1. Quy trình triển khai docker

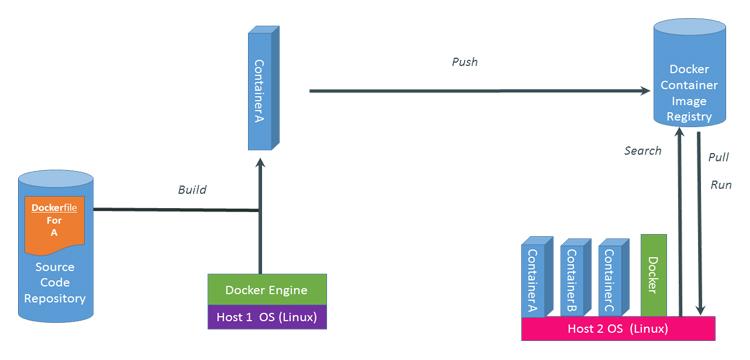
Docker được xây dụng dựa trên 3 thành phần chính:

* Docker daemon: daemon (server) nhận lệnh từ docker client thông qua CLI hoặc RestAPI
* Docker client: Có thể ở trên cùng một host hoặc khác host với docker daemen
* Docker registry (hub): là một dịch vụ máy chủ cho phép lưu trữ các docker image của cá nhân, công ty, team,… Dịch vụ Docker Registry có thể được cung cấp bởi tổ chức thứ 3 hoặc là dịch vụ nội bộ được xây dựng riêng nếu bạn muốn. Một số dịch vụ Docker Registry phổ biến như : Azure Container Registry, Docker Hub...

Một hệ thống Docker thường sẽ được thực thi dựa theo 3 bước chính như sau:

**Bước 1: Build**

Đầu tiên, bạn cần tạo một dockerfile, trong dockerfile này thì nó chính là code của chúng ta. Nó sẽ được Build tại một máy tính đã được cài đặt sẵn Docker Engine. Sau khi đã build, thì ta sẽ có được Container, trong Container đã có chứa ứng dụng đi kèm bộ thư viện của chúng ta.



*Quy trình, các bước hoạt động của Docker là gì?*

**Bước 2: Push**

Sau khi đã có được Container thì chúng ta cần phải thực hiện Pusch Container này lên cloud rồi lưu nó tại đó.

**Bước 3: Pull và Run**

Nếu như một máy tính khác đang muốn sử dụng Container thì chúng ta cần buộc máy thực hiện việc Pull container này về máy. Tất nhiên, máy này cũng cần phải được cài Docker Engine, sau đó sẽ thực hiện Run Container.

1. **Các câu lệnh thường dùng với Docker Container**

* **docker container commit:** Tạo image mới từ những thay đổi của container.
* **docker container create:** Tạo container mới.
* **docker container exec:** Chạy các command khi container đang hoạt động.
* **docker container kill:** Chấm dứt hoạt động của một hoặc nhiều container.
* **docker container pause:** Tạm dừng tất cả tiến trình bên trong một hoặc nhiều container.
* **docker container run:** Chạy các command trong một container mới.
* **docker container start:** Chạy một container hoặc nhiều container đã dừng.
* **docker container rename:** Đổi tên container.
* **docker container restart:** Khởi động lại một hoặc nhiều container.

1. **Các câu lệnh thường dùng với Docker Image**

* **docker images:** Liệt kê các image
* **docker image build:** Build image từ file Dockerfile.
* **docker image history:** Hiện thị lịch sử của image.
  + **docker image import:** Import nội dung từ *tarball* để tạo ra filesystem của image.
  + **docker image inspect:** Hiển thị thông tin chi tiết của một hoặc nhiều image.
  + **docker image load:** Nạp image từ file \*.tar hoặc STDIN.
  + **docker image prune:** Xóa các image không sử dụng.
  + **docker image pull:** Pull một image hoặc repository từ Docker HUB đăng ký.
  + **docker image push:** Đẩy image, repository lên Docker HUB.
  + **docker image save:** Lưu một hoặc nhiều image vào file \*.tar.
  + **docker image tag:** Gắn tag cho TARGET\_IMAGE tương ứng với SOURCE\_IMAGE.ư

1. **So sánh docker và kubernetes**

Sự khác nhau giữa Kubernetes và Docker là:

* + - Docker là một nền tảng container hóa giúp dễ dàng phát triển, đóng gói ứng dụng từ môi trường này sang môi trường khác.
    - Kubernetes là một hệ thống giúp điều phối các container cho các nền tảng như Docker.
* Như vậy, chúng không phải là đối thủ của nhau, do đó không thể so sánh trực tiếp vì cả hai giải quyết những vấn đề khác nhau.
* Nếu thực sự muốn so sánh hai khái niệm này thì điểm khác biệt chính giữa chúng là Docker chỉ chạy trên một node còn Kubernetes lại chạy trên một cụm - cluster. Ngoài ra, Docker có thể được sử dụng mà không cần tới Kubernetes, còn Kubernetes phải cần một môi trường container hóa như Docker để hoạt động.
* Tùy vào tình huống thực tế việc kết hợp cả Docker và Kubernetes sẽ tăng thêm tính hiệu quả trong việc phát triển phần mềm.