interface, similar to a traditional terminal. This terminal typically operates in a window on the graphical desktop and can be used for entering text commands and interacting with the operating system, even in a graphical user interface (GUI) environment where a graphical interface is also available. Virtual terminals enable users to execute text-based commands and manage the system directly through the command line.

**Хід роботи**

***Готував матеріал студент Панчук О.***

Робота в графічному режимі в ОС сімейства Linux (робота з інтернет-джерелами):

* 1. Оберав графічну оболонку Gnome для ОС сімейства Linux, яку розглянув. Структуру Gnome робочого простору користувача, та описав основні його компоненти
* Applications Tab:

The Applications tab contains all the programs that are installed in the system and available for the user to use. It is divided into categories, making it easier to find the required program. Users can search for programs using the search function or browse them in alphabetical order.

* Places Tab:

The Places tab provides quick access to various sections of the file system and other storage locations. Here, you can find links to the user's Home folder, Public files, Desktop, Documents, Downloads, and other important sections.

* System Menu:

The System menu contains tools and system settings that allow you to configure the operating system. This includes system settings, user and group management, system update programs, network configuration, and much more.

* Activities Overview:

Activities Overview is the central workspace in GNOME, where users can multitask effectively. It includes the desktop, open program windows, and allows for quick launching of applications and switching between them. The Activities Overview also has a panel on the left side, providing quick access to all installed programs and nested sections, including Applications and Places.

The GNOME graphical user interface is known for its simplicity and user-friendliness, and these workspace components help make interaction with the system more convenient and efficient.

* 1. Запуск програм у графічній оболонці Gnome in Linux . Дослідіть можливості запуску додатків різними способами
* Using Applications Overview:

Press the "Super" key (usually the key with the OS logo) or click on "Activities" in the top-left corner of the screen.

You will see the Activities Overview, where you can search for and launch programs. Simply enter the name of the program in the search field or scroll through the pages to view all installed programs.

* Launching via "Applications":

Click on "Applications" on the application panel in the top-left corner of the screen. You will see a list of program categories.

Click on the category you're interested in and select the program you want to launch.

* Using the Dock:

GNOME also supports docks or panels on the desktop where you can pin shortcuts for frequently used programs. Simply click on the shortcut to launch the program.

* Launching from the Command Line:

You can launch a program from the command line by entering the program's name and pressing "Enter." For example, to launch Firefox, type "firefox" and press "Enter."

* Using Keyboard Shortcuts:

GNOME includes hotkeys that allow you to quickly launch programs. For example, Alt + F2 opens the Run dialog, where you can enter the program's name to launch it.

* Using the Search Function:

Press the "Super" key and start typing the program's name. GNOME will provide you with search results, and you can launch the program directly from the results.

1.3. Вихід з системи та завершення роботи в Linux. Як виконати в графічному інтерфейсі Gnome наступні дії

* Switching to the root user:

To switch to the root user in the GNOME graphical interface:

Click on the icon representing your user account in the top right corner of the screen.From the menu, select "Switch User."Enter the username "root" and the password for the root account. You are now logged into the system as root.

* Rebooting the system:

To reboot the system in the GNOME graphical interface:

Click on the icon representing your user account in the top right corner of the screen.From the menu, choose "Power Off" or "Log Out."

A dialog window will appear, allowing you to select "Restart." Click on this option, and the system will restart.

* Shutting down the system:

To shut down the system in the GNOME graphical interface:

Click on the icon representing your user account in the top right corner of the screen.

From the menu, select "Power Off."

A dialog window will appear, allowing you to choose "Power Off." Click on this option, and the system will shut down.

1. I use an iPhone smartphone running the iOS 17.1 operating system.

Виконав Петрик С.С.

2.1 Its main menu features a graphical interface that includes:

***Dock***: A row of frequently used app icons at the bottom of the screen.

***Home Screen***: The primary screen with app icons, folders, and widgets.

***Notification Center***: A place for notifications and widgets, accessed by swiping down from the top.

***Control Center***: Quick settings and actions, accessed by swiping down from the top-right corner.

***Search***: A search function for finding apps, contacts, and more.

***App Library***: Automatic organization of apps into categories, introduced in iOS 14.

iOS uses a graphical user interface (GUI) called "UIKit." UIKit is a graphical user interface (GUI) framework used in iOS, providing a set of tools and components for designing and building user interfaces in Apple's mobile operating system. It includes standard design elements, such as buttons, labels, text fields, and more, along with predefined styles and guidelines to create consistent and user-friendly app interfaces. UIKit is fundamental to the visual design and interaction of iOS apps.

2.2 The menu for configuring mobile phone components.

***Wi-Fi***: Allows you to connect to Wi-Fi networks and manage network preferences.

***Bluetooth***: Enables or disables Bluetooth connectivity and manages paired devices.

***Cellular***: Provides information about your cellular data usage, carrier settings, and options to enable or disable cellular data.

***Personal Hotspot***: Allows you to share your iPhone's internet connection with other devices.

***Data Usage***: Provides details on data usage for different apps and the option to reset data statistics.

***Airplane Mode***: Quickly turns off all wireless connections, such as Wi-Fi, cellular, and Bluetooth.

***Wi-Fi Calling***: Enables the use of Wi-Fi for making calls when a cellular signal is weak.

***Notifications***: Customizes notification settings for apps, including notification style, sounds, and alerts.

***Sounds & Haptics***: Adjusts ringtone, vibration, and sound settings for various notifications and system sounds.

***Do Not Disturb***: Configures settings to silence calls and notifications during specific time periods or while the device is in use.

***Display & Brightnes***s: Allows you to adjust screen brightness, text size, and display settings.

***Wallpaper***: Sets the wallpaper for the lock screen and home screen.

***Privacy***: Manages app permissions and controls access to location, contacts, photos, and more.

***Battery***: Provides information on battery usage by apps and enables low power mode.

***Screen Time***: Monitors and manages device usage, including app limits and content restrictions.

***Face ID & Passcode*** (or Touch ID & Passcode): Manages security settings, including biometric authentication (Face ID or Touch ID) and passcode changes.

***Emergency SOS***: Configures emergency contact information and SOS settings.

***Touch ID/Face ID & Passcode***: Manages biometric authentication and passcode settings.

***Accessibility***: Customizes settings for accessibility features, such as VoiceOver, Magnifier, and more.

***General***: Provides various system-wide settings, including software updates, reset options, and device information.

***VPN & Device Management***: Manages Virtual Private Network (VPN) configurations and device management profiles.

***Reset***: Options to reset various aspects of the device, such as network settings, keyboard dictionary, and content.

2.3 Using keyboard shortcuts to perform special actions.

* You can simultaneously press the volume up and power button to take a screenshot.
* Press and hold the volume down and power button to power off the smartphone.
* Additionally, on the iPhone 15, a new additional button has been introduced, which allows you to set your custom shortcuts.

2.4 System Login and Device Shutdown, Battery Power Settings

System Login and Device Shutdown:

* To log into the system on the iPhone, you can use various authentication methods such as Face ID or entering your passcode.
* To shut down the device, press and hold the side button and either volume button simultaneously until the "slide to power off" slider appears on the screen. Slide it to power off the device.

Battery Power Settings:

* iPhone provides several battery-related settings to help manage battery life efficiently.
* You can access these settings by going to "Settings" > "Battery."
* In the Battery settings, you can view your battery's current health, which indicates how well it holds a charge.
* You can enable "Low Power Mode" to conserve battery life when your battery is running low. This mode reduces background activity and certain visual effects.
* Additionally, you can see a breakdown of battery usage by apps and services, allowing you to identify which apps are consuming the most power.
* The "Optimized Battery Charging" feature is available to slow down battery aging by learning your daily charging patterns and avoiding overcharging.

**Відповіді на контрольні запитання**

***Готував матеріал студент Петрик С.С.***

Control Questions:

1.

Ubuntu Server

Ubuntu Server by Canonical is a versatile distribution for both PCs and servers, often leading in rankings. It is a popular choice for various applications, including cloud platforms.

Debian Server

Debian, one of the earliest Linux distributions, forms the foundation for many others, such as Ubuntu and RHEL. Debian's strength lies in its stability, crucial for servers, as it mitigates distribution shortcomings. It offers stable, testing, and unstable versions, with stable having a longer support cycle.

Red Hat Enterprise Linux (RHEL)

Red Hat Enterprise Linux is a leading commercial distribution known for its long-term support. It's well-suited for cloud servers and data centers.

Fedora Server

Fedora, developed by the community and based on RHEL, offers the latest software with frequent releases. It's suitable if you prefer the yum environment and value newer software versions. It's versatile for both bare-metal and cloud deployments.

OpenSUSE Leap

OpenSUSE Leap is a stable distro with annual releases, using older software versions for stability. It's adaptable and flexible, available not only for servers but also desktops and laptops. There is no separate ISO image for servers; you can install the server via the OpenSUSE installer.

SUSE Linux Enterprise Server (SLES)

SUSE Linux Enterprise Server is a commercial Linux server distribution popular among enterprises. New SLES releases occur every 3-4 years. It offers various tools and commercial support and is easily transitioned from OpenSUSE Leap.

2.

Bourne Shell (sh):

* Simple and lightweight.
* Limited interactive features.
* Basic scripting capabilities.
* Standard on most Unix-like systems.

C Shell (csh):

* Features C-like syntax.
* Good interactive features.
* Lacks advanced scripting capabilities.
* History and alias support.

Bourne Again Shell (Bash):

* Most popular shell.
* Combines features of sh and csh.
* Excellent scripting capabilities.
* Rich interactive features.
* Extensive community support.

tcsh:

* An improved version of csh.
* Enhancements in interactive features.
* Command-line editing and history improvements.
* Limited scripting capabilities.

Korn Shell (Ksh):

* Combines features of sh and csh.
* Powerful scripting capabilities.
* Advanced command-line editing and history.
* Standard on some Unix systems.

zsh:

* Highly customizable and extensible.
* Robust scripting and interactive features.
* Advanced tab completion and prompt customization.
* Widely used among power users.

In summary, the choice of shell depends on your specific requirements and preferences. Bash is a versatile and widely supported choice for both interactive use and scripting. Ksh and zsh offer advanced features, making them suitable for power users and developers. Tcsh and csh, while less popular today, are still used in some legacy environments. The Bourne shell (sh) serves as a simple and portable option.

3.

A package manager is a crucial tool in Linux that is used for installing, updating, configuring, and removing software packages on a system. It helps maintain software dependencies, tracks installed packages, and ensures software installations are consistent and efficient.

Some commonly known package managers in Linux include:

APT (Advanced Package Tool):

* Used in Debian and Ubuntu-based distributions.
* Provides commands like apt-get and apt to manage packages.

YUM (Yellowdog Updater, Modified):

* Commonly found in Red Hat, CentOS, and Fedora-based distributions.
* Utilizes commands like yum to manage packages.

DNF (Dandified YUM):

* A successor to YUM, used in modern Fedora and RHEL distributions.
* Replaces the older YUM package manager.

Pacman:

* Used in Arch Linux and Arch-based distributions like Manjaro.
* Provides commands like pacman to manage packages.

4.

Several security measures are commonly used in Linux systems to enhance security.

User Accounts and Permissions: Linux uses user accounts and permissions to control access to system resources.

Firewalls: Tools like iptables and firewalld manage network traffic and protect against unauthorized access.

SELinux (Security-Enhanced Linux): Enforces mandatory access controls and defines security policies.

AppArmor: Confines application programs to reduce potential damage from security breaches.

Filesystem Encryption: LUKS and eCryptfs enable filesystem-level encryption to safeguard data.

Package Signing: Digital signatures verify software package authenticity and integrity.

Intrusion Detection Systems (IDS): Tools like Snort and Suricata monitor network traffic for suspicious activity.

Regular Updates and Patch Management: Keeping the system and software up to date with security patches is crucial.

Auditing and Logging: Linux generates logs for security analysis and maintains a record of activities.

Authentication Mechanisms: Various methods like passwords and SSH keys ensure secure user access.

Intrusion Prevention Systems (IPS): IPS tools actively respond to suspicious network activity.

5.

The use of virtualization has become highly relevant for several reasons:

Resource Efficiency: Virtualization allows better utilization of hardware resources, reducing the need for physical servers and saving costs on hardware and power.

Isolation: Virtualization provides strong isolation between virtual machines (VMs), enhancing security by preventing cross-contamination between applications and workloads.

Flexibility and Scalability: VMs can be easily provisioned and scaled, making it simpler to adapt to changing workloads and demands.

Development and Testing: Virtual environments are ideal for development and testing, enabling the creation of isolated testbeds for software and applications.

Cloud Computing: Virtualization is fundamental to cloud computing, allowing cloud providers to efficiently manage and allocate resources to users.

Legacy System Support: It enables the hosting of legacy applications and operating systems, extending the lifespan of older software.

Server Consolidation: Organizations can consolidate multiple servers onto a single physical host, reducing infrastructure complexity.

6.

Containerization is a lightweight form of virtualization that encapsulates and isolates applications and their dependencies, allowing them to run consistently across different computing environments. Containers provide a consistent and efficient way to package, distribute, and deploy software, making it easier to develop, test, and run applications in various environments.

7.

Advantages and Disadvantages of Using Open Source Software (OSS):

Advantages:

* Free Availability: OSS is freely available for use, reducing licensing costs.
* Convenient Modification: The code is open and can be adapted to specific user needs.
* Community Support: An active developer and user community ensures quick issue resolution and readily available assistance.
* Security: Open source code allows inspection for potential vulnerabilities and ensures security.
* Compatibility and Integration: Many OSS programs support standards, making integration into various environments easier.

Disadvantages:

* Support: Some OSS projects may have limited support compared to commercial software.
* Insufficient Documentation: Some projects may be less well-documented, making them harder to use.
* Limited Functionality: In some cases, OSS may lack certain features offered by commercial software.
* Support Costs: If support is needed, the cost of providing it can be significant, especially in business environments.

8.

In Linux, by default, there can be 6 active virtual consoles (terminals) numbered from 1 to 6. They are accessible through key combinations *Ctrl+Alt+F1* to *Ctrl+Alt+F6*.

To switch between virtual consoles, use these key combinations. For example, to switch to the first virtual console, use *Ctrl+Alt+F1*, for the second one, use Ctrl+Alt+F2, and so on.

Example:

* You are in a graphical interface (e.g., Xfce).
* To switch to the first virtual console, press *Ctrl+Alt+F1*.
* To return to the graphical interface, switch back by pressing, for instance, *Ctrl+Alt+F7* or *Ctrl+Alt+F8*, depending on your distribution.

9.

The virtual console typically numbered as F7 (Ctrl+Alt+F7) serves as the graphical user interface (GUI) in most Linux systems. This is where you see the graphical desktop and user interface, such as GNOME, KDE, Xfce, or another, depending on your chosen desktop environment.

10.

In a typical Linux system, it is not possible to register multiple times with the same system username simultaneously. Each user has a unique User ID (UID) that is used to determine access rights and identify the user.

However, a scenario where the same user can open multiple sessions in different virtual consoles or terminal windows is possible, but each session will have its separate session. This is useful for managing different tasks or switching between them without logging out of the user's account.

This can also be useful in cases of remote access to the system via SSH, where one user may have multiple concurrent sessions under the same system username.

However, it is important to use this feature carefully, as having a large number of concurrent sessions can potentially overload the system

**Висновки**

During the course of this work, we had the opportunity to familiarize ourselves with the interfaces of Linux operating systems. We conducted practical exercises within Linux operating system environments, as well as in a mobile operating system. We explored their graphical user interfaces, the fundamental procedures for logging in and out of the system, and closely examined the structure of the desktop environment. Additionally, we became acquainted with essential actions and settings necessary for working within these systems.

This work has facilitated our understanding of fundamental concepts in Linux operating systems and has equipped us with practical skills for using their interfaces. We have acquired proficiency in logging in, logging out, launching applications, and performing basic operations within the graphical user interfaces of these systems, which are crucial for further work within Linux operating systems.