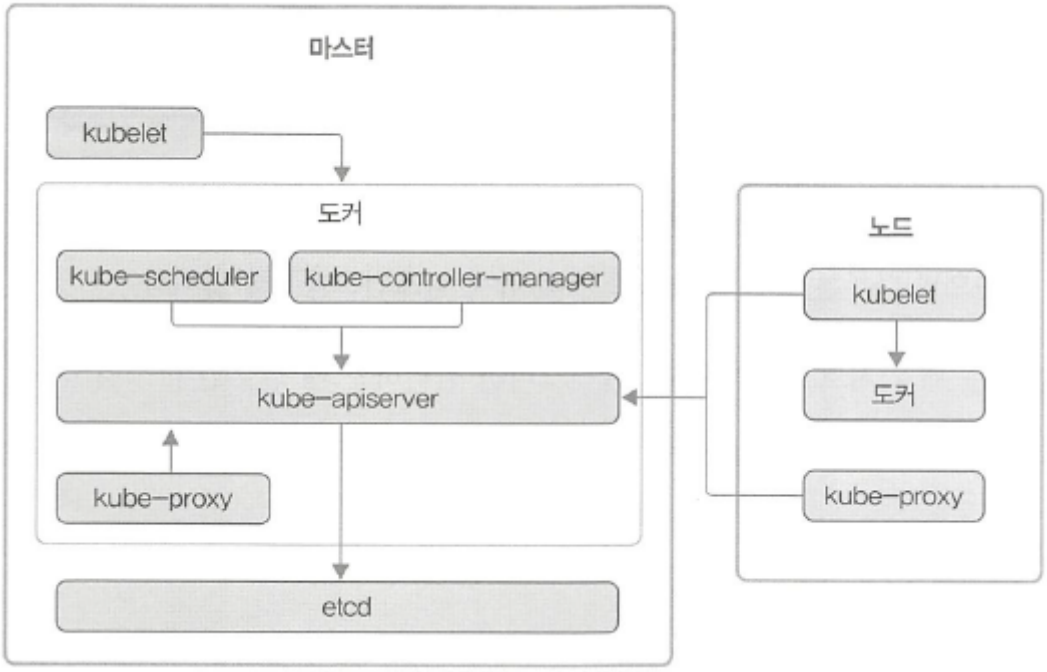
k8s - 쿠버네티스 아키텍처 (3)

클러스터와 마스터 노드의 구조를 알아보자

클러스터의 전체 구조도

그림 4-2 마스터와 노드의 구성과 통신 구조



kube-apiserver : 쿠버네티스의 모든 정보 통신, (etcd 접근 유일하게 가능)

kubelet : 노드 내 도커 관리

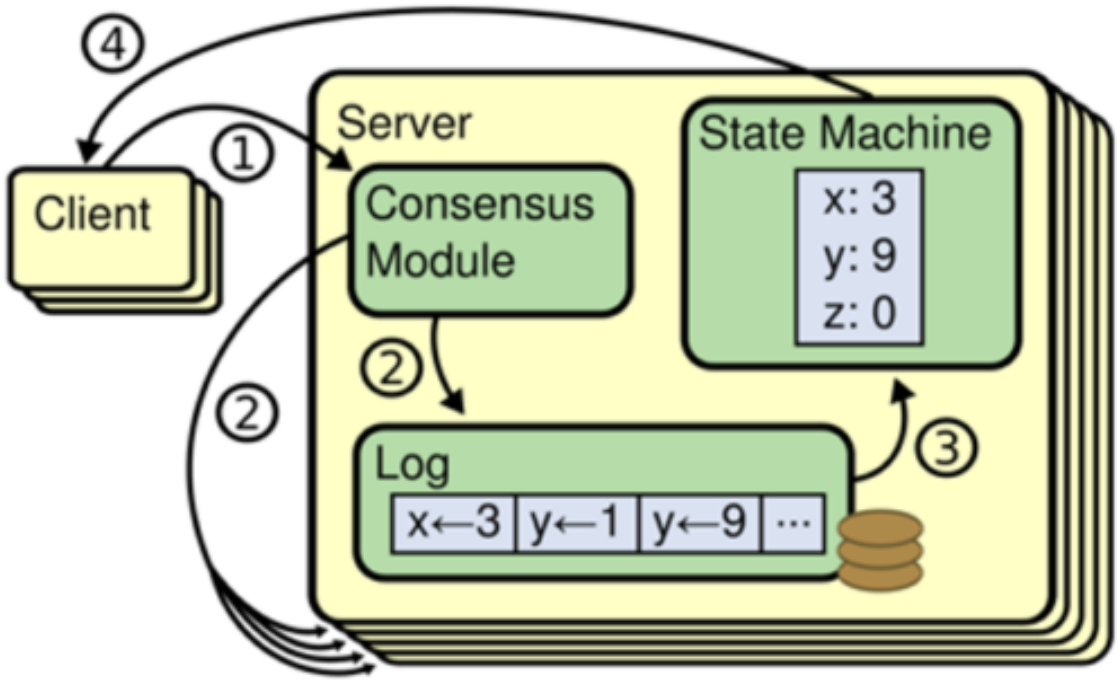
마스터용 컴포넌트

etcd

+ *추가참조* : https://tech.kakao.com/2021/12/20/kubernetes-etcd/

key:value 형태의 값 저장소 (컨테이너 x, 별도의 프로세스)

상태 관리를 위해 상태 복제 데이터를 여러 서버에 지속적으로 복제 한다고 Replicated state machine



Consensus를 확보하기 위해 Raft알고리즘을 활용.

의사 결정을 위한 최소 서버(Quorum) 갯수 만큼 로그 데이터를 복제.

전체 서버의 로그를 저장해 분산 컴퓨팅의 지속성과 안정성을 유지.

→ 지속적으로 백업 필요

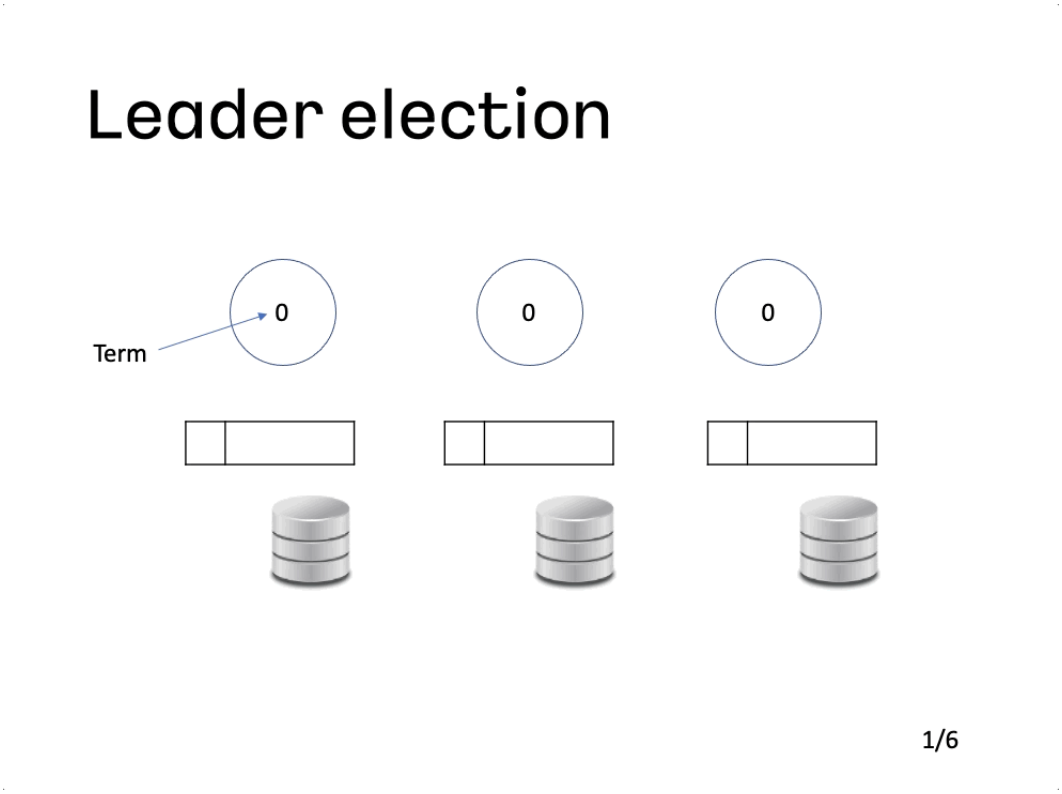
Raft 알고리즘 (뗏목 합의 알고리즘)

임시 마스터 노드 중, 실제 활용되는 마스터 노드 1개를 선출 하는 방법 알고리즘

1. 리더는 각 서버에 hearbeat를 각 시간마다 보낸 이후 생존을 알립니다.
리더 선출 시 서버간 term이라는 의사 결정권 count를 1 증가시킵니다.
2. 유저에게 wirte요청을 보내면 해당 로그를 Quorum만큼 로그를 공유합니다.

2-2. 만약, leader가 다운되면 hearbeat를 넘기지 못하고, timeout 시,
서버끼리의 인덱스를 비교해 가장 최신의 로그의 인덱스(lastIndex)를 가지고 있는 서버에게 리더를 임시 위임합니다. term을 1 증가시킵니다.

2-3. 이전의 리더가 복구 되면 다시 자신이 리더인줄 아는 구 리더가 hearbeat를 보냅니다.
이후, 자신의 term이 다른 서버보다 뒷처진 것을 알고 자신을 follower로 변경합니다.



kube-apiserver

+ 추가참조 :

- https://coffeewhale.com/apiserver
- https://www.sysnet.pe.kr/2/0/12566

클러스터 요청이 유요한지 검증

복잡한 컴포넌트가 아닌, 쿠버네티스가 만든 단순 rest api서버

각 컴포넌트에 맞는 api에서 권한 확인 후 목록 조회 return

수평적으로 여러 서버에 확장 가능

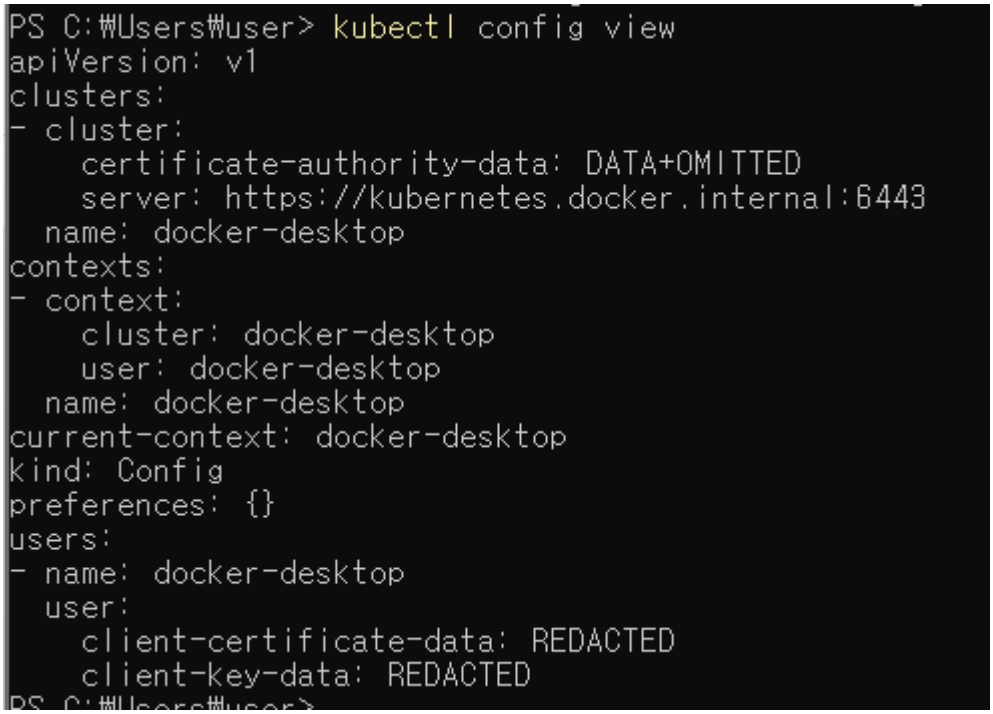
API 서버 정보 확인

```
kubectl cluster-info
kubectl config view
```

```
PS C:₩Users₩user> kubectl cluster-info
Kubernetes control plane is running at https://kubernetes.docker.internal:6443
CoreDNS is running at https://kubernetes.docker.internal:6443/api/v1/namespaces/kube-system/services/kube-dns:dns/proxy
```

API 서버의 주소는 kubeconfig 파일 내부에 있습니다.

- `clusters[0].cluster.server` 로 접근 가능



인증 키를 확인하는 방법 → JWT Bearer 토큰으로 인증 시 필요

```
kubectl config view --minify --raw --output 'jsonpath={..cluster.certificate-authority
-data}'
```

우리는 간단히 kubectl에서 프록시를 띄워줍시다. 왜냐면 저는 시크릿이 없거든요

```
kubectl get serviceaccount default
```

```
PS C:₩Users₩user> kubectl get serviceaccount default
NAME      SECRETS   AGE
default   0         21d
```

```
# hosts내부 docker local dns 내부 api server접속 시 ssl 오류
curl https://kubernetes.docker.internal:6443/api/

# 프록시가 없어 커넥션 에러
$ curl http://127.0.0.1:8001/api/

#프록시 띄운 뒤
kubectl proxy
$ curl http://127.0.0.1:8001/api/
```

http://127.0.0.1:8001/ 접속 이후

/openapi/v2에 나온 json 형태의 파일을 스웨거로 변경 시 swagger문서로도 확인 가능합니다

https://editor.swagger.io/

🔥 아래부터는 좀더 공부해서 공유드리겠습니다.

kube-scheduler

https://m.blog.naver.com/alice_k106/221511412970

현재 클러스터 내부에서 자원 할당이 가능한 노드 중 알맞은 노드를 선택 해 새로운 파드 실행

etcd

etcd