

도커와 쿠버네티스 첫 걸음

2.1. 컨테이너 이미지의 생성, 실행, 공유

- Dockerfile : 이미지를 빌드할 때 수행할 지시 사항 목록.
- 이미지 레이어 : 이미지는 여러 개의 레이어로 구성된다. 여러 이미지에서 레이어를 공유할 수 있으므로 이미지 저장 및 전송이 훨씬 효율적.
 - 이미지를 만들 때 도커 파일의 각 명령에 대해 새 레이어가 만들어진다.
- `docker build -t kugia` : kugia라는 이미지 빌드.
- `docker run -name kugia-container -p 8080:8080 -d kugia` : 이미지에서 새 컨테이너 실행.
 - `-d` : 콘솔에서 분리되어 백그라운드에서 실행.
 - `-p` : 로컬 컴퓨터의 포트 8080은 컨테이너 내부의 포트 8080에 매핑됨.
- `docker ps` : 실행 컨테이너 목록 조회.
 - `-a` : 중지된 컨테이너까지 출력.
- `docker inspect <컨테이너-이름>` : 개별 컨테이너의 자세한 정보 조회.
- `docker exec -it <컨테이너-이름> bash` : 컨테이너 내부에서 bash 실행.
- `docker stop <컨테이너-이름>` : 컨테이너 중지.
- `docker rm <컨테이너-이름>` : 컨테이너 삭제.
- `docker tag <이미지-이름> <도커허브-id>/<이미지-이름>` : 이미지에 태깅 → 도커 허브에 이미지 푸시 가능.
- `docker push <도커허브-id>/<이미지-이름>` : 도커 허브에 이미지 푸시.

2.2. 쿠버네티스 클러스터 설정

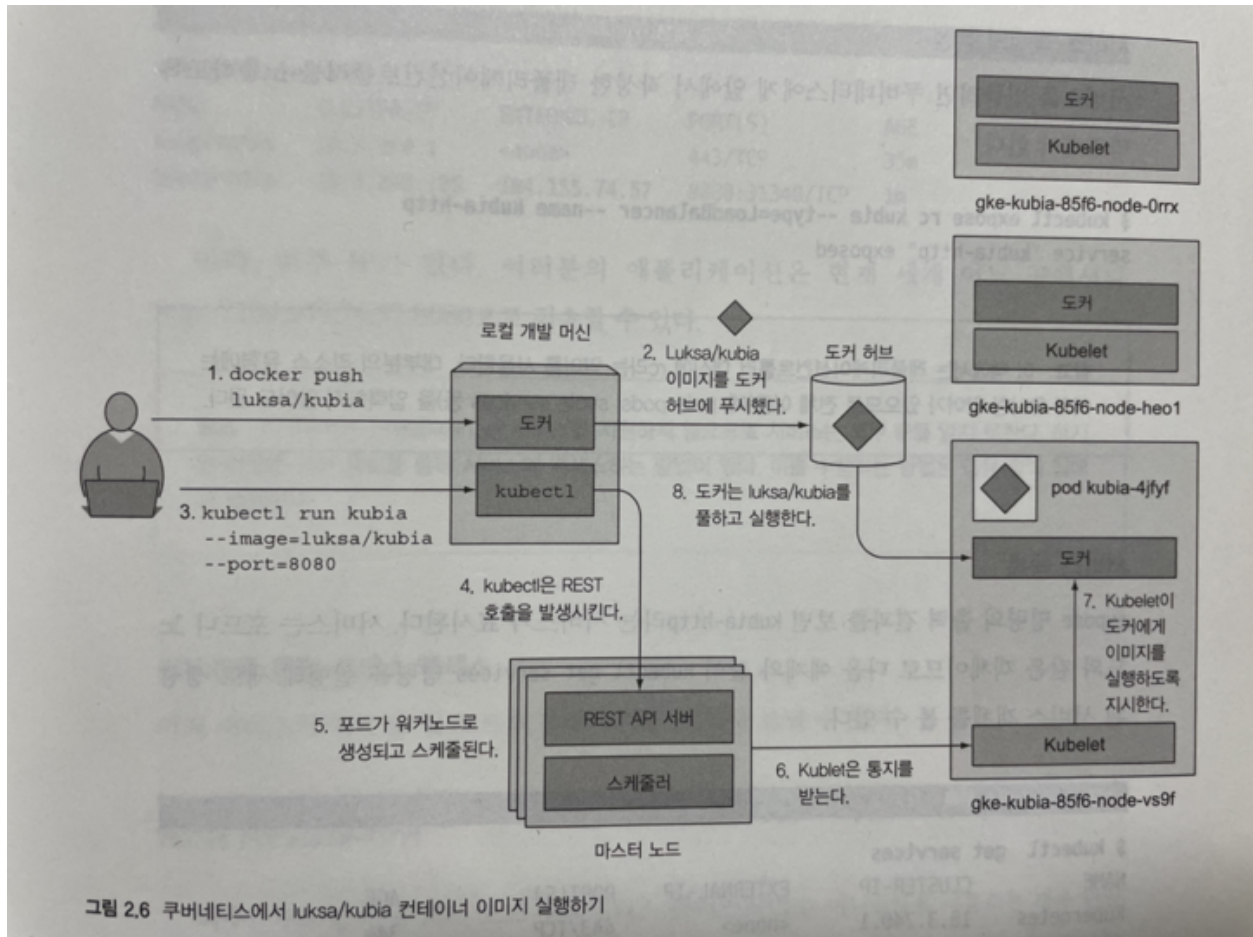
- minikube : 쿠버네티스를 테스트하고 로컬에서 애플리케이션을 개발하는 데 유용한 단일 노드 클러스터를 설정하는 도구.
- `kubectl cluster-info` : 클러스터가 작동하고 있는지 확인.

- `kubectl get nodes` : 클러스터의 모든 노드 나열.
- `kubectl describe node <노드-이름>` : 객체 세부사항 출력.

2.3. 쿠버네티스에서 첫 번째 앱 실행

보통 배포하려면 JSON 또는 YAML 매니페스트를 이용하지만 명령어로 앱을 실행해보자.

- `kubectl run <레플리케이션컨트롤러-이름> --image=<도커허브-id>/<이미지-이름> --port=8080 --generator=run/v1`
 - `--image` : 실행하려는 컨테이너 이미지 지정.
 - `--port` : 앱이 포트에서 수신 대기하고 있음.
 - `--generator` : 보통은 사용하지 않지만, 여기서는 디플로이먼트 대신 레플리케이션 컨트롤러를 생성.
 - ⇒ 레플리케이션 컨트롤러 생성
- **포드(Pod)** : 하나 이상의 밀접하게 관련된 컨테이너로 구성된 그룹.
 - 동일한 리눅스 네임스페이스와 동일한 워커 노드에서 항상 함께 실행됨.
 - 애플리케이션을 실행하는 별도의 논리적 시스템.
 - 자체 IP, 호스트 이름, 프로세스 등이 있다.
 - `kubectl get pods` : 파드 나열.



- 레플리케이션 컨트롤러는 새로운 파드를 생성하고 스케줄러에 의해 워커 노드 중 하나에 스케줄된다.

웹 애플리케이션 액세스

- 파드에는 자체 IP 주소가 있지만, 이 주소는 클러스터 내부에 있어 외부에서 액세스 불가능
⇒ **로드밸런서** 형태의 서비스 생성.
- 로드밸런서의 공용 IP를 통해 파드에 연결 가능.
- `kubectl expose pod kubia --type=LoadBalancer --name kubia-http`
: 파드를 노출.
- `kubectl get services` : 서비스 객체 조회.
- (참고) 미니큐브는 로드밸런서 서비스를 지원하지 않으므로 외부 ip를 얻지 못함. 하지만 `minikube service kubia-http`를 실행해 서비스에 액세스할 수 있는 ip 및 포트를 얻을 수 있음.

- http-kubia가 필요한 이유 : 파드는 언제든지 사라지고 대체될 수 있기 때문에 끊임없이 변화하는 파드 IP 주소 문제를 해결하고 여러개의 파드를 노출시킨다.

애플리케이션의 수평 스케일링

- 파드는 레플리케이션 컨트롤러가 관리
- `kubectl get replicationcontrollers`
 - desired 열 : rc가 유지해야할 파드 복제본 수.
 - current 열 : 현재 실행 중인 파드의 실제 수.
- `kubectl scale rc Nubia --replicas=3` : 파드는 항상 3가지 인스턴스를 유지해야 함.
- `kubectl get pods -o wide` :
 - -o : wide를 사용해 추가 열 표시 요청 가능. 파드의 IP와 파드가 실행되는 노드 표시.
- `kubectl describe pod <pod-name>` : 파드가 예정된 노드, 시작된 시간, 실행 중인 이미지 등 세부 정보 표시 가능.

쿠버네티스 웹 대시보드도 제공됨.