



쿠버네티스 동작 원리

☰ 태그

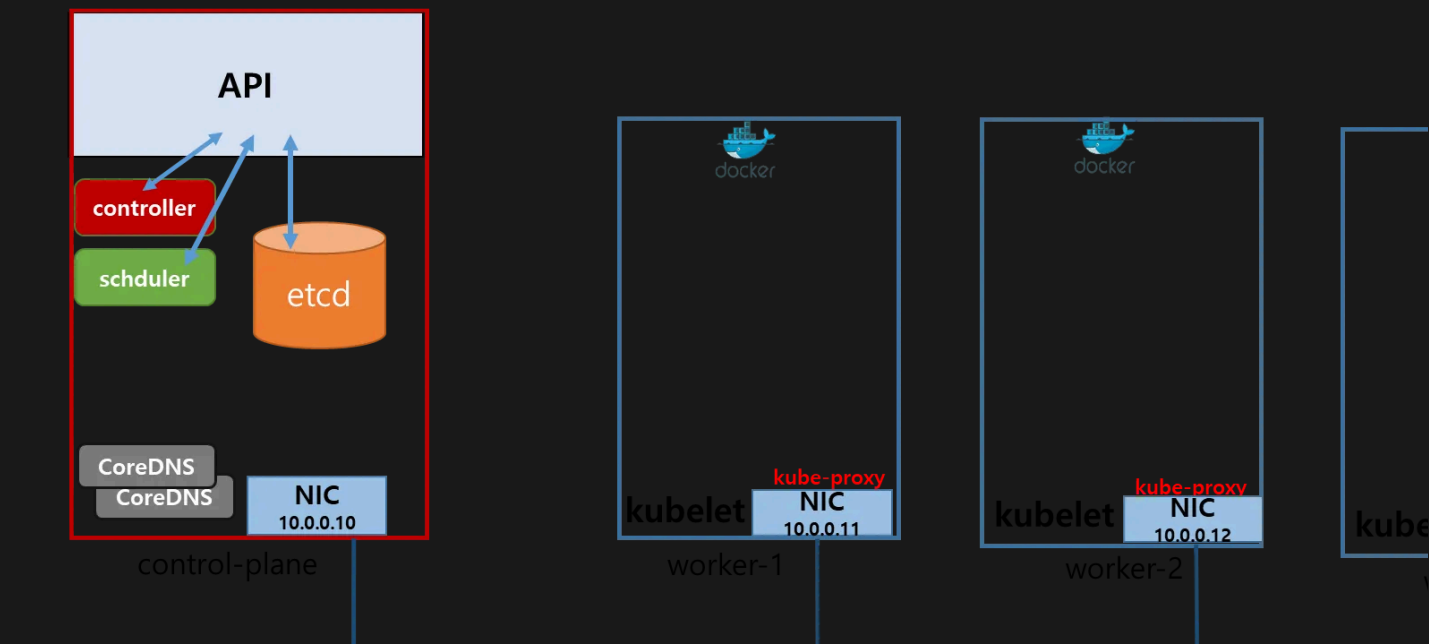
과정소개

1. Kubernetes Components

2. Kubernetes API 리소스 관계도

1. Kubernetes Components

- Kubernetes 인프라



- **Control plane components**

- Control plane 컴포넌트는 일반적으로 하나의 노드에서 동작하지만,고가용성 환경이나 대규모 클러스터에서는 여러 노드에 분산된 형태도 가능하다.
- **API Server** : 쿠버네티스 REST API를 공개한다. 모든 데이터를 etcd 클러스터에 저장하므로 쉽게 수평 확장이 가능하다. API 서버는 쿠버네티스 control-plane 을 구체화 한 것이다.
- **etcd** : 분산데이터 저장소. 쿠버네티스는 이것을 사용해 전체 클러스터의 상태를 저장한다. 규모가 작고 일시적인 클러스터의 경우에는 etcd 단일 인스턴스가 다른 마스터 컴포넌트와 함께 동일한 노드에서 동작할 수 있다. 하지만 대규모 클러스터의 경우 일반적으로 이중화와 고가용성을 위해 3개에서 많으면 5개 노드의 etcd 클러스터를 갖기도 한다.
- **controller manager** : 컨트롤러 매니저는 API를 사용해 클러스터의 상태를 감시하고, 클러스터를 원하는 상태로 조정한다. 컨트롤러 매니저에는 replication controller, pod controller, service controller, endpoint controller 등이 포함된다.
- **scheduler** : Kube scheduler는 node에 pod를 스케줄링 하는 역할을 담당한다. 서비스의 리소스 요구사항, 서비스 요구사항, hardware와 software 정책 제약 사항 등 여러 가지 상호작용을 요인으로 고려해야 하므로 사실상 매우 복잡한 작업이다.
- **Add-On Component**
 - coreDNS(Kubernetes DNS Server) : 쿠버네티스 1.3버전부터 core DNS 서비스를 표준클러스터에 포함하였고, 일반적인 Pod로 스케줄링 된다. headless 서비스를 제외한 모든 서비스는 DNS 이름을 가지며 Pod 역시 DNS 이름을 가진다. 이것은 자동검색(automatic discovery)에 매우 유용하다.
 - CNI(Container Network Interface Plugin)
 - Ingress
 - Dashboard

- **Worker Node components**

- 클러스터의 Worker Node가 클러스터 Master components와 상호작용하고, 클러스터를 실행하고 업데이트하기 위한 워크로드를 받기 위해서는 몇 가지 관련 components가 필요하다.
- **kubeproxy** : 각 node에서 저 수준의 네트워크 관리 업무를 수행한다. 쿠버네티스 서비스를 지역적으로 반영하고 TCP와 UDP 포워딩을 수행하며 환경 변수나 DNS를 통해 클러스터 IP를 찾는다.
- **kubelet**
 - Kubelet은 쿠버네티스를 대표하는 Node라 할 수 있다.
 - Kubelet은 마스터 컴포넌트와 통신을 수행하며 실행 중인 포드를 관리하고 감독한다.
 - API 서버에서 Pod Secret 다운로드, 볼륨 마운트, 포드의 컨테이너 실행, Node와 각 Pod의 상태 보고, 실행 중인 컨테이너의 활성 여부 조사

2. Kubernetes API 리소스 관계도

- CKA 시험의 주요 목적은 Kubernetes 기본 요소에 대한 실제 지식을 인증하는 것이다.

- 아래 그림을 통해 Kubernetes 리소스의 관계를 이해하고, 하나의 문제에서 여러 개념을 결합해서 생각할 수 있어야 한다.

