“Київський фаховий коледж зв’язку”

Циклова комісія комп’ютерної та програмної інженерії

**ЗВІТ ПО ВИКОНАННЮ**

**ЛАБОРАТОРНОЇ РОБОТИ №3**

з дисципліни: «Операційні системи»

**Тема: “Команди Linux для управління процесами”**

Виконав(ла/ли) студент(ка/и)

групи КСМ-33:

Литвин Артем

Перевірила викладач

Сушанова В.С.

Київ 2025

**Мета роботи:**

1. Отримання практичних навиків роботи з командною оболонкою Bash.
2. Знайомство з базовими командами для управління процесами.

**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

**1. Словник базових англійських термінів**

| Термін (англ.) | Переклад | Призначення та функціональність |
| --- | --- | --- |
| **Process** | Процес | Програма, що виконується системою. |
| **PID** | Ідентифікатор процесу | Унікальний номер, присвоєний кожному процесу (Process ID). |
| **PPID** | Ідентифікатор батьківського процесу | PID процесу, який запустив поточний процес (Parent PID). |
| **TTY** | Термінал | Термінальний пристрій, з якого був запущений процес. |
| **Signal** | Сигнал | Попередньо визначене повідомлення, що надсилається процесам для керування їхньою поведінкою (наприклад, зупинка або завершення). |
| **Top** | Top | Утиліта для моніторингу процесів у реальному часі та загального стану системи. |
| **ps** | ps | Утиліта для створення знімка стану процесів на певний момент часу. |
| **Kill** | Kill | Команда для надсилання сигналів процесам за їхнім PID. |
| **Killall** | Killall | Команда для надсилання сигналів процесам за їхнім ім'ям. |
| **Zombie process** | Процес-зомбі | Процес, що завершився, але його батьківський процес ще не забрав його статус виходу. |

**Хід роботи:**

* 1. Початкова робота в CLI-режимі в Linux ОС сімейства Linux:
  2. Запустіть операційну систему Linux Ubuntu. Виконайте вхід в систему та запустіть термінал ***(якщо виконуєте ЛР у 401 ауд.)***.
  3. Запустіть віртуальну машину Ubuntu\_PC ***(якщо виконуєте завдання ЛР через академію netacad)***
  4. Запустіть свою операційну систему сімейства Linux ***(якщо працюєте на власному ПК та її встановили)*** та запустіть термінал.
  5. **Дайте відповіді на наступні питання:**

### **1. Як вивести вміст директорії /proc? Де вона знаходиться та для чого призначена? Охарактеризуйте інформацію про її вміст?**

Вміст директорії /proc виводиться командою ls /proc. Вона знаходиться у кореневому каталозі (/). Директорія /proc є віртуальною файловою системою (псевдофайловою системою), що існує лише в оперативній пам'яті. Вона призначена для надання інтерфейсу для взаємодії з ядром (kernel) та процесами системи. Вміст /proc включає: каталоги з цифрами, що відповідають PID запущених процесів (наприклад, /proc/1234), та файли з інформацією про конфігурацію і статистику ядра (наприклад, cpuinfo, meminfo).

### **2. Як вивести інформацію про поточні сеанси користувачів. Якою командою це можна зробити?**

Інформацію про користувачів, які наразі увійшли в систему, можна вивести командою w або who. Команда w є більш детальною, оскільки вона показує не лише користувача і термінал, але й те, що він зараз виконує**.**

### **3. Які дії можна зробити в терміналі за допомогою комбінацій Ctrl + C, Ctrl + D та Ctrl + Z?**

* Ctrl + C: Надсилає сигнал INT (Interrupt) для негайного завершення процесу, що працює на передньому плані.
* Ctrl + D: Надсилає сигнал EOF (End-of-File), що зазвичай призводить до виходу з поточної оболонки (exit) або завершення введення даних.
* Ctrl + Z: Надсилає сигнал TSTP (Terminal Stop) для призупинення процесу та переведення його у фоновий стан (Stopped).

### **4. Чим відрізняється фоновий процес від звичайного. Де вони використовуються?**

Звичайний (Foreground) процес взаємодіє з терміналом (приймає ввід/виводить дані) і блокує термінал, доки не завершиться. Використовується для інтерактивних програм. Фоновий (Background) процес працює незалежно від терміналу, не приймає вводу і дозволяє терміналу залишатися доступним для інших команд. Використовується для тривалих завдань, які не потребують втручання користувача.

### **5. Опишіть наступні команди та поясніть що вони виконують – команда jobs, bg, fg.**

* jobs: Виводить список усіх призупинених або фонових процесів, запущених у поточній оболонці. Вказує їхній статус (Running або Stopped) та ідентифікатор завдання ([1], [2], тощо).
* bg (Background): Відновлює призупинений процес та переводить його на задній план (background), дозволяючи йому виконуватись, не блокуючи термінал. Використовується з ідентифікатором завдання, наприклад, bg %1.
* fg (Foreground): Відновлює фоновий або призупинений процес та переводить його на передній план (foreground), щоб користувач міг взаємодіяти з ним. Використовується з ідентифікатором завдання, наприклад, fg %1.

### **6. Якою командою можна переглянути інформацію про запущені в системи фонові процеси та задачі?**

Для перегляду інформації про фонові процеси, запущені в поточній оболонці, використовується команда jobs. Для перегляду всіх системних фонових процесів (включаючи ті, що запущені іншими користувачами або системою), використовується команда ps -ef або ps -aux.

### **7. Як призупинити фоновий процес, як його потім відновити та при необхідності перезапусти?**

* Призупинити фоновий процес: Якщо фоновий процес працює, ви можете надіслати йому сигнал STOP за його ідентифікатором завдання: kill -STOP %JobID (наприклад, kill -STOP %1).
* Відновити:
  + На передньому плані (foreground): fg %JobID.
  + На задньому плані (background): bg %JobID (або kill -CONT %JobID).
* Перезапустити: Для перезапуску необхідно спочатку завершити процес (kill %JobID або killall <ім'я>), а потім запустити його знову.

**Контрольні запитання:**

### **1. Яке призначення директорії /proc в системах Linux. Яку інформацію вона зберігає?**

**Директорія /proc є віртуальною файловою системою (псевдофайловою системою) у Linux. Її призначення — бути інтерфейсом для взаємодії з ядром (kernel) та процесами, що виконуються, відображаючи інформацію про систему у вигляді файлів та каталогів. Вона зберігає:**

* Інформацію про процеси: Кожен каталог, названий числом, відповідає PID (ідентифікатору процесу) і містить детальну інформацію про його поточний статус, використання пам'яті, дескриптори файлів тощо.
* Конфігурацію та статистику ядра: Файли, що відображають поточний стан системи, наприклад, cpuinfo (інформація про процесор) та meminfo (інформація про пам'ять).

### **2. Як серед будь-яких трьох процесів динамічно визначати, який з них в поточний момент часу використовує найбільший обсяг пам'яті? Який відсоток пам’яті він споживає від загального обсягу?**

**Щоб динамічно визначити, який процес використовує найбільший обсяг пам'яті, використовується команда top:**

1. Запустіть команду top у терміналі.
2. В інтерактивному режимі натисніть Shift + M (або просто M), щоб відсортувати список процесів за спаданням використання фізичної пам'яті (%MEM).
3. Процес, що використовує найбільше пам'яті, буде відображений першим у списку.
4. Відсоток споживаної пам'яті від загального обсягу буде вказаний у стовпці %MEM.

### **3. Як отримати ієрархію батьківських процесів в системах Linux? Наведіть її структуру та охарактеризуйте.**

**І**єрархію процесів можна отримати за допомогою команди ps -ef --forest або утиліти pstree.

* Структура: Процеси утворюють деревоподібну структуру, де кожен дочірній процес (PID) був запущений певним батьківським процесом (PPID).
* Характеристика: Корінь дерева завжди — процес init (PID 1) або systemd у сучасних дистрибутивах. Це головний процес, який керує всіма іншими. Якщо батьківський процес завершується, його дочірні процеси "усиновлюються" кореневим процесом.

### 

### 

### 

### 

### 

### 

### 

### **4. Чим відрізняється команда top від ps?**

Команда ps створює статичний знімок стану процесів у системі на момент запуску команди. Вона не оновлюється автоматично і використовується для звітів чи скриптів.

Команда top надає динамічний моніторинг процесів у реальному часі. Вона постійно оновлює інформацію (за замовчуванням кожні 3 секунди) і ідеально підходить для діагностики та пошуку "проблемних" процесів із високим навантаженням.

### **5. Які додаткові можливості реалізує htop в порівнянні з top?**

htop — це інтерактивна та візуально покращена версія top, яка реалізує такі додаткові можливості:

* Інтерфейс: Використовує кольори, підтримує горизонтальне/вертикальне прокручування та відображає графіки використання ресурсів для кожного ядра CPU окремо.
* Управління: Дозволяє легко надсилати сигнали kill (завершення) процесам, вибираючи їх курсором, без необхідності вводити PID.
* Ієрархія: За замовчуванням відображає процеси у вигляді дерева.

### 

### 

### 

### 

### 

### 

### 

### 

### 

### 

### **6. Опишіть компоненти вашої мобільної ОС для здійснення моніторингу запущених в системі процесів?**

Основними компонентами є:

1. Налаштування $\rightarrow$ Програми/Запущені сервіси: Стандартний графічний інтерфейс для перегляду активних фонових сервісів та їхнього споживання оперативної пам'яті.
2. Режим розробника (Developer Options): Містить більш детальні інструменти, як-от "Статистика процесів" або "Використання пам'яті", які надають детальний або агрегований звіт про використання RAM програмами.

### 

### 

### 

### 

### **7. Чи підтримує Ваша мобільна ОС термінальне керування роботою процесів, опишіть як саме.**

Так:

ADB Shell: За допомогою комп'ютера та Android Debug Bridge (ADB) можна отримати доступ до командного рядка пристрою. У цьому середовищі доступні команди ps (перегляд процесів) та kill PID (керування процесами).

1. Емулятори Терміналу: Сторонні програми, як-от Termux, створюють середовище Linux на пристрої, де доступні команди ps та kill для керування процесами, які працюють у цьому середовищі.

### **8. Чи можливо поставити сторонні програмні засоби, що дозволяють організувати управління та моніторинг роботою процесів у Вашому мобільному телефоні. Коротко опишіть їх.**

Так:

* Termux (Android): Емулятор терміналу, який створює повноцінне середовище Linux. Дозволяє встановлювати утиліти htop та top для детального моніторингу, а також використовувати команди kill та ps для управління процесами.
* Advanced Task Manager Apps: Графічні програми, які надають розширені функції моніторингу, очищення пам'яті та примусового закриття певних фонових програм, перевищуючи стандартний функціонал системи.

## 

## **Conclusion**

This laboratory work successfully helped me practically understand how to manage programs in Linux using the command line. I achieved the main goal: learning how to "see" and "control" all running programs, which Linux refers to as processes.