ОДЕСЬКИЙ НАЦІОНАЛЬНИЙ ПОЛІТЕХНІЧНИЙ УНІВЕРСИТЕТ

ІНСТИТУТ КОМП’ЮТЕРНИХ СИСТЕМ

КАФЕДРА «ІНФОРМАЦІЙНИХ СИСТЕМ»

Лабораторна робота № 7 з дисципліни «Операційні системи»

Тема: ***««Команди управління процесами в ОС Unix»***

**Виконав:**

Студент групи AI-202

Перець Сергій Дмитрович

Одеса 2020

**Мета роботи**: отримання навичок в управлінні процесами в ОС Unix засобами командної оболонки.

**Завдання до лабораторної роботи**

# Завдання 1 Моделювання багатозадачності

Нехай оперативна пам'ять на комп'ютері-сервері становить N Гб. Системні процеси ОС займають до M Гбайт пам'яті, а кожна програма користувача може використовувати до K Гбайт пам'яті. Нехай в середньому процеси програм користувачів витрачають F% свого часу на очікування завершення вводу/виводу. Визначте середню завантаженість процесора, використовуючи значення з таблиці 3 варіантів.

# Завдання 2 Перегляд таблиці процесів

1. Отримайте ієрархію всіх процесів із зазначенням імен користувачів, їх запустили.
2. Отримайте ієрархію процесів, запущених від імені вашого профілю і з зазначенням PID цих процесів.
3. Отримайте список процесів, запущених в поточному терміналі, зі спрощеним набором колонок виведення таблиці процесів.
4. Отримайте список процесів, запущених від імені вашого користувача, з розширеним набором колонок таблиці процесів.
5. Отримайте список процесів, запущених від імені вашого користувача із зазначенням наступного набору колонок: PID, TTY, PPID, STAT, NI, CMD 6. Отримайте список всіх сплячих процесів зі спрощеним набором колонок виведення таблиці процесів.

7. Отримайте список процесів, відсортованих по PID, і визначте:

1. загальна кількість запущених процесів;
2. кількість процесів, які виконуються;
3. кількість сплячих процесів.

8. Отримайте список процесів, відсортованих за % використання процесора.

# Завдання 3 Керування станами процесів

1. У поточному терміналі виконайте команду ping localhost, але не завершуйте її роботу.
2. Запустіть другий термінал доступу до Linux-сервера.
3. У другому терміналі для команди ping отримаєте таблицю її процесу (колонки PID, STAT, CMD).
4. У другому терміналі призупиніть виконання процесу команди ping
5. У першому терміналі отримайте список фонових процесів
6. У другому терміналі відновіть виконання припиненого процесу
7. У другому терміналі зупиніть виконання процесу команди ping
8. У першому терміналі запустіть команду ping в фоновому режимі так, щоб він не був автоматично зупинений навіть після закриття терміналу, з якого був запущений.
9. Закрийте перший термінал.
10. У другому терміналі для команди ping отримаєте таблицю її процесу (колонки PID, STAT, CMD). Зробіть висновок про стан процесу.
11. Завершіть роботу процесу.

# Завдання 4 Управління пріоритетами процесів

1. Створіть bash-програму, що виконує операцію циклічного складання за формулою: x = x + n, де початкове значення х = кількість букв вашого прізвища, n - кількість букв у вашому імені. Ім'я програми збігається з транслітерацією вашого прізвища з розширенням .sh, наприклад, ivanov.sh 2. Запустіть bash-програму у фоновому режимі.

1. Перегляньте таблицю процесів для запущеного процесу, пов'язаного з bash- програмою, з урахуванням набором колонок: PID, PPID, STAT, NI,% CPU, CMD.

1. Виконайте команду призупинення запущеного процесу.
2. Ще раз перегляньте таблицю процесів для призупиненого процесу з урахуванням набором колонок: PID, PPID, STAT, NI, %CPU, CMD. Зробіть висновки про його стан.
3. Виконайте команду продовження виконання припиненого процесу.
4. Ще раз перегляньте таблицю процесів для процесу, який продовжив виконуватися, з урахуванням набором колонок: PID, PPID, STAT, NI, %CPU, CMD. Зробіть висновки про його стан.
5. Створіть два файли як символічні посилання на створену bash-програму з іменами як ім'я поточного файлу з додаванням цифр 2 і 3, відповідно, наприклад:

ivanov2.sh, ivanov3.sh

1. Запустіть два файли у фоновому режимі.
2. Перегляньте таблицю процесів для трьох запущених процесів з урахуванням набором колонок: PID, PPID, STAT, NI, %CPU, CMD. Зробіть висновки за поточними значеннями NI та %CPU.
3. Зменшить пріоритет виконання одного з трьох процесів.
4. Перегляньте таблицю процесів для трьох запущених файлів з урахуванням набором колонок: PID, PPID, STAT, NI, %CPU, CMD. Зробіть висновки щодо змін значень% CPU для кожного процесу: як вони змінилися?

**Хід роботи**

# Завдання 1 Моделювання багатозадачності

Нехай оперативна пам'ять на комп'ютері-сервері становить *N* Гб. Системні процеси ОС займають до *M* Гбайт пам'яті, а кожна програма користувача може використовувати до *K* Гбайт пам'яті. Нехай в середньому процеси програм користувачів витрачають *F*% свого часу на очікування завершення вводу/виводу. Визначте середню завантаженість процесора, використовуючи значення з таблиці 3 варіантів.



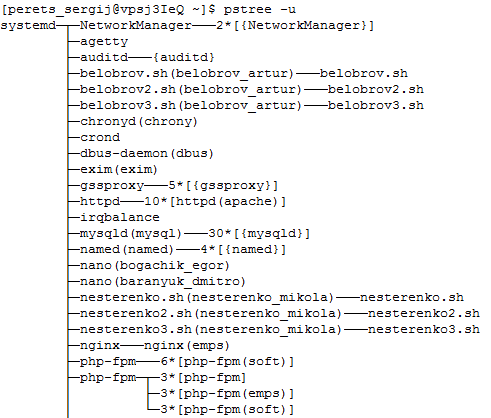
*N* = 11 Гб – оперативна пам’ять,

*M* = 2 Гб – макс. пам’ять, яку займають системні процеси ОС, *K* = 0,9 Гб – макс. величина пам’яті, яку займає одна програма,  *p* = *F*% = 0,3 – відсоток часу, витрачаємий в середньому на очікування завершення вводу/виводу

На програми виділяється в сумі до 12 – 1 = 11 Гб, а отже, одночасно в пам’яті можливо розмістити до *n* = 11 : 0,5 = 22 програм. Тоді середня завантаженість процесора становить 1 – *pn* = 1 – 0,316, або **близько 100%**.

# Завдання 2 Перегляд таблиці процесів

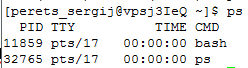
1. Отримайте ієрархію всіх процесів із зазначенням імен користувачів, які їх запустили.



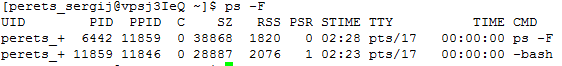
1. Отримайте ієрархію процесів, запущених від імені вашого профілю із зазначенням PID цих процесів.



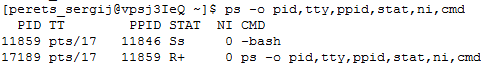
1. Отримайте список процесів, запущених в поточному терміналі, зі спрощеним набором колонок виведення таблиці процесів.



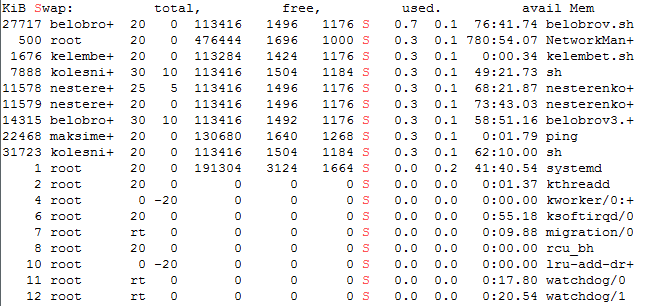
1. Отримайте список процесів, запущених від імені вашого користувача, з розширеним набором колонок таблиці процесів.



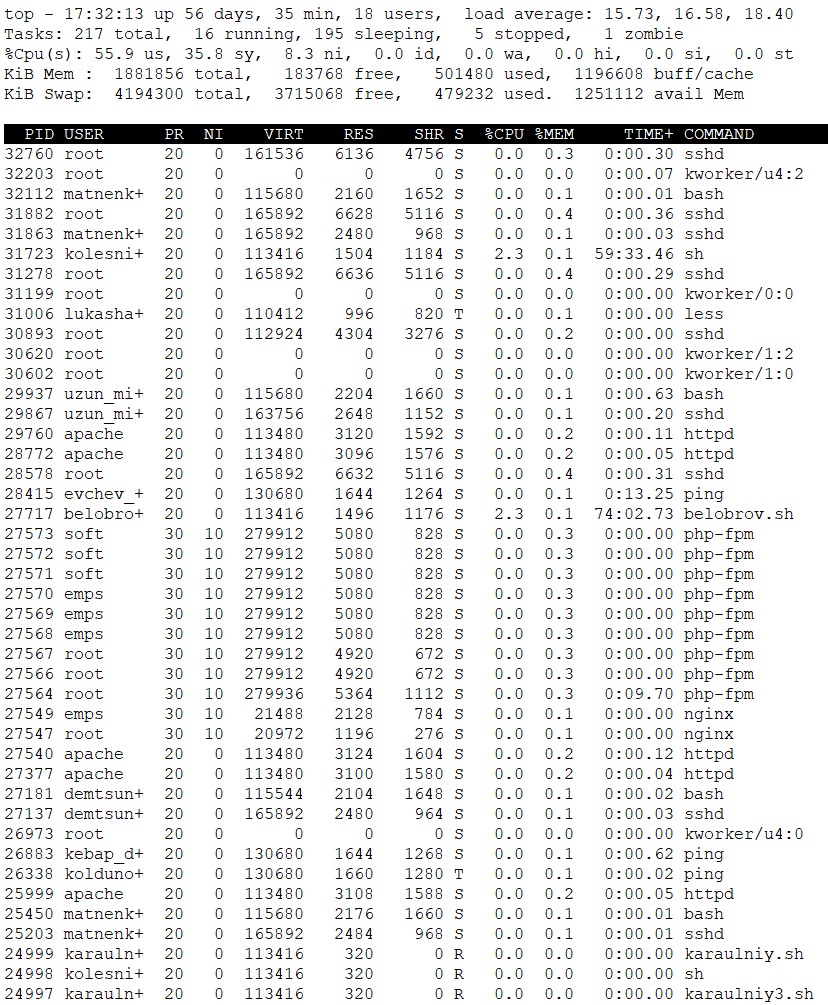
1. Отримайте список процесів, запущених від імені вашого користувача із зазначенням наступного набору колонок: PID, TTY, PPID, STAT, NI, CMD



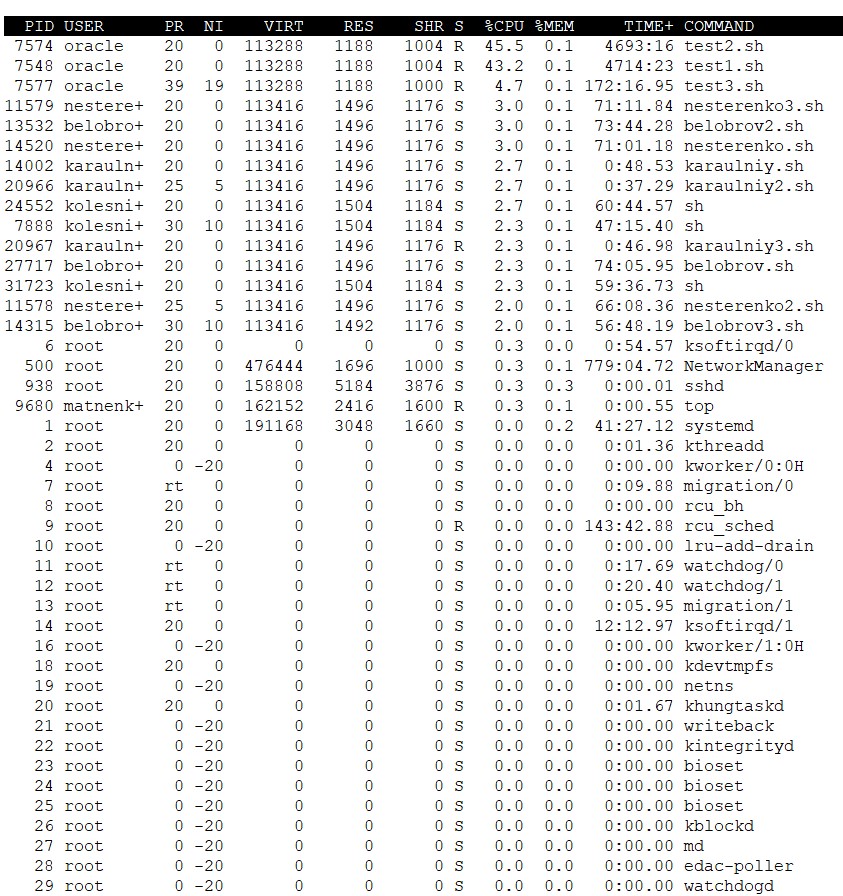
1. Отримайте список всіх сплячих процесів зі спрощеним набором колонок виведення таблиці процесів.



1. Отримайте список процесів, відсортованих по PID, і визначте:
2. загальна кількість запущених процесів;
3. кількість процесів, які виконуються;
4. кількість сплячих процесів.



8. Отримайте список процесів, відсортованих за % використання процесора.



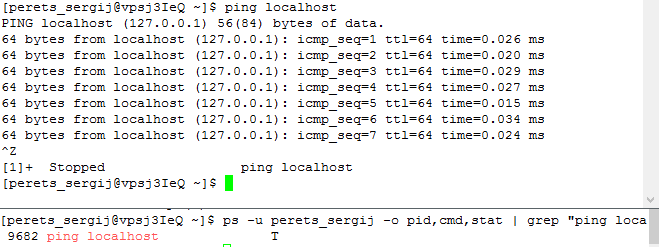
# Завдання 3 Керування станами процесів

1. У поточному терміналі виконайте команду ping localhost, але не завершуйте її роботу.

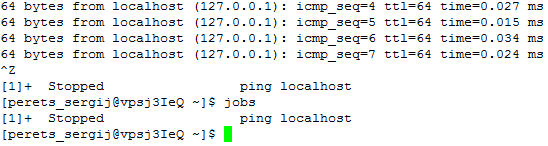


1. Запустіть другий термінал доступу до Linux-сервера.
2. У другому терміналі для команди ping отримаєте таблицю її процесу (колонки PID, STAT, CMD).
3. У другому терміналі призупиніть виконання процесу команди ping До пунктів

3-4:

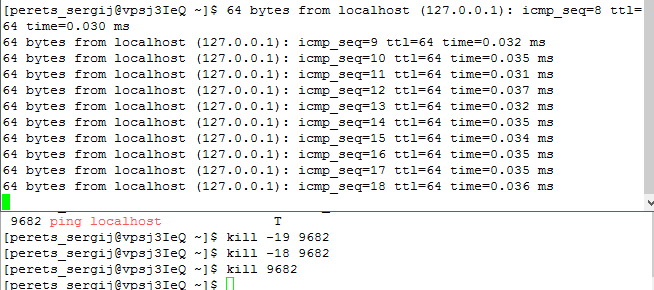


1. У першому терміналі отримайте список фонових процесів



1. У другому терміналі відновіть виконання припиненого процесу

7. У другому терміналі зупиніть виконання процесу команди ping До пунктів 6-7:

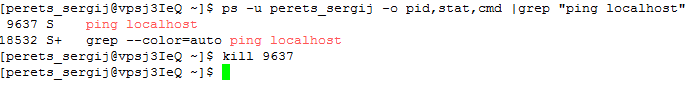


1. У першому терміналі запустіть команду ping в фоновому режимі так, щоб він не був автоматично зупинений навіть після закриття терміналу, з якого був запущений.



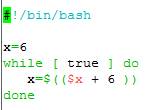
1. Закрийте перший термінал.
2. У другому терміналі для команди ping отримаєте таблицю її процесу (колонки PID, STAT, CMD). Зробіть висновок про стан процесу.
3. Завершіть роботу процесу.

До пунктів 10-11:



# Завдання 4 Управління пріоритетами процесів

1. Створіть bash-програму, що виконує операцію циклічного складання за формулою: x = x + n, де початкове значення х = кількість букв вашого прізвища, n - кількість букв у вашому імені. Ім'я програми збігається з транслітерацією вашого прізвища з розширенням .sh, наприклад, ivanov.sh Скрипт:



1. Запустіть bash-програму у фоновому режимі.



1. Перегляньте таблицю процесів для запущеного процесу, пов'язаного з bash- програмою, з урахуванням набором колонок: PID, PPID, STAT, NI,% CPU, CMD.
2. Виконайте команду призупинення запущеного процесу.

До пунктів 3-4:



1. Ще раз перегляньте таблицю процесів для призупиненого процесу з урахуванням набором колонок: PID, PPID, STAT, NI, %CPU, CMD. Зробіть висновки про його стан.

Стан процесу змінився: процес виконання зупинено.

1. Виконайте команду продовження виконання припиненого процесу.



1. Ще раз перегляньте таблицю процесів для процесу, який продовжив виконуватися, з урахуванням набором колонок: PID, PPID, STAT, NI, %CPU, CMD. Зробіть висновки про його стан.



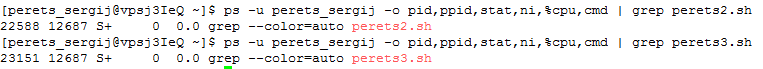
Стан процесу змінився: виконання процесу відновилося.

1. Створіть два файли як символічні посилання на створену bash-програму з іменами як ім'я поточного файлу з додаванням цифр 2 і 3, відповідно, наприклад: ivanov2.sh, ivanov3.sh



1. Запустіть два файли у фоновому режимі.

1. Перегляньте таблицю процесів для трьох запущених процесів з урахуванням набором колонок: PID, PPID, STAT, NI, %CPU, CMD. Зробіть висновки за поточними значеннями NI та %CPU.



Значення NI не змінилися. Значення %CPU для зв’язка …2 - найбільше, найменше

– для зв’язка …3.

1. Зменшить пріоритет виконання одного з трьох процесів.
2. Перегляньте таблицю процесів для трьох запущених файлів з урахуванням набором колонок: PID, PPID, STAT, NI, %CPU, CMD. Зробіть висновки щодо змін значень% CPU для кожного процесу: як вони змінилися? До пунктів 11-12:

Значення %CPU збільшилися для оригінального файлу та зв’язка …2, зменшилося для зв’язка …3; для оригінального файлу воно стало найбільшим.

**Висновок.** В процессі виконання лабораторної роботи ми засвоїли базові навички в управлінні процесами в ОС Unix засобами командної оболонки.