

XXIII. Thực hành quản lý Docker container

- Chúng ta cần phải đăng nhập với quyền root , vì vậy hãy chạy "sudo -i" ngay khi đăng nhập.

1. Chạy image hello-world và busybox

- Đầu tiên chúng ta kiểm tra xem Docker đã được cài đặt và dịch vụ docker.service đã được kích hoạt chưa:

```
docker run hello-world
```

```
docker run --name hi hello-world
```

- Nếu chúng ta không chắc image đang ở đâu, hoặc chính xác là tên gọi là gì, chúng ta có thể tìm kiếm:

```
docker search busybox
```

- Chúng ta muốn image được đưa lên đầu danh sách và có thể tải nó về server bằng lệnh:

```
docker pull docker.io/busybox
```

- Bây giờ chúng ta có thể chạy image này. Chúng ta sẽ đặt tên nó là "busy" và thiết lập thông số cho phép chúng ta tương tác với container trong terminal:

```
docker run --name busy -it busybox /bin/sh
```

- Nhập "exit" để thoát.

2. Hiển thị container đang chạy

- Chúng ta có thể xem các containers đang chạy sử dụng lệnh "docker ps". Nếu chúng ta muốn xem tất cả các container của chúng ta, ngày cả là các containers này có đang chạy hay không, thì chúng ta chạy lệnh sau:

```
docker ps -a
```

- Chúng ta sẽ thấy một số container có trong hệ thống. Vậy Hãy xóa bỏ container "hi":

```
docker rm hi
```

- Chạy lại "docker ps -a" để kiểm tra nó đã bị xóa chưa.
- Hiển thị các images
- Để hiển thị images trong kho lưu trữ cục bộ trong server, chúng ta có thể chạy lệnh này:

```
docker images
```

3. Nếu chúng ta không cần image nữa, chúng ta có thể loại bỏ nó, thực hiện các bước sau:

- Dừng tất cả các container đang sử dụng image "hello-world" bằng cách chạy lệnh:

```
docker stop $(docker ps -a -q --filter ancestor=hello-world)
```

- Xóa tất cả các container đang sử dụng image "hello-world" bằng cách chạy lệnh:

```
docker rm $(docker ps -a -q --filter ancestor=hello-world)
```

- Xóa image "hello-world" bằng lệnh:

```
docker rmi -f hello-world
```

- Trong trường hợp này, chúng ta đã xóa image "hello-world". Kiểm tra xem nó đã bị xóa hay chưa, bằng cách chạy lệnh sau:

```
docker images
```

- Nếu image này đã ko còn được hiển thị, kết quả sẽ không còn chứa image "hello-world" nữa.

4. Tạo một Container apache2 dựa trên Image httpd:2.4 và ánh xạ localhost: 8080 đến Cổng 80 của Container

- Chúng ta sẽ sử dụng lệnh "docker run" để tạo một container có tên là "apache2", dựa trên image "httpd:2.4". Chúng ta cũng sẽ ánh xạ cổng 8080 trên localhost thành cổng 80 trên container bằng cách sử dụng "-p 8080:80".

```
docker run --name apache2 -p 8080:80 httpd:2.4
```

- Để thoát khỏi container đang chạy, nhấn Ctrl + c. Sau đó để kiểm tra nó có tồn tại hay không, chạy:

```
docker ps -a
```

- Để khởi động container, chúng ta có thể sử dụng lệnh:

```
docker start apache2
```

- Bây giờ chúng ta sẽ có thể thấy container này với lệnh "docker ps" (không có tùy chọn "-a").

5. Thêm thông tin

- Nếu chúng ta cần xử lý sự cố hoặc muốn biết có vấn đề gì đang xảy ra với container, chúng ta có thể xem các nhật ký logs và thống kê của container bằng hai lệnh sau:

```
docker logs apache2
```

```
docker stats apache2
```

- Chúng ta phải nhấn Ctrl + c để thoát khỏi lệnh thứ hai.
- Container này dự kiến cung cấp một trang web, vì vậy hãy kiểm tra
lynx -dump http://localhost:8080

6. Trong Container apache2, chúng ta sẽ cập nhật file index.html mặc định, sau đó xác nhận và lưu các thay đổi

- Để thực thi một lệnh bên trong 1 container đang chạy, chúng ta phải chạy lệnh sau đây :

```
docker exec -it apache2 bash
```

- Trong container, chúng ta sẽ được làm việc dưới quyền root và đang ở trong thư mục /usr/local/apache2. Chạy lệnh "ls" để xem có gì ở trong thư mục này, sau đó chạy "cd htdocs" để vào thư mục trang web index.html.
- Bây giờ chúng ta muốn thay đổi trang web để tất cả người dùng có thể truy cập xem container apache2 :

```
echo 'apache2 container' > index.html
```

- Sau đó, chúng ta có thể chạy lệnh exit để thoát khỏi container, và kiểm tra xem thay đổi của chúng ta có tác dụng hay không bằng cách chạy lệnh sau:

```
lynx -dump http://localhost:8080
```

- Nếu chúng ta khởi động lại container ở bước này, thay đổi của chúng ta sẽ bị mất, do đó chúng ta sẽ commit nó.
- Vậy chúng ta sẽ commit các thay đổi và khởi động lại container apache2 để xác nhận là các thay đổi đã được áp dụng. Bằng cách sử dụng lệnh:

```
docker commit -m 'Updated index.html' apache2
```

- Chúng ta đã commit các thay đổi cho container. Sau đó, chúng ta sẽ dừng container bằng lệnh:

```
docker stop apache2
```

- Và kiểm tra xem container đã stop bằng lệnh:

```
docker ps -a
```

- Tiếp theo, chúng ta khởi động lại container bằng lệnh:

```
docker start apache2
```

- Và cuối cùng, chúng ta sẽ kiểm tra website mà container đang cung cấp bằng lệnh:
lynx -dump http://localhost:8080