

On-Premise CI/CD 개발환경 구성

Ver.1.0.0

2021.11.29

IT팀 선임연구원
이탁곤

목차

구성목록

전체 구성도

CI/CD파이프라인 구성

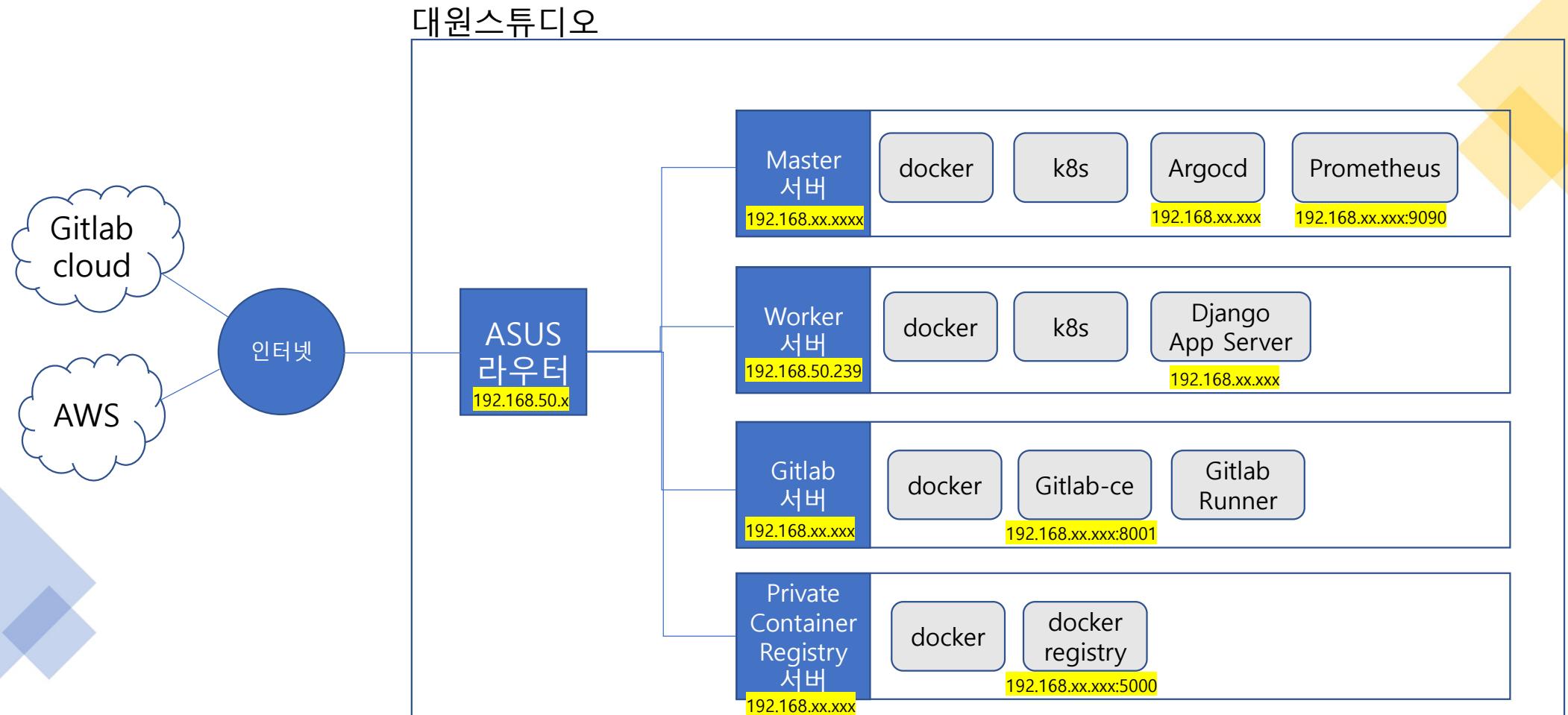
테스트 시나리오

상세내역

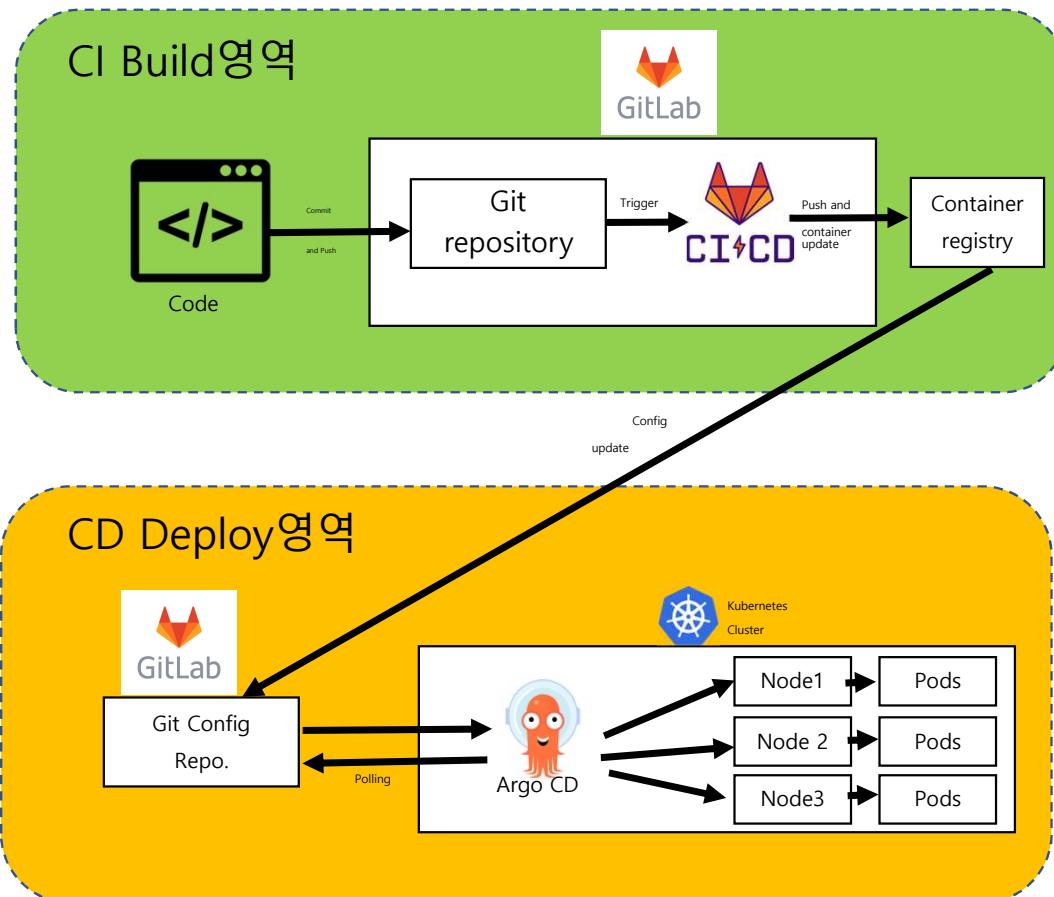
구성목록

대분류	소분류	name	version	설명
OS	Linux	ubuntu	20.04	Ubuntu 20.04 버전
k8s	orchestration	kubernetes	1.22	HA 및 Loadbalance
		calico	3.17	Overlay 네트워크. k8s 클러스터링 네트워크
	Load-balancer	metallb	0.9.5	on-premise 환경에서 서비스 외부 IP 노출
	package manager	helm	3.7.0	k8s cluster 패키지 관리
	Auto deploy	argocd	2.1.5	자동 지속적 배포
	kube-prometheus	Prometheus	20.0.1	모니터링 서버
		Grafana	6.17.5	모니터링 시각화
	Kubernetes IDE	Lens	5.2.7	k8s UI manage tool
	docker	docker	20.10	docker container
		container registry	2.7.0	docker container 저장소
SVC	Git	gitlab-ce	14.4.2	issue manage, source version control, Continuous Integration
application-service server	WAS	django	3.1.3	django framework 서버
	WSGI	gunicorn	20.1.0	UNIX용 순수 Python WSGI 서버
	WEB	nginx	1.20	웹서버
REST-API	schema	drf_yasg-1.20	1.20	Django REST-Framework 자동 문서화 openapi
Language	백엔드	python	3.9.8	파이썬 버전 3.9
	프론트엔드	VUE	2.6.10	

전체 구성도



CI/CD 파이프라인 구성(GitOps방식)



사전작업

- Git 저장소: 테스트 및 도커 컨테이너 빌드, 배포에 대한 스크립트 작성(.gitlab-ci.yml)
- Git config 저장소: 상태 업데이트 및 배포를 위한 helm chart작성(chart.yaml)

Commit and Push

- 개발자가 Code commit 후 push

컨테이너 빌드

- Git 저장소의 .gitlab-ci.yml의 순서에 따라 도커 컨테이너 빌드 진행

컨테이너 레지스트리

- 빌드된 컨테이너 이미지는 container registry에 push됨

Git config 저장소 업데이트

- Git config 저장소에 버전 업데이트

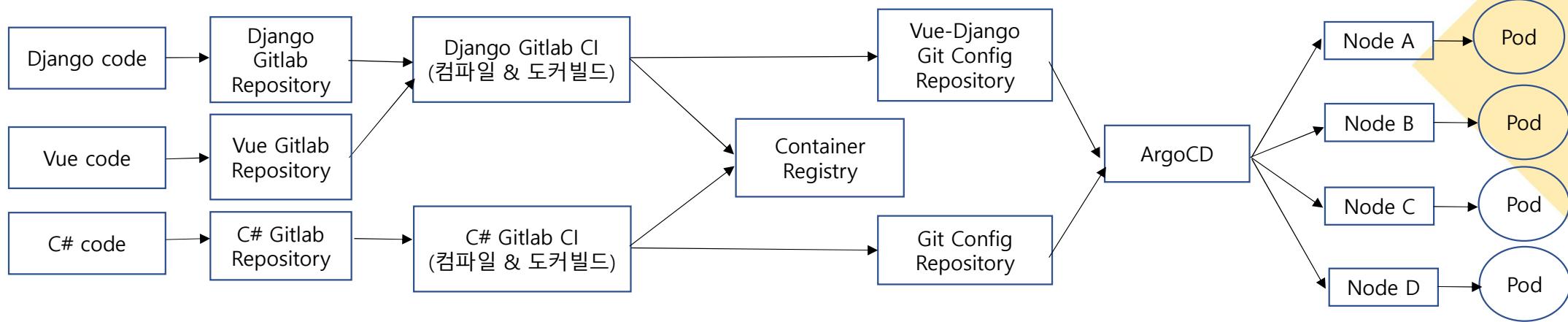
Argocd Polling

- Argocd(Kubernetes를 위한 선언적인 gitops 연속전달 도구)가 3분마다 git config 저장소의 배포설정을 polling(Option: webhook설정시, git config 저장소에 상태 업데이트 시 즉시 Argocd로 push)

ArgoCD 자동배포

- Kubernetes cluster내 Argocd가 수신하여 worker node에 pod를 생성하여 docker image배포

테스트 시나리오



1. Vue-Django 컨테이너 빌드 후, Node A에 배포
2. Vue-Django 컨테이너 빌드 후, Node B에 배포
3. Vue-Django 컨테이너 빌드 후, Node C에 배포
4. Vue-Django 컨테이너 빌드 후, Node A~B에 배포
5. Vue-Django 컨테이너 빌드 후, Node A~C에 배포
6. (옵션) C# 컨테이너 빌드 후, Node D에 배포

상세내역

- OS
 - 1. 모든 OS(docker build환경 포함)는 ubuntu 20.04로 통일
- K8s
 - 1. Ubuntu NTP설정 필수: Master서버와 Worker서버간 시간이 맞아야 health check 가능
 - 2. 외부에 서비스 노출 필요 시, MetalLB 사용하여 loadbalancer타입으로 노출
 - 3. K8s 모니터링 시, grafana 사용
- Private container registry
 - 1. registry 버전은 2.7.0
 - 2. registry list를 쉽게 보기 위해서 registry web 설치
 - 3. Push된 Docker image 삭제시엔 수동 삭제
 - 4. Push, pull을 위해 사설인증서 생성

상세내역

- Argocd
 - 1. Master서버에선 관리, 실제 파일은 Worker서버 pod에 배포되 있음
 - 2. 배포시 해당 worker 서버에 Private container registry 사설 인증서 추가
- Gitlab-ce
 - 1. repository 추가 후 CI 사용시, gitlab runner 설치
 - 2. 서버 접속 위해서는 repository에 Private container registry 사설 인증서 추가
- Django app server
 - 1. API서버 자동 문서화를 위해 swagger 사용