НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БІОРЕСУРСІВ І ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ

Кафедра комп'ютерних наук Дисципліна «Операційна система Linux »

# Лабораторна робота №6

Створити домен на базі ОС Linux.

Виконав Студент групи ІПЗ-20006б **Симон Дмитрій Вікторович**

# 2023

**Мета:** Набуття компетенцій по роботі з Web 2.0 технологіями. Основи програмування сокетів для TCP з'єднань мовою Python.

# Завдання №1

* здобути відомості про Web 2.0 технології, зокрема сформувати навички з інсталяції та конфігурації програмних продуктів LAMP-стеку;
* навчитися основам програмування сокетів для TCP з’єднань мовою Python в реалізаціях серверів і клієнтів.

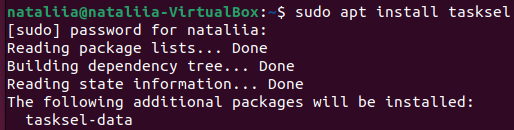
Web 2.0 - це більш інтерактивна та спільна версія інтернету і друге покоління інтернету, яким ми зараз активно користуємось.

На початку свого існування інтернет був здебільшого для читання інформації на статичних веб-сторінках, але з появою Веб 2.0 люди почали брати активну участь у створені контенту і взаємодіяти з іншими онлайн.

В порівнянні: Веб 1.0 був схожий на бібліотеку, де можна було читати книги. Веб 2.0 перетворив Інтернет на велику спільноту, де люди не лише читають, але й пишуть, діляться фотографіями, відео тощо. Тобто мова йде про всім відомі соціальні мережі, блоги, YouTube та інші інтерактивні веб-сайти.

# Хід роботи

1. Встановити стек LAMP та виконано проміжні налаштування. У терміналі віртуальної машини запустити такі команди:
2. apt install tasksel – встановлюємо утиліту для одночасного встановлення декількох пакетів;



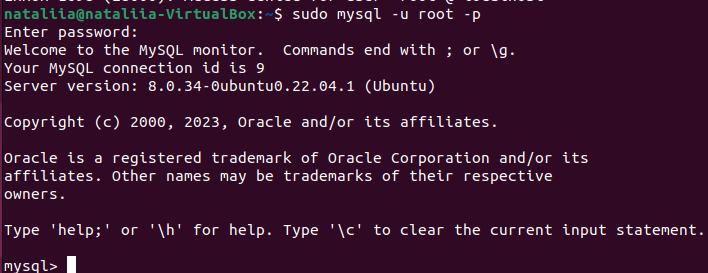
1. tasksel install lamp-server – встановлюємо LAMP-сервер;



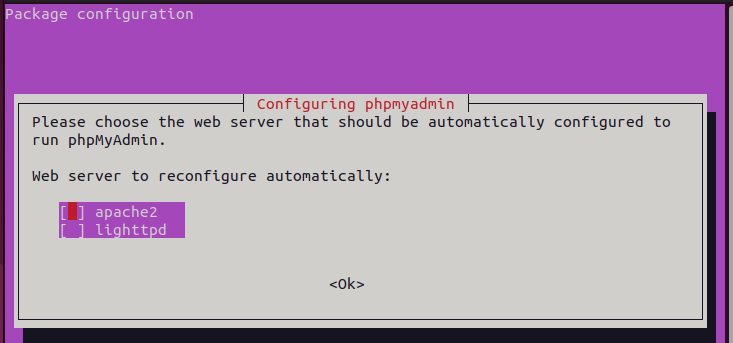
Встановлюємо mySQL server



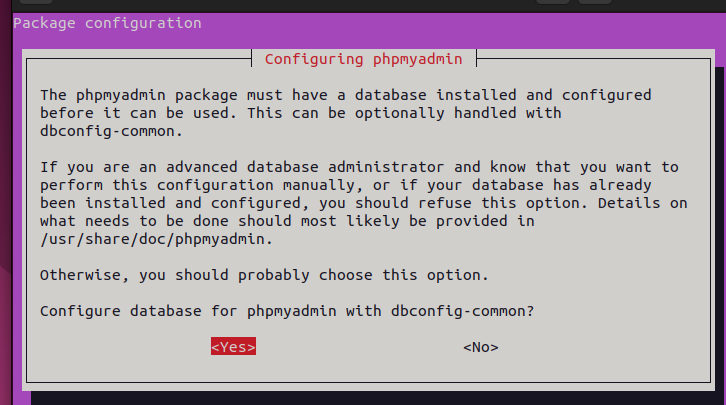
1. sudo mysql -u root -p – налаштовуємо пароль для адміністратора бази даних;



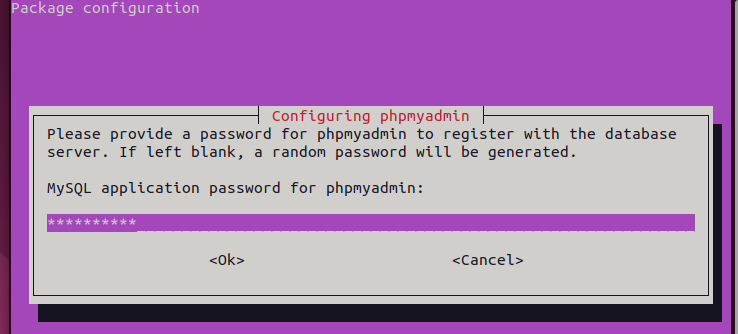
1. sudo apt-get install phpmyadmin – встановлюємо phpMyAdmin, у процесі встановлення обираємо веб-сервер apache2 і вводимо той пароль, що встановили в попередньому пункті;



Натискаємо YES.



Вводимо пароль



1. mysql -u root -p – входимо в консоль mysql як root (з відповідним паролем);
2. CREATE USER 'ІМ’Я\_КОРИСТУВАЧА'@'localhost' IDENTIFIED BY 'ЙОГО\_ПАРОЛЬ' – створюємо користувача;



1. GRANT ALL PRIVILEGES ON \*.\* TO 'ІМ’Я\_КОРИСТУВАЧА'@'localhost'

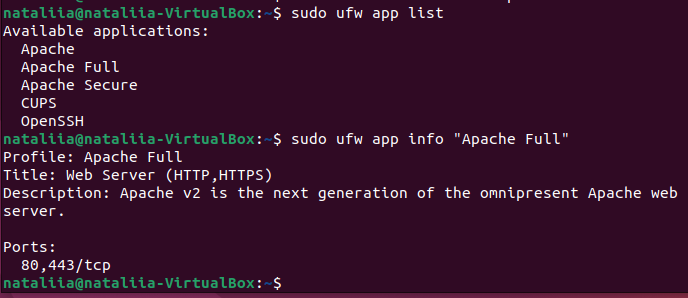
– даємо повні права цьому користувачу;



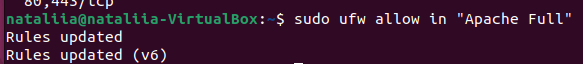
1. FLUSH PRIVILEGES – застосовуємо права; exit – виходимо із консолі mysql.



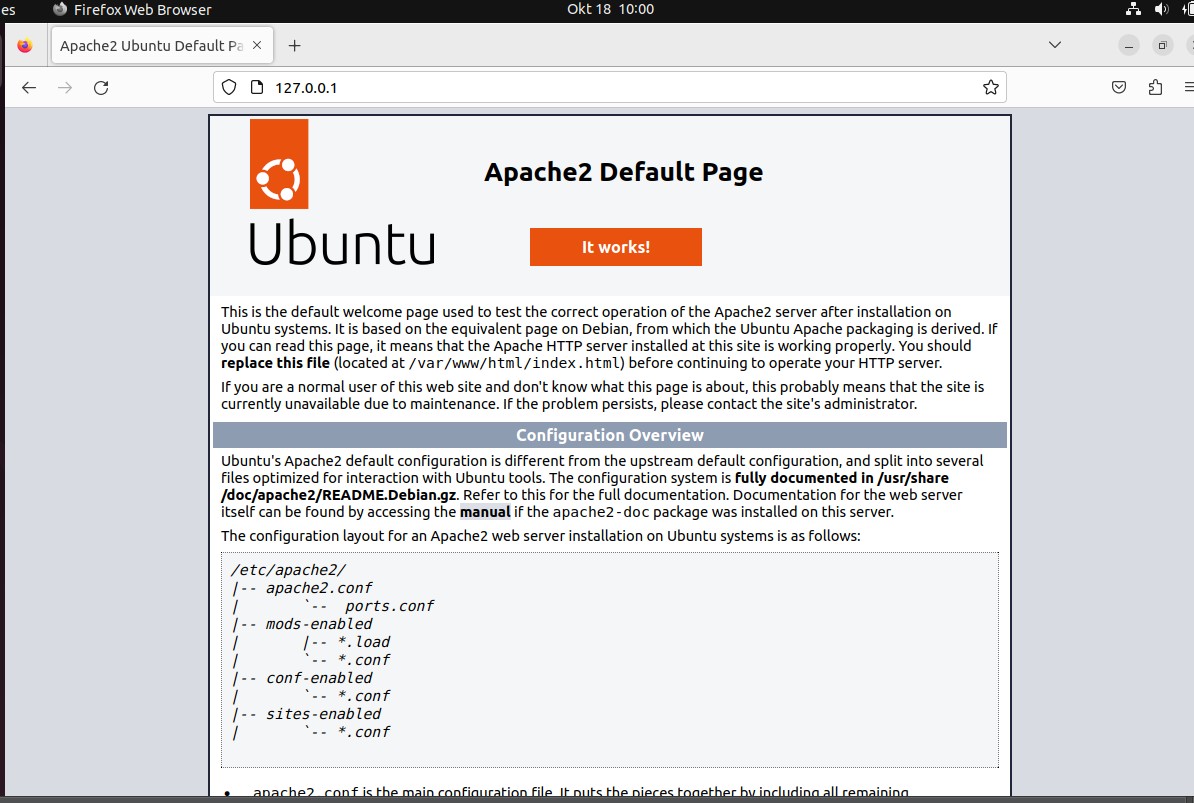
1. Перевіримо фаєрвол для дозволу веб-трафіку. Переконаємось, що UFW має профіль для Apache: sudo ufw app list
2. Перевіримо налаштування профіля Apache Full, яка повинна дозволяти трафік для портів 80 та 443. sudo ufw app info "Apache Full"



1. Дозволимо вхідний HTTP HTTPS трафік для нашого профіля. sudo ufw allow in "Apache Full"

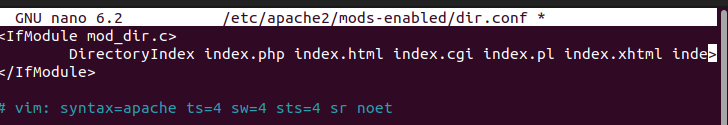


1. Перевіримо результат установки, відкривши в браузері публічну IP- адресу сервера



1. Перевіримо функціонування серверу. Для цього змінимо шлях Apache який надає файл, коли запитується директорія. Стандартно, Apache відкриває index.html, змінимо його на index.php. Відкриємо sudo nano

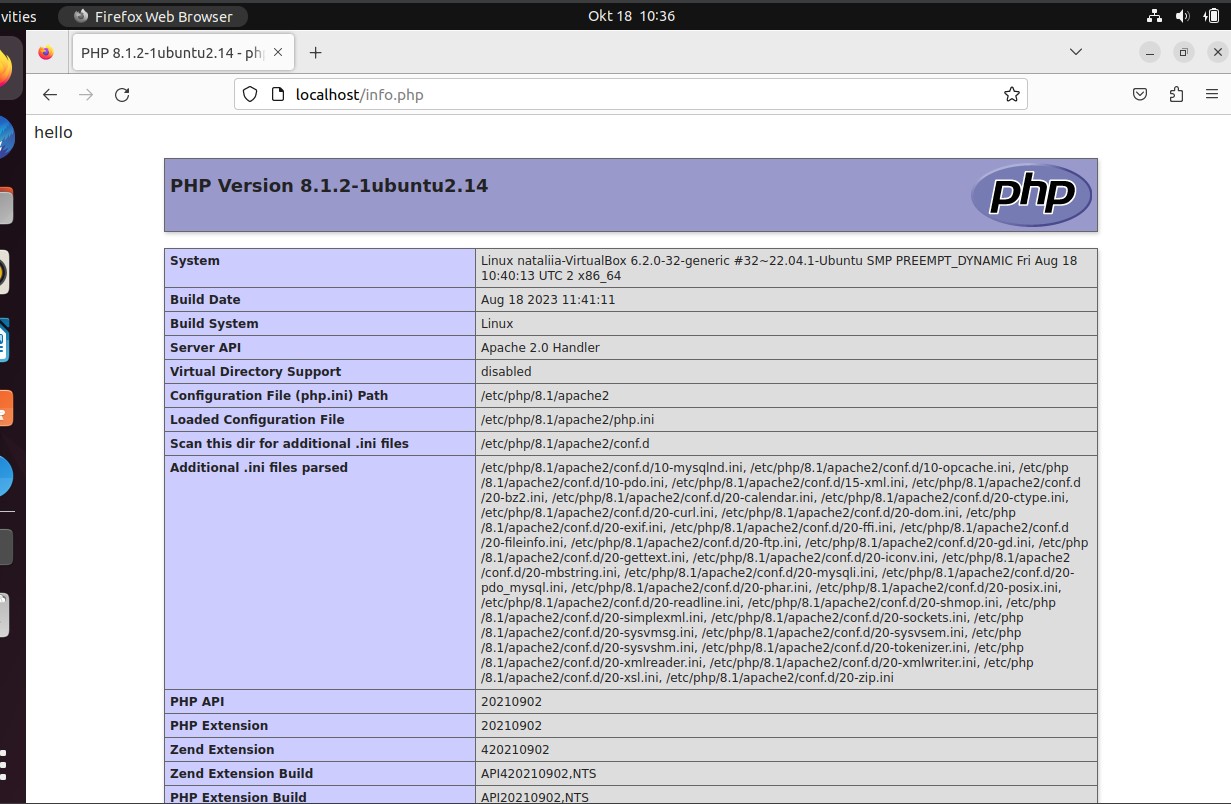
/etc/apache2/mods-enabled/dir.conf та модифікуємо даний скрипт.



1. Перевіримо функціонування серверу. Створимо sudo nano

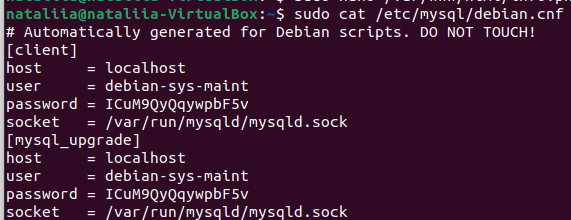
/var/www/html/info.php, в якому помістимо наступний скрипт. Збережемо файл.





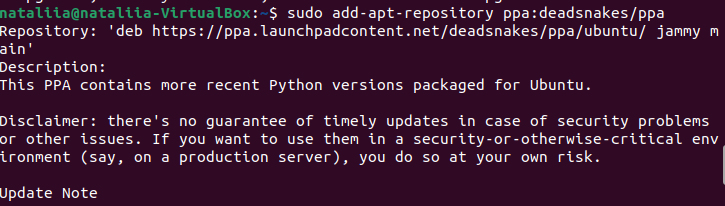
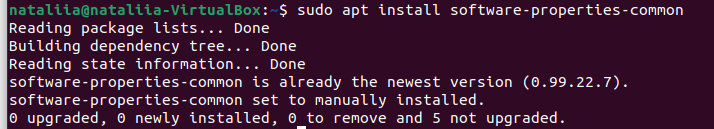
Таким чином, було встановлено стек Lamp та створено сервер для наступного виконання програмування стеків протоколу TCP.

Дані користувача mysql знаходяться за адресою cat /etc/mysql/debian.cnf.

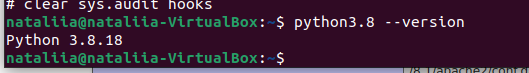


1. Програмування сокетів для TCP з’єднань мовою Python
2. Встановимо дистрибутив Python для подальшого програмування. Встановимо необхідні бібліотеки sudo apt install software-properties-common,

додамо deadsnakes PPA в список джерел системи sudo add-apt-repository ppa:deadsnakes/ppa.



1. Як тільки репозиторій буде ввімкнено, встановимо Python sudo apt install python3.8 та перевіримо його інсталювання python3.8 –version



За допомогою перевірки версії, бачмо що пайтон було успішно встановлено. Створимо програму веб-сервер, яка містить наступний код:

import socket

host = '127.0.0.1' # IP-адреса сервера port = 12345 # Порт сервера

with socket.socket(socket.AF\_INET, socket.SOCK\_STREAM) as server\_socket: server\_socket.bind((host, port))

server\_socket.listen()

print(f"Сервер слухає на {host}:{port}") conn, addr = server\_socket.accept()

with conn:

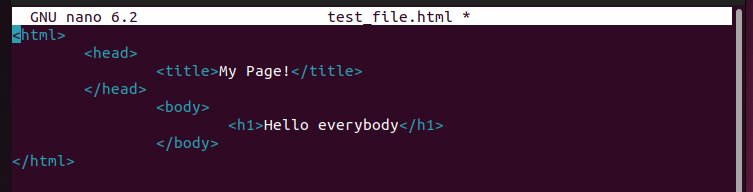
print(f"З'єднання встановлено з {addr[0]}:{addr[1]}") while True:

data = conn.recv(1024) if not data:

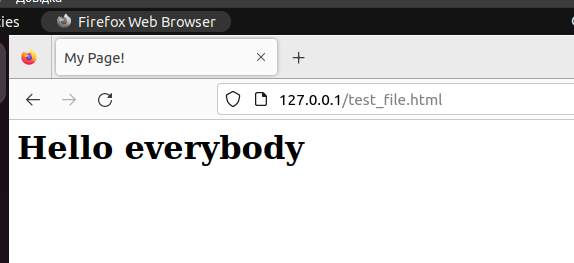
break conn.sendall(data)



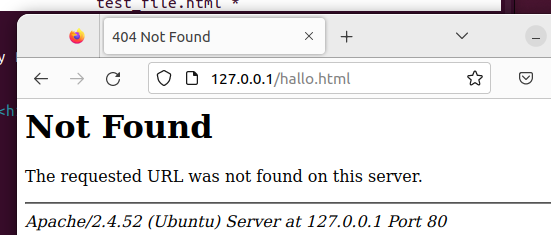
Після написання програми веб-серверу, створимо тестову сторінку на нашому сервері в директорії /var/www/html за допомогою команди vim test\_file.html.



Відкриємо створений файл у браузері хоста сервера та іншого хоста. Для цього в адресний рядок введемо [http://127.0.0.1/test\_file.html.](http://127.0.0.1/test_file.html)



При запиті неіснуючого файлу отримуємо помилку 404.



# Висновок.

В даній роботі було опановано принципи роботи з програмами та текстом в операційній системі Linux. Використано навички застосування дозволів та керування ними, встановлення для інших користувачів.