

1.3 쿠버네티스

🕒 생성일	@2021년 3월 11일 오후 10:33
☰ 태그	

1,2. 쿠버네티스의 기원 및 넓은 관점

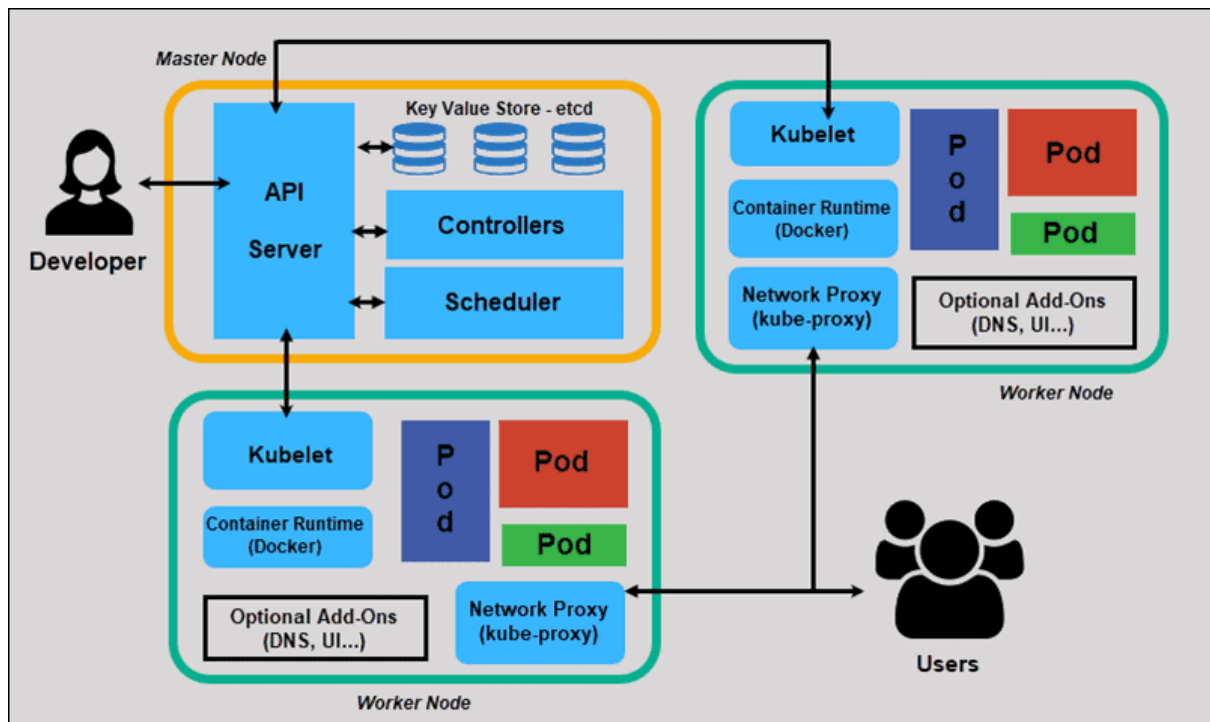
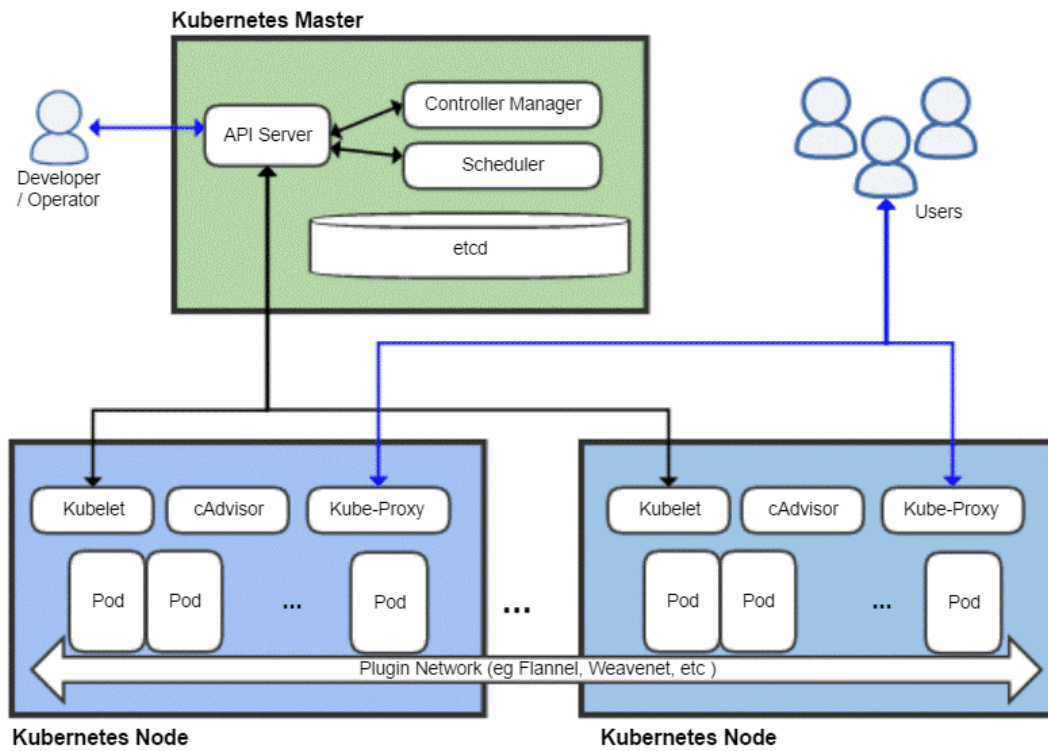


쿠버네티스 : 컨테이너화된 애플리케이션을 쉽게 배포하고 관리할 수 있게 해주는 소프트웨어 시스템

- 클러스터에 있는 노드의 개수와 상관없이 쿠버네티스에 애플리케이션을 배포하는 것은 항상 동일
- 모든 노드가 하나의 거대한 컴퓨터인 것처럼 실행 가능

3. 쿠버네티스 클러스터 아키텍처의 이해

쿠버네티스 클러스터의 노드는 두가지 유형의 노드로 나눌 수 있다.



- * 마스터노드 : 전체 쿠버네티스 시스템을 제어하고 관리, *컨트롤플레인* 실행
- * 워커노드 : 실제 배포되는 컨테이너 애플리케이션을 실행

컨트롤 플레인

: 클러스터 제어 및 작동 시킴

- 쿠버네티스 API 서버 : 개발/운영자, 컨트롤 플레인 구성요소와 통신
- 스케줄러 : 앱 배포 담당 - 앱이 각 구성요소를 워크 노드에 할당
- 컨트롤러 매니저 : 구성요소 복제본, 워커노드 추적, 노드 장애 처리 등의 모니터링 수행 (클러스터 단의 기능)
- etcd : 클러스터 구성 정보의 분산 데이터 저장소

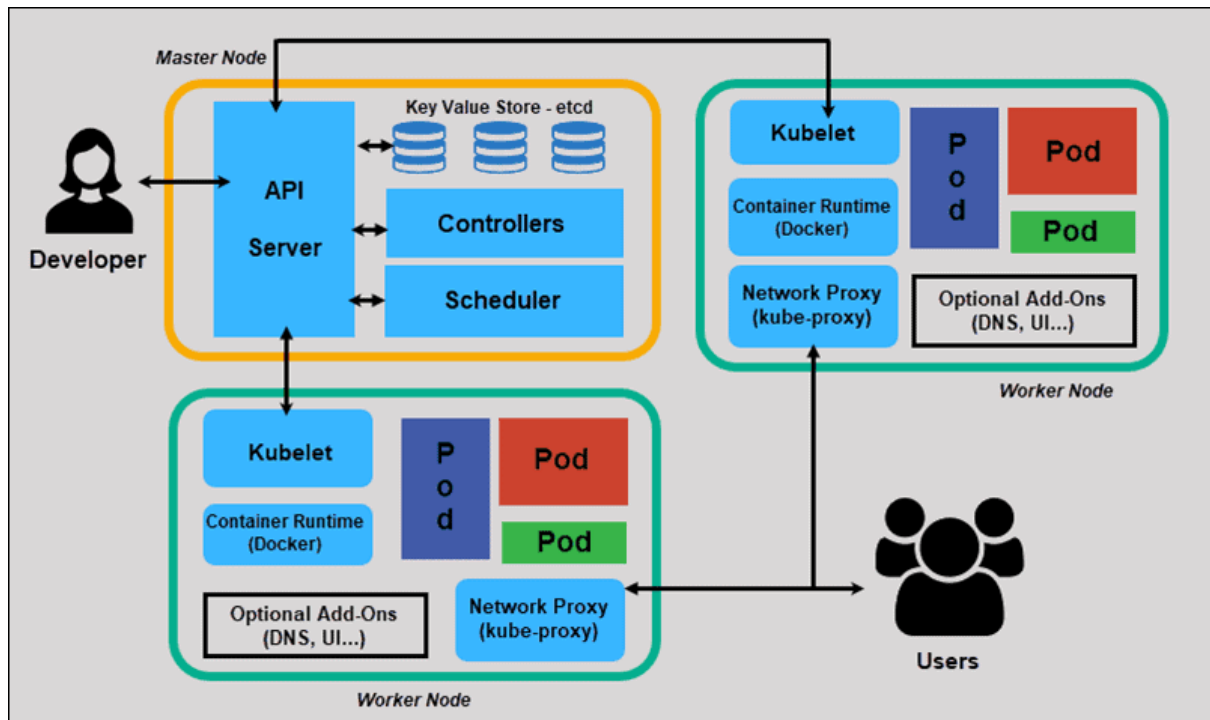
노드(=워커노드, 쿠버네티스 노드)

: 컨테이너화된 앱을 실행하는 시스템. 앱 실행 + 모니터링 + 앱에 서비스 제공

- Kubelet : API서버와 통신하고 노드의 컨테이너(그림에서 Pod안에 존재)를 관리
- Kube-Proxy : 애플리케이션 구성요소간 네트워크 트래픽 로드밸런싱 역할
- 컨테이너 런타임 : 컨테이너를 실행하는 .. 도커
- Pod : 한개 이상의 컨테이너가 그룹화된 세트

4. 쿠버네티스에서 애플리케이션 실행

디스크립션으로 컨테이너 실행하는 방법



(내 이해를 위해 그림 한번 더나옴)

Pod를 통한 격리성 관리

Pod로 묶인 컨테이너들은 함께 배치되고, 서로 격리해서는 안됨

- (그림엔 없지만) 앱 디스크립터에서 Pod 옆에는 병렬로 실행해야 하는 각 파드의 복제본 수 기록됨

API 서버가 애플리케이션 디스크립션 처리시

- 스케줄러 : 각 컨테이너에 필요한 리소스 계산
 - → 해당 시점에 각 노드에 할당되지 않은 리소스를 기반으로 사용 가능한 워커 노드에 지정된 컨테이너 할당
 - 각 Pod의 지정된 복제본 수를 사용가능한 워커노드로 할당
- Kubelet : 컨테이너 런타임(도커)에 필요한 컨테이너 이미지를 가져와서 컨테이너 실행 지시

실행된 컨테이너 유지

애플리케이션이 실행 후..

- 쿠버네티스: 애플리케이션의 배포 상태가 개발자가 제공한 디스크립션과 일치하는지 지속적으로 확인
 - 디스크립션에서 항상 5개 웹서버 인스턴스 실행하게 지정 → 쿠버네티스가 정확히 5개 인스턴스 실행
 - 장애시 쿠버네티스가 자동으로 다시 시작
 - 워커노드 전체가 종료되면 해당 워커노드의 모든 컨테이너 노드 새로 스케줄링

복제본 수 스케일링

애플리케이션이 실행되는 동안 복제본 수를 늘릴지 줄일지 결정 (← 쿠버네티스가 자동으로 조정하도록 설정 가능)

이동한 애플리케이션 접근

컨테이너가 클러스터 안에서 이동하는 경우

- 실행중인 노드가 정지됨 (장애상황)
- 다른 컨테이너를 위한 공간을 만드려고 노드에서 제거됨
- 컨테이너가 외부 클라이언트나 클러스터에서 실행 중인 다른 컨테이너에 서비스를 제공하는 경우

쿠버네티스 : 하나의 고정 IP 주소로 모든 컨테이너로 노출 → 동일한 서비스를 제공하는 컨테이너 주소를 클러스터에서 실행 중인 모든 애플리케이션에 노출

Kube-Proxy : 서비스를 제공하는 모든 컨테이너에서 서비스 연결이 로드밸런싱되도록 한다. 서비스의 IP 주소는 일정하게 유지됨 (이동 후 새 IP를 받지만, 서비스는 여전히 동일한 IP를 가진다)

→ 컨테이너가 클러스터 내에서 이동하더라도 컨테이너에 항상 연결 가능

5. 쿠버네티스 사용의 장점

- 운영자가 더이상 앱 배포 처리할 필요 없음 → 앱 배포의 단순화 (모든 워커 노드를 하나의 배포 플랫폼으로 제공)
- 하드웨어 리소스 활용도 높아짐 : 애플리케이션이 클러스터 간에 자유롭게 이동할 수 있으므로 클러스터에서 실행되는 다른 애플리케이션 구성 요소를 혼합해 클러스터 노드에

배치 가능

- 상태 확인과 자가 치유 : 앱의 구동 상태를 모니터링하다가 특정 노드 장애 발생시 자동으로 다른 노드로 리스케줄
- 오토 스케일링 : 쿠버네티스가 리소스를 모니터링하고 알아서 인스턴스 수 조정하게끔 설정할 수 있음
- 앱 개발 단순화