

1일차: 감잡기

Kubernetes for 금융결제원

- [1~2교시] 쿠버네티스란 무엇일까?
 - 쿠버네티스를 왜 쓰는것일까?
 - 쿠버네티스 이전의 세상
- [3~5교시] 쿠버네티스는 어떻게 쓰는것일까?
 - 접속방법
 - Pod 란?
 - ReplicaSet? Deployment?
 - Service 란?
 - Ingress 란?
- [6교시] 쿠버네티스를 어떻게들 쓰고 있을까?

쿠버네티스란 무엇일까?

Q. 쿠버네티스를 한문장으로 정의해보세요

Q. 쿠버네티스를 한문장으로 정의해보세요

(컨테이너 오케스트레이터 빼고)

A. 리소스 매니저

리소스 = 텍스트로 표현 가능한 모든것

A. 리소스 매니저

리소스 = 텍스트로 표현 가능한 모든것

리소스 매니저?

앞으로 자주 언급하겠지만...

- Kubernetes 는 기본적으로
yaml 이라고 불리는 포맷으로 리소스를 선언할 수 있다.
- 어떤 앱을 인터넷을 통해 접근가능하게 하기까지
단 3개의 리소스만 선언하면 나머지는 Kubernetes 가 알아서 해줌
 - Deployment
 - Service
 - Ingress

Kubernetes 특징

1. 모든것을 리소스로 관리한다.
2. 리소스를 선언한 하면 나머지는 알아서 작동한다.
(선언형 API 라고도 표현)

Kubernetes 이전까지의 API - 업무에 비유

일일이 누가 뭘 어떻게 해야하는지 관리해야한다.

Before: 카카오톡으로 업무지시

카카오페이에 API 하나 열어줘야하는데

- A님 개발은 어떻게하고
- B님 방화벽 신청은 이거 어디가서 어떻게 열어주고
- C님 누구랑 협조요청하고
- D님 ... (이하생략)

누가 무엇을 어떻게 수행 지시

Kubernetes 의 선언형 API - 업무에 비유

리소스를 선언하면 나머지는 알아서 잘 한다.

After: 프로젝트 세부 스펙 공지

카카오페이에 API 하나 열어줘야하는데

- 어떤 기능의 API에 대해서
- TCP 몇번 포트 쓰고
- 누구를 통해서 뭘 하고
- (이하생략)

=> 이메일로 PDF 파일 전 직원에게 공유

=> 나머지는 직원들이 알아서 역할분담해서 잘 수행

Kubernetes 이전까지의 API

Before: SSH 로 접속해서 일일이 수행

- x.x.x.x 서버는 마스터노드 담당하고
- x.x.x.x 서버는 워커노드 담당하고
- x.x.x.x 서버는 Bastion 으로 쓰고
- x.x.x.x 서버는 Audit 로그 쌓아두고
(등등)

Kubernetes 의 선언형 API

After: 그냥 뭐가 필요한지 작성

- 마스터노드는 2대 필요하고
- 워커노드는 200대 필요하고
- Bastion 서버는 2대 필요하고
- Audit 로그는 어디다 쌓고

- 서버 100대 있으니깐 알아서 써, 부족하면 말하고

(등등)

쿠버네티스의 핵심은

1. 리소스를 선언만 하면 나머지는 알아서 관리하는 하나의 플랫폼이라는 것
2. 그 리소스의 종류는 사용자 마음대로 확장이 가능하다는 것
3. 단지 이런식으로 설명하면 말이 많아지고 복잡해지고 쉽게 납득이 안가니깐
컨테이너 오케스트레이션 을 대표 기능으로 표방하는 것 일뿐...

Q. 그래서 왜 이것에 다들 환호하는 것일까요?

A. 쿠버네티스 이전의 역사

<https://drive.google.com/file/d/1UWc58RmA7oU4j3OfhAJSIjAo0IoMhiWC/view?usp=sharing>

Q. 그래서 왜 이것에 다들 환호하는 것일까요?

A. 그래서 왜 이것에 다들 환호하는 것일까요?

- 0. 기존것들과 비교했을때
=> SSH, Ansible, Puppet, OpenStack, AWS EC2...
- 1. 일관된 작동을 보장하니깐
=> Docker
- 2. 클러스터를 관리하는데 편하니깐
=> Mesos (DCOS), Docker Swarm
- 3. 요구사항에 맞게 확장까지 되니깐 (Database, Storage, Network, etc...)
=> Kubernetes
- 4. 남들이 만들어둔 확장 로직을 가져와서 쓸 수 있으니깐 (CI/CD, Logs, Metrics, etc...)
=> Kubernetes
- 5. 심지어 생태계도 크니깐 (CNCF)
=> Kubernetes