### Giới thiệu chung về Docker

* Docker là một nền tảng cho phép dễ dàng building, deploying (shipping) và running ứng dụng
* Nó gần giống như máy ảo nhưng có một sự khác biệt về bản chất

| Máy ảo | Docker |
| --- | --- |
| -Cần cài lớp ảo (Hypervisor).  -Nó sẽ phân chia tài nguyên của máy (CPU, RAM)  -App chạy trên máy ảo thì sử dụng các thư viện, hệ điều hành của máy ảo đó | -Không phải chia sẻ tài nguyên  -Sử dụng chính hệ điều hành của máy host và bổ sung thêm các thư viện cần thiết.  => Chạy nhanh hơn |

### 2. Một số khái niệm cơ bản

* Container là một môi trường “cách ly” gồm đầy đủ các thư viện, file cài config…để ứng dụng chạy được.
* Image là một template để tạo container. Hiểu cách khác là các phần mềm đã được đóng gói và quản lý trong docker.
* Chỉ có thể đọc mà không thể thay đổi
* Thực thi images tạo ra containers

### 3. Các câu lệnh

| Docker search imageName: Tìm kiếm một image  Docker images: hiển thị toàn bộ images đang có  Docker ps: hiển thị toàn bộ container đang chạy  Docker ps -a: hiển thị toàn bộ container kể cả không chạy  Docker pull imageName: Tải về một image | |
| --- | --- |
| Docker run imageName:tag/idImage: Tạo và run một container từ một image  --name ten\_bien : Đặt tên container  -it:  -v: chia sẻ dữ liệu  --volumes-from: Mount volumes từ container xác định  --mount: Attach a filesystem mount to the container  --rm : tự động xóa container khi chạy xong | |
| Exit: thoát và tắt container | -Ctrl P+ctrl Q: thoát ra khỏi container (về host) nhưng container vẫn chạy  -docker stop tên\_container |
| Docker start id/name\_container: chạy một container | |
| Docker exc containerName command: thực thi một lệnh cho một container khi đang ở host | |
| Docker attach containerName: từ host nhảy vào trong container có tên là containerName | |
| Docker rm id/name\_container: xóa một container  -f: xóa kể cả khi container đang chạy | |

* Tạo image từ container:

Docker commit containerName đặt\_tên\_image:tag

* Lưu images ra file tar:

Docker save imageName > path\_to\_file\_tar

* Import image

Docker import path\_to\_file\_tar

### 4. Chia sẻ dữ liệu trong docker

#### Chia sẻ dữ liệu giữa container-host và container-container

* Container-host (Container đồng thời ghi dữ liệu trong container và một thư mục ở host)
* Docker run -v path\_host:path\_container containerName:tag/containerId
* Container-container
* Docker run -v --volumes-from containerName:tag/containerId

#### Volume

* Truy cập, ghi dữ liệu vào volume, khi xóa container thì dữ liệu không bị xóa ở volume

Docker run --mount source= volume\_name,target=path containerName:tag/containerId

* Ánh xạ dữ liệu ở volume lên máy host

Docker volume create --opt device= path --opt type=none --opt o=bind tên\_volume

* Khi truy cập vào volume đã ánh xạ vào host thì không dùng --mount mà dùng -v: docker run -v tên\_volume:path containerName:tag/containerId
* Kiểm tra thông số volume: docker volume inspect tên\_volume

### 5. Network

* Khi run các container mặc định kết nối với mạng bridge
* Tạo network: docker network creat -d bridge tên\_mạng
* Kết nối container với mạng nào đó

Docker run --network tên\_mạng containerName

* Kết nối một container đang chạy tới network

Docker network connect tên\_network container\_cần\_kết\_nối

* Kiểm tra mạng nào đó

Docker network inspect tên\_mạng