

## Лабораторна робота №2

**Тема:** налаштування та конфігурація ізольованих мереж у Docker із використанням VLAN для забезпечення сегментації трафіку між контейнерами.

**Мета:** Ознайомитися з механізмом VLAN у Docker. Навчитися створювати ізольовані мережі за допомогою macvlan. Забезпечити ізоляцію трафіку між контейнерами, що знаходяться в різних VLAN. Перевірити взаємодію контейнерів у межах одного VLAN та відсутність зв'язку між різними VLAN.

### Хід роботи:

- Визначити доступні мережеві інтерфейси у системі.
- Створити дві ізольовані VLAN-мережі у Docker (vlan10 та vlan20).
- Запустити два контейнери та підключити їх до відповідних VLAN.
- Перевірити IP-адреси контейнерів.
- Перевірити мережеву ізоляцію між VLAN.
- Оформити звіт із висновками щодо результатів роботи.

### Очікуваний результат:

- Контейнери у різних VLAN не повинні мати зв'язок між собою.
- Контейнери в межах одного VLAN мають успішно комунікувати.
- Отримані висновки щодо роботи VLAN у Docker.

### Основні команди

#### Перевірка доступних мережевих інтерфейсів:

`ip link show`

#### Створення VLAN-мереж у Docker:

```
docker network create -d macvlan \
```

```
--subnet=192.168.10.0/24 \
```

```
--gateway=192.168.10.1 \
```

```
-o parent=eth0 \
```

```
vlan10 docker network create -d macvlan \
```

```
--subnet=192.168.20.0/24 \
```

```
--gateway=192.168.20.1 \
```

```
-o parent=eth0 \ vlan20
```

### **Створення та підключення контейнерів до VLAN:**

```
docker run -dit --name container1 --network=vlan10 alpine sh
```

```
docker run -dit --name container2 --network=vlan20 alpine sh
```

### **Перевірка IP-адрес контейнерів:**

```
docker inspect container1 | grep "IPAddress"
```

```
docker inspect container2 | grep "IPAddress"
```

### **Перевірка зв'язку між контейнерами:**

```
docker exec -it container1 ping -c 3 <IP-адреса container2>
```