“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»

Виконав(ла/ли) студент(ка/и)

групи РПЗ-33

Шульга Олексій

Перевірила викладач

Сушанова В.С.

Київ 2025

**Мета роботи:**

1. Знайомство з інтерфейсами ОС Linux.

2. Отримання практичних навиків роботи в середовищах ОС Linux та мобільної ОС – 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 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**

#### 1. Словник термінів

* CLI (Command Line Interface): Інтерфейс командного рядка — текстовий спосіб взаємодії з комп'ютером, де команди вводяться у вигляді тексту.
* Kernel: Ядро — центральна частина операційної системи, що керує апаратними ресурсами.
* GUI (Graphical User Interface): Графічний інтерфейс користувача — спосіб взаємодії з ОС за допомогою графічних елементів (іконок, вікон, кнопок).
* Virtual Terminal: Віртуальний термінал — текстова консоль, що працює паралельно з графічним інтерфейсом, доступ до якої здійснюється комбінаціями клавіш (наприклад, Ctrl + Alt + F1 ... F6).
* Terminal Emulator: Емулятор терміналу — програма в графічному середовищі, що імітує роботу віртуального терміналу, надаючи доступ до командного рядка.
* Shell: Оболонка — командний інтерпретатор, який надає користувачу інтерфейс для доступу до сервісів ядра ОС.
* Package Manager: Менеджер пакетів — утиліта, яка автоматизує процеси встановлення, оновлення та видалення програмного забезпечення в операційній системі.
* Distribution (Distro): Дистрибутив — повний пакет програмного забезпечення, що включає ядро Linux, системні утиліти, бібліотеки та додаткові програми, об'єднані в єдину ОС.
* Open Source Software: Програмне забезпечення з відкритим вихідним кодом — ПЗ, вихідний код якого доступний для перегляду, модифікації та розповсюдження.

#### 2. Визначення понять

* CLI-режим: Це режим роботи операційної системи, де користувач взаємодіє з нею, вводячи команди у вигляді тексту через термінал або консоль. На відміну від графічного інтерфейсу, тут немає вікон, іконок чи миші, що робить його більш ефективним для автоматизації завдань і роботи на серверах.
* Термінал на основі графічного інтерфейсу користувача (GUI-based terminal): Це програма, що запускається у віконному середовищі GUI і надає доступ до командного рядка. Вона дозволяє виконувати команди в текстовому режимі, не виходячи з графічної оболонки. Прикладом є програми Terminal в Gnome, Konsole в KDE.
* Віртуальний термінал: Це повноцінна текстова консоль, яка працює поза графічною оболонкою, як правило, на окремій віртуальній консолі. У Linux їх може бути кілька, і користувач може перемикатися між ними за допомогою комбінацій клавіш Ctrl + Alt + Fn (де n — номер консолі). Наприклад, Ctrl + Alt + F1 — для першого віртуального терміналу.

**Хід роботи**

#### 1. Робота в графічному режимі ОС Linux (Приклад Gnome)

Структура робочого простору користувача (Gnome):

* Верхня панель: Розташована у верхній частині екрана. Вона містить:

"Діяльності" (Activities): Кнопка або гарячий кут у лівому верхньому куті, що відкриває огляд всіх запущених вікон, віртуальних робочих столів та панель пошуку.

Панель меню (Menu bar): Зазвичай містить іконки програм, що працюють, та системні індикатори (мережа, звук, батарея, дата).

Центральна частина: Тут відображається дата і час.

Системне меню: У правому верхньому куті, що дозволяє вийти з системи, перезавантажитися, вимкнути комп'ютер, заблокувати екран або змінити налаштування.

* Панелі швидкого доступу (Dock): Розташована зліва на екрані, де знаходяться іконки найчастіше використовуваних програм та запущених вікон.
* Пошук: Доступний через кнопку "Діяльності". Просто починайте вводити назву програми або файла, і система знайде відповідні результати.
* Доступ до робочих столів: У режимі "Діяльності" (Activities) праворуч відображаються мініатюри віртуальних робочих столів. Можна перетягувати вікна між ними або створювати нові.

#### 2. Запуск програм

* Через панель швидкого запуску (Dock): Достатньо просто натиснути на іконку програми, що знаходиться на бічній панелі.
* Через пошук в меню / глобальне меню: Натисніть кнопку "Діяльності" (Activities) або гарячу клавішу Super (Windows), потім почніть вводити назву програми (наприклад, "Firefox"). Після того, як програма з'явиться в результатах пошуку, натисніть Enter або клікніть по іконці.
* Через віджет запуску: Це може бути та ж сама панель Dock або спеціалізований віджет (наприклад, Kupfer або Synapse для інших DE), де можна швидко запускати програми, шукати файли та виконувати інші дії.

#### 3. Вихід з системи та завершення роботи

* Зміна користувача на root: У Linux з міркувань безпеки зазвичай не використовують прямого входу під root. Замість цього використовують команду sudo в терміналі для виконання адміністративних завдань. Щоб переключитися на іншого користувача, можна використовувати команду su.
* Перезавантаження системи: Натисніть на системне меню у верхньому правому куті, оберіть "Вимкнути/Вийти", а потім у спливаючому вікні оберіть "Перезавантажити" (Restart).
* Вимкнення системи: Аналогічно, через системне меню, оберіть "Вимкнути/Вийти" (Power Off / Log Out) та натисніть "Вимкнути" (Power Off).

Робота в середовищі мобільної ОС (Приклад IOS)

### 1. Користування iOS

Як користувач ви працюєте з графічним інтерфейсом iPhone або iPad:

* Панель керування (Control Center) – швидкий доступ до Wi-Fi, Bluetooth, яскравості, режиму «Не турбувати».
* Налаштування (Settings) – зміна параметрів системи, сповіщень, конфіденційності.
* App Store – установка та оновлення програм.
* Файли та iCloud – робота з хмарними та локальними файлами.
* Жести та багатозадачність – перемикання між програмами, використання поділу екрану (на iPad).

### 2. Розробка під iOS

Для створення власних додатків потрібні:

macOS + Xcode – офіційне середовище розробки від Apple.

Мови програмування:  
Swift – сучасна мова для iOS/macOS.  
Objective-C – більш стара, але досі підтримується.

Інструменти:  
Interface Builder – візуальне створення інтерфейсу.  
Simulator – тестування застосунків без фізичного пристрою.  
App Store Connect – публікація та аналітика застосунків.  
  
3. Безпека та особливості iOS  
Система працює в закритому середовищі: усі додатки в «пісочниці», що обмежує доступ до системних файлів.  
Регулярні оновлення iOS підтримують високий рівень захисту.  
  
4. Практичні поради для роботи

* Для звичайного користування – регулярно робіть резервні копії в iCloud.
* Для розробки – встановіть останню версію Xcode і створіть безкоштовний або платний Apple Developer Account.

**Відповіді на контрольні запитання**

1. Приклади серверних додатків Linux:
   1. Бази даних: MySQL, PostgreSQL, MongoDB.
   2. Сервери розсилки повідомлень: Exim, Postfix, Sendmail.
   3. Файлообмінники: Samba (для Windows-мереж), NFS (для Linux/Unix), ProFTPD.
2. Порівняння оболонок:
   1. Bourne shell (sh): Одна з найстаріших оболонок. Проста, мінімалістична, використовується як стандартна оболонка на багатьох Unix-подібних системах.
   2. Bourne Again shell (Bash): Найпопулярніша оболонка в Linux. Є розширенням sh, додає автодоповнення, історію команд, псевдоніми та інші зручні функції.
   3. C shell (csh) та tcsh: Орієнтовані на програмістів, які звикли до синтаксису мови С. tcsh — покращена версія csh.
   4. Korn shell (ksh): Поєднує найкращі риси sh та csh, вважається потужною та швидкою оболонкою.
   5. Z shell (zsh): Сучасна, розширювана оболонка. Заснована на ksh, але має безліч покращень, як-от потужніша автодоповнення, корекція команд та підтримка плагінів.
3. Менеджер пакетів: Це утиліта, яка спрощує процес керування програмним забезпеченням. Замість того, щоб завантажувати, компілювати та встановлювати кожну програму вручну, менеджер пакетів автоматизує цей процес.
   1. Приклади: APT (Debian, Ubuntu), YUM / DNF (Fedora, CentOS), Pacman (Arch Linux).
4. Засоби безпеки в Linux:
   1. Права доступу: Користувачі та групи мають чітко визначені права на читання, запис та виконання файлів.
   2. sudo: Механізм, що дозволяє користувачам виконувати адміністративні команди без прямого входу як root.
   3. Брандмауер (Firewall): Наприклад, iptables або ufw, що контролюють мережевий трафік.
   4. SELinux / AppArmor: Системи примусового контролю доступу (MAC), які обмежують дії програм, навіть якщо вони запущені від імені користувача з високими привілеями.
   5. Журнали (Logs): Система веде докладні записи подій, що допомагає виявляти несанкціоновану активність.
5. Чому віртуалізація актуальна:
   1. Ізоляція: Кожна віртуальна машина є ізольованою, що підвищує безпеку.
   2. Ефективне використання ресурсів: На одному фізичному сервері можна запустити багато ВМ, що економить апаратні ресурси та електроенергію.
   3. Гнучкість: Легко переносити, масштабувати та клонувати віртуальні машини.
   4. Тестування: Ідеальне середовище для тестування нового ПЗ або налаштувань, оскільки зміни не впливають на основну систему.
6. Контейнеризація: Це спосіб віртуалізації на рівні операційної системи, де додатки запускаються в ізольованих "контейнерах". На відміну від ВМ, контейнери використовують ядро хостової ОС, що робить їх значно легшими, швидшими та менш ресурсозатратними. Найпопулярніший інструмент для контейнеризації — Docker.
7. Переваги/недоліки відкритого ПЗ:
   1. Переваги:
      1. Безкоштовність: Часто доступне безкоштовно.
      2. Гнучкість: Можна змінювати код під власні потреби.
      3. Безпека: Відкритий код дозволяє спільноті швидко знаходити та виправляти вразливості.
   2. Недоліки:
      1. Відсутність офіційної підтримки: Часто немає централізованої технічної підтримки.
      2. Складність: Може вимагати додаткових навичок для налаштування та використання.
      3. Залежність від спільноти: Життєздатність проєкту залежить від активності спільноти.
8. Кількість віртуальних терміналів:
   1. За замовчуванням у Linux може бути від 6 до 12 віртуальних консолей.
   2. Для доступу до них використовуються комбінації клавіш: Ctrl + Alt + F1 до Ctrl + Alt + F6.
   3. Щоб повернутися до графічного інтерфейсу, використовується Ctrl + Alt + F7 (в деяких дистрибутивах F1).
9. Віртуальна консоль для графічної оболонки:
   1. Графічна оболонка (X Window System) зазвичай запускається на віртуальній консолі F7. Тобто, щоб перейти до неї з CLI, використовують комбінацію Ctrl + Alt + F7.
10. Реєстрація в системі декілька разів під одним ім'ям:
    1. Так, це можливо. Користувач може увійти в систему кілька разів, наприклад, через різні термінали або з різних віртуальних консолей.
    2. Переваги:
       1. Багатозадачність: Дозволяє одночасно виконувати різні завдання під одним обліковим записом, не перемикаючись між сесіями.
       2. Віддалений доступ: Користувач може увійти до системи через локальний термінал і одночасно через SSH з віддаленого комп'ютера.

**Висновки:**

У ході виконання лабораторної роботи я ознайомився з основними поняттями операційної системи Linux: командним рядком (CLI), графічним інтерфейсом (GUI) та ядром. Було розглянуто структуру робочого простору в графічній оболонці, способи запуску програм і керування сеансом користувача. Також проаналізовано ключові особливості мобільних операційних систем, їхній графічний інтерфейс та можливості налаштування. Окрему увагу приділено концепціям віртуалізації, контейнеризації та безпеки, що дало змогу сформувати цілісне уявлення про сучасні методи адміністрування систем. Отже, проведене дослідження підтверджує, що Linux — це потужна, гнучка й безпечна операційна система, придатна як для настільного використання, так і для серверних рішень.

**Conclusions:**   
  
During the laboratory work, I became familiar with the basic concepts of the Linux operating system: command line interface (CLI), graphical user interface (GUI), and the kernel. The structure of the workspace in the graphical shell was examined, as well as ways to launch programs and manage user sessions. Key features of mobile operating systems, their graphical interfaces, and customization options were also analyzed. Special attention was paid to the concepts of virtualization, containerization, and security, which allowed for the formation of a comprehensive understanding of modern system administration methods. Therefore, the research conducted confirms that Linux is a powerful, flexible, and secure operating system suitable for both desktop use and server solutions.