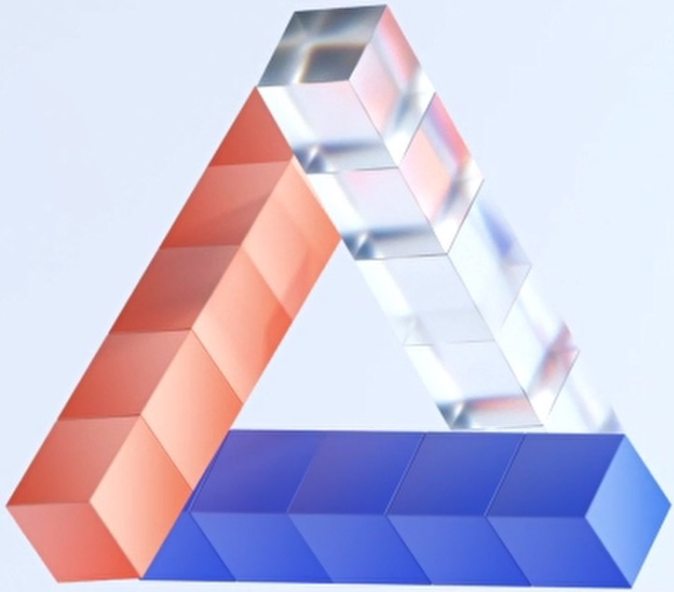




블록체인 서비스 인프라 현황

블록체인/인증플랫폼개발팀





1. Digital Asset CO의 플랫폼과 서비스 소개

2. Public Cloud에서 MSA와 DevOps



Blockchain = Decentralized Global Scale Platform

Bitcoin : Global Bank



Ethereum : Global Smart Contract Platform



NFT : Global Digital Assets Proof of Ownership



DID / VC(Verifiable Credential) : Global Identification

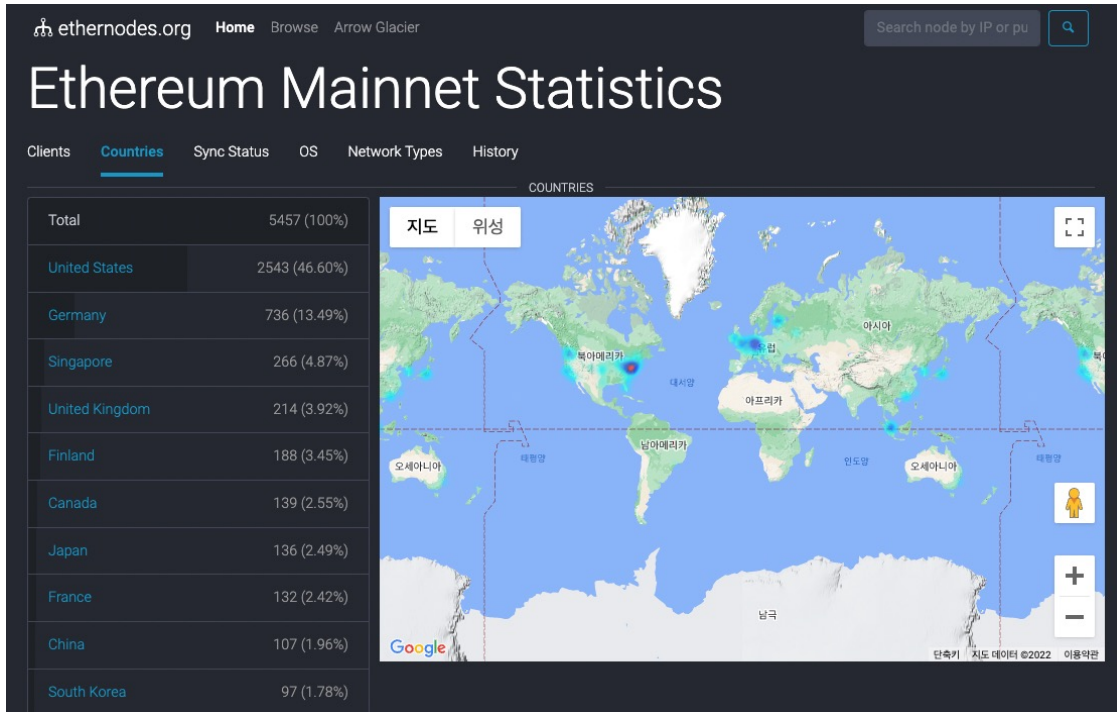


IPFS : Global Storage

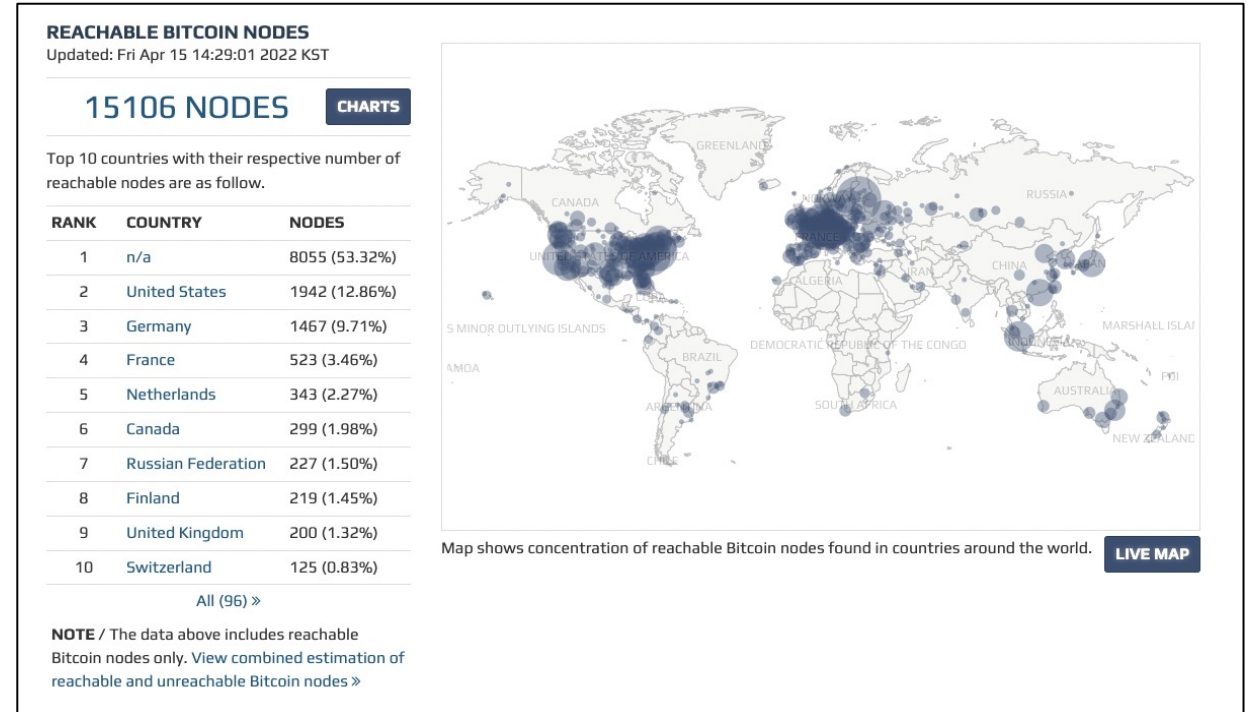
Why Global Scale Platform?



전세계 10000대 이상의 Node 참여 중인 단일 플랫폼



참고 : <https://ethernodes.org/countries>



참고 : <https://bitnodes.io/>



Vision & Objective

VISION

4차 산업시대의 핵심인
개인정보보호를 위한 기술
을 확보하고 WEB3.0 시대
를 위한 서비스 제공

Privacy

개인정보보호(공개키 암호화)

SSI

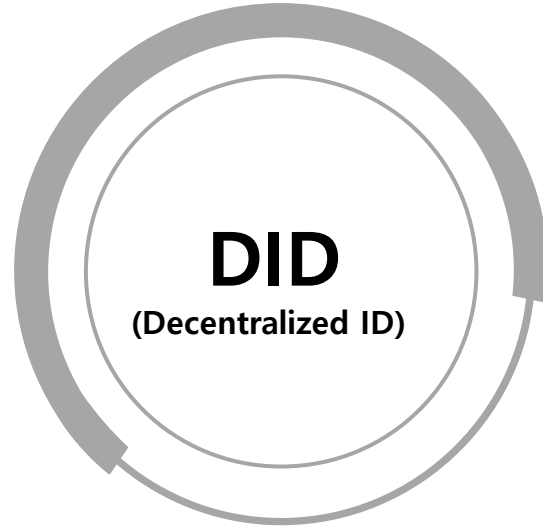
Self Sovereign Identity

Digital
Asset

Digital 자산의 안전한 보관



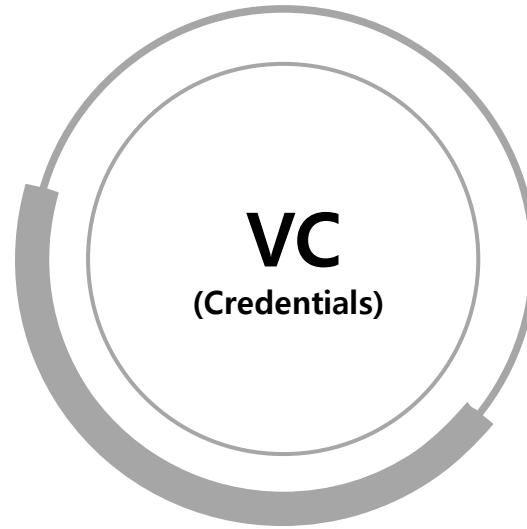
Blockchain Digital Assets



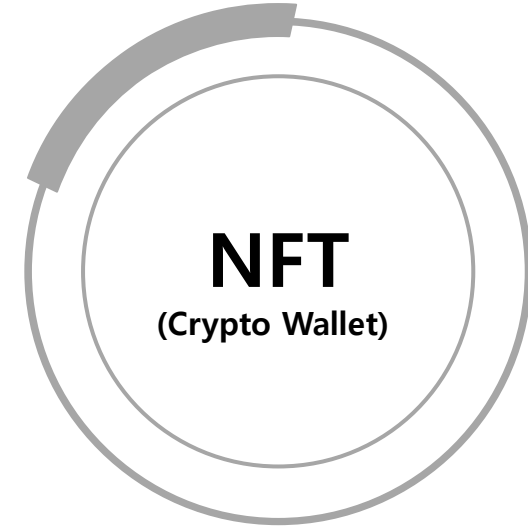
DID
(Decentralized ID)



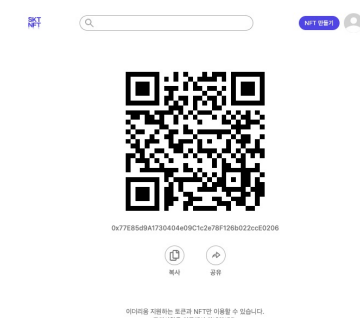
Scheme
did:example:123456789abcdefghi
DID Method DID Method-Specific Identifier



VC
(Credentials)



NFT
(Crypto Wallet)

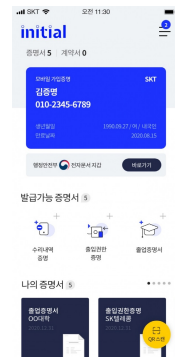
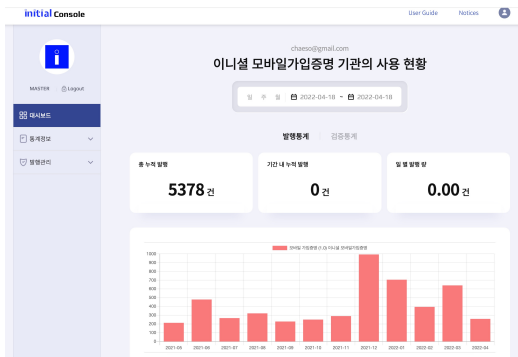




initial Console Platform

DID / VC 발급,검증,보관

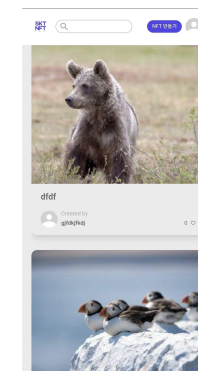
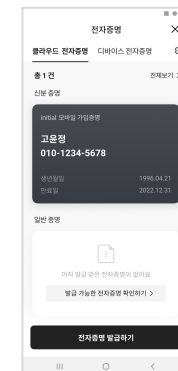
- 발급/검증 Web Console (기업용 VC 발급 시스템)
- 발급/검증 APIs
- initial Mobile App
- 150개 이상의 API 제공

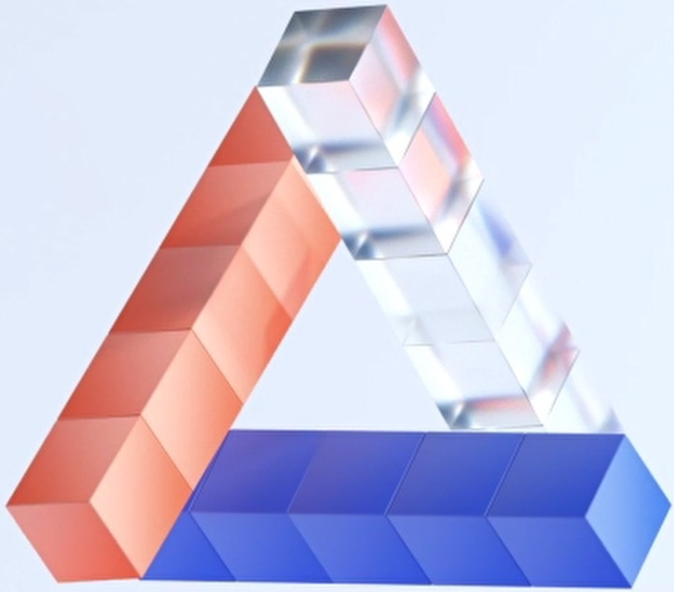


Mobile Wallet Platform

Digital Asset Wallet WebApp

- initial Cloud 개인 전자증명 지갑
- 행정안전부 전자문서 지갑
- 디지털 NFT 자산 지갑
- 전자계약 시스템
- 200개 이상의 API 제공





1. Digital Asset CO의 플랫폼과 서비스 소개
- 2. Public Cloud에서 MSA와 DevOps**



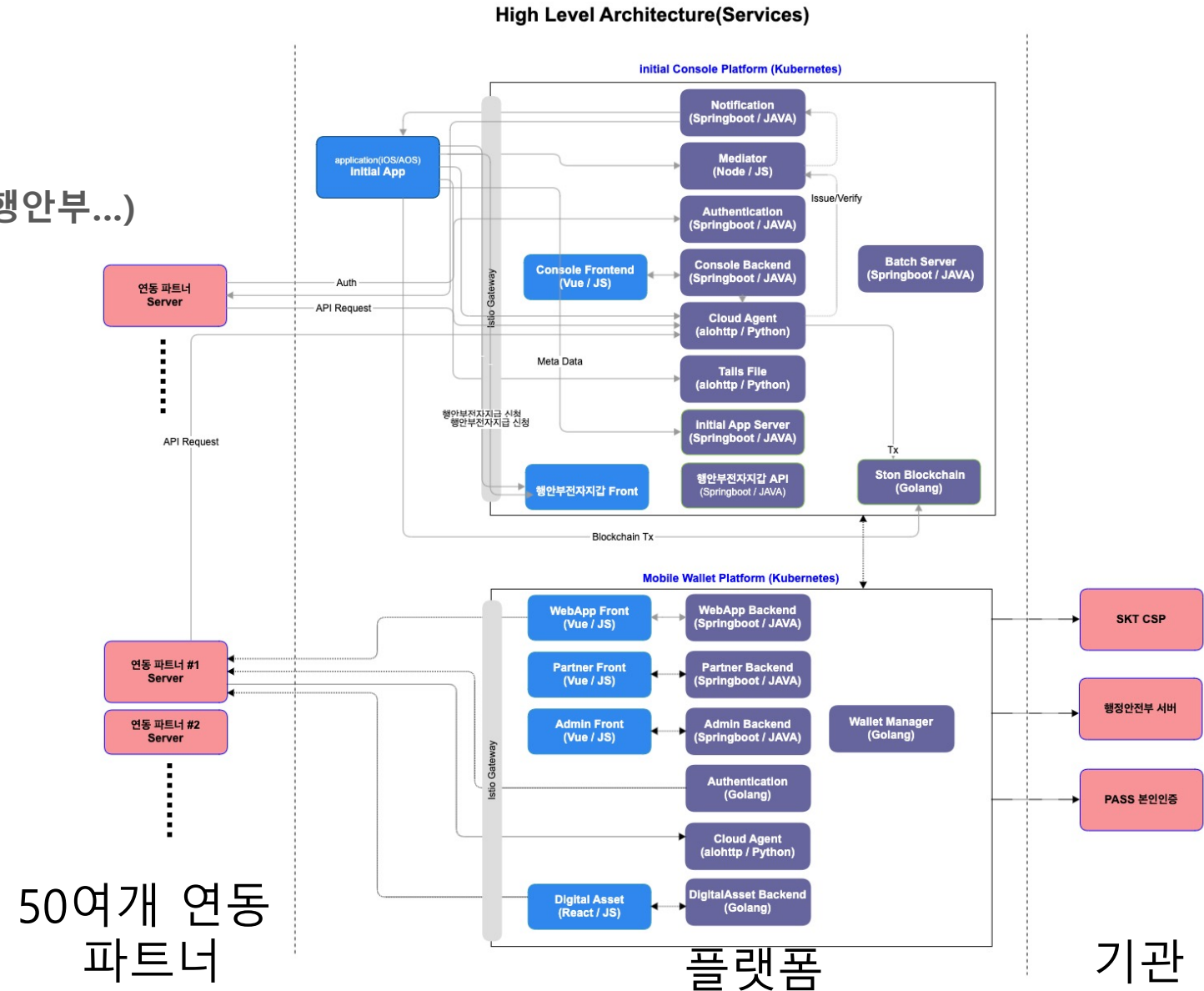


다양한 주체들과 연동

- 단순 Request/Response 형태가 아닌

파트너(삼성전자/LG) – 플랫폼 – 기관/국가망(CSP, 행안부...)

연동된 복잡한 형태





블록체인 서비스는 각 Layer에서 저장할 데이터를 아래와 같이 규정

- Blockchain은 Public Node 개념으로 모두 접근할 수 있기 때문에 민감하지 않는 data를 기록
- Web3.0의 최적화된 서비스 가능

웹 3.0은 모든 자료와 정보를 분산화하여 현재 집중화된 권력과 데이터를 개인에게 되돌려주는 차세대 네트워크 구조를 의미하며, 블록체인 기반 기술을 중점으로 하여 말 그대로 탈중앙 웹을 구현하는 것을 목표

사용자 공간(Mobile, 사용자 Cloud Storage..)

Business Logic Server

Blockchain Node

- 개인키
- 민감 개인 정보
- 개인정보 (망분리등 보안 처리)
- 민감하지 않은 Data(공개키, Hash)
- 개인정보 추적이 불가능한 Data (익명기반 정보)



생각의 흐름

재미있고, Hip 하고, Agile하고, 새롭고, 효율적이고, 즐겁게 할 수 있는 개발

최소 예산

DevOps

Open Source

MSA

Public Cloud

내재화 개발

Global Support

Kubernetes

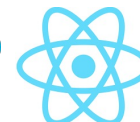
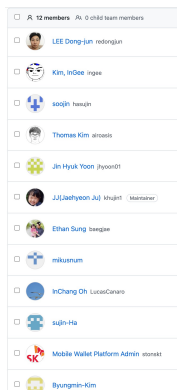
BP(Front/App)

자체 운영

GitHub

Automation

Multi Language



MSA(Micro Service Architecture)

Plan

Git

Build

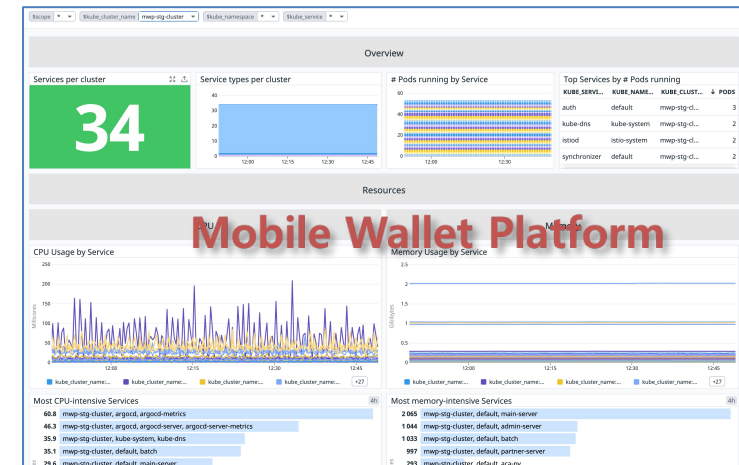
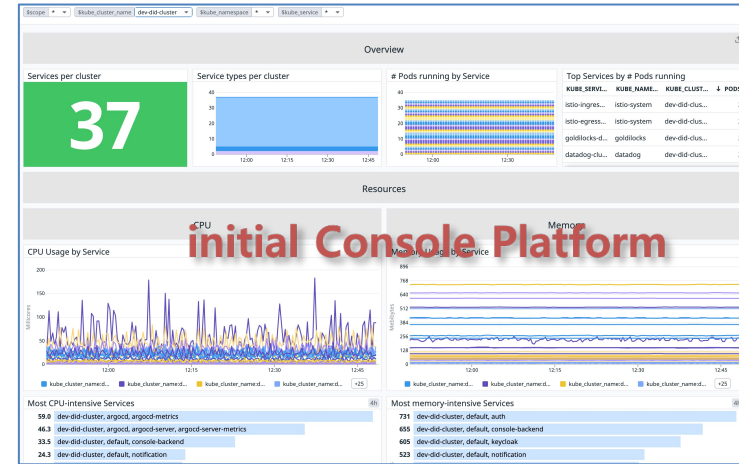
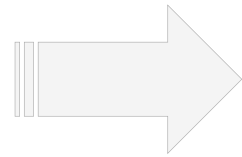
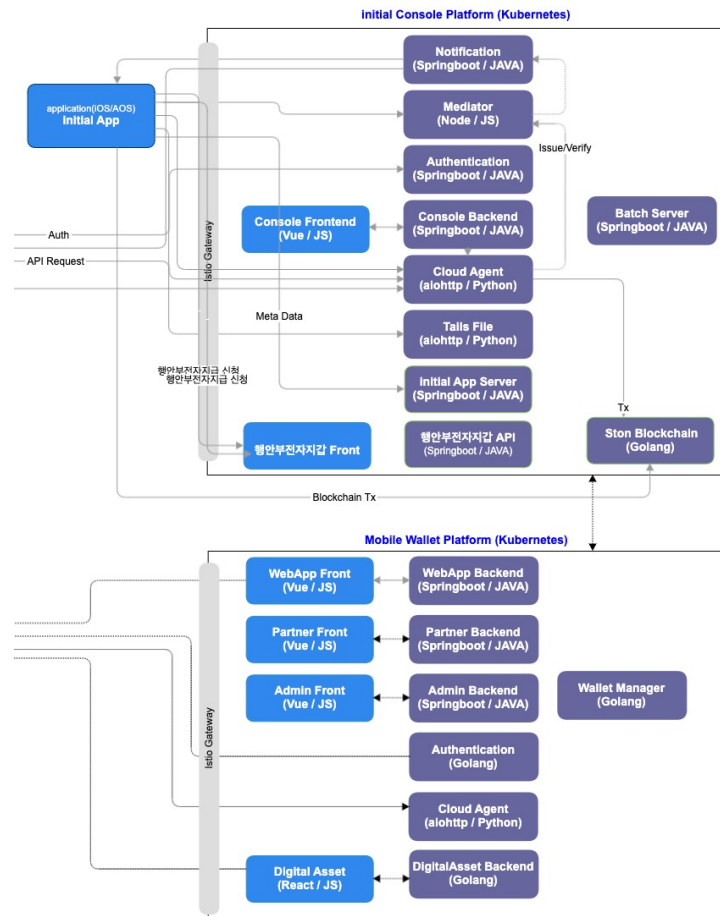
Config

Deploy



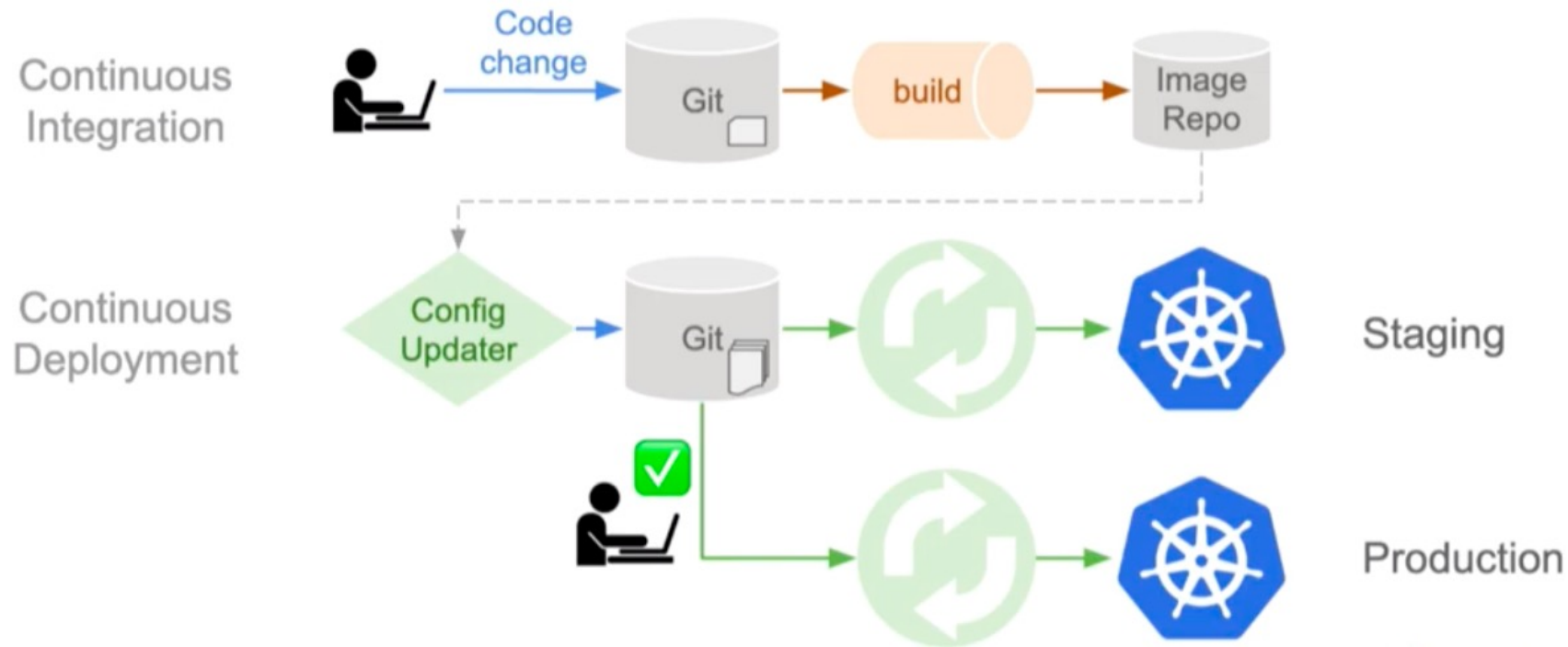
다양한 Language 및 Framework 사용으로 Monolithic Architecture 불가능
→ MSA 선택

High Level Architecture(Services)





Kubernetes에 application을 배포하기 위해 필요한 설정(deployment, service, persistent volume 등)을 git에 저장하고, 그 설정을 kubernetes 환경에 배포하고, 해당 상태를 유지하도록 지원하는 기술





이번 세미나에서는 각 기술의 상세한 내용을 다루지 않고, 저희의 경험을 공유합니다.
상세 기술 정보는 Kubernetes 공식 사이트의 Spec과
Devocean의 전문가들의 글을 참고하시면 됩니다.

<https://kubernetes.io/ko/>

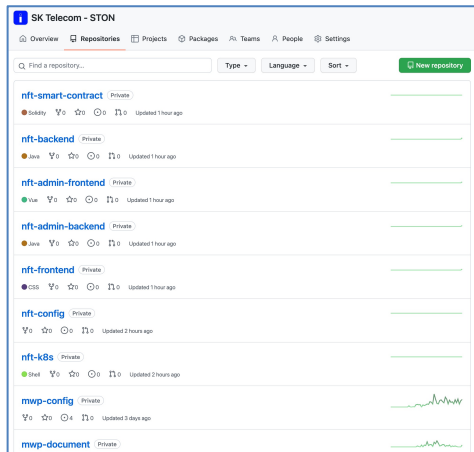
<https://devocean.sk.com/>



GitHub 기반 Build 환경 구축

- GitHub Actions 활용한 Build 및 Deploy

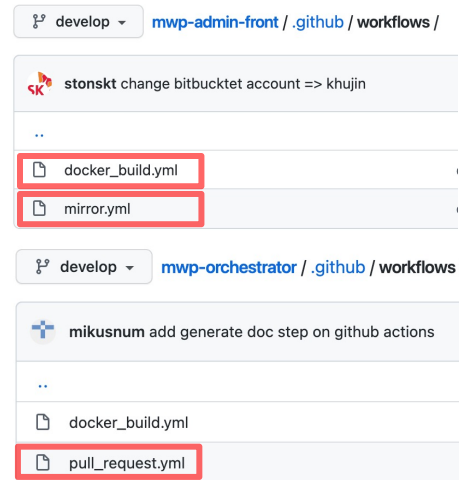
Source Code



GitHub

- Private Repository

Actions Workflow



build.yml

- Build 및 Docker image upload

mirror.yml

- SKT Bitbucket으로 Code backup

pull_request.yml

- PR시 lint + unit test 실행

Trigger (Push)

```
on:
  push:
    branches:
      - develop
    paths-ignore:
      - '.github/**'
  workflow_dispatch:
```

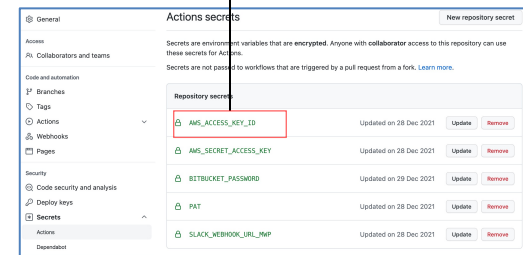
Build Trigger 조건

- 모든 push에 대해서 build
- develop branch만 사용

<https://docs.github.com/en/actions>

Build with Secret

```
54
55 update-configuration:
56   needs: build-and-push-image
57   runs-on: ubuntu-20.04
58   concurrency: customize
59
60   steps:
61   - name: Checkout mwp-config
62     uses: actions/checkout@v2
63     with:
64       repository: sktston/mwp-config
65       token: ${{ secrets.PAT }}
66   - name: Setup Kustomize
67     id: setup-kustomize
68     uses: imranismail/setup-kustomize@v1
69     with:
70       kustomize-version: "3.9.4"
71   - name: Update the image tag with Kustomize
72     id: update-config-with-kustomize
73     run: |
74       git config user.name github-actions
75       git config user.email github-actions@github.com
76       git checkout ${GITHUB_REF##*/}
77       git pull origin ${GITHUB_REF##*/}
78       cd overlays/${GITHUB_REF##*/}
79       kustomize edit set image ${{ secrets.SECR_REGISTRY }}/${{{ secrets.SECR_REPOSITORY }}}:${{{ secrets.SECR_IMAGE_TAG }}}
80       git add kustomization.yaml
```



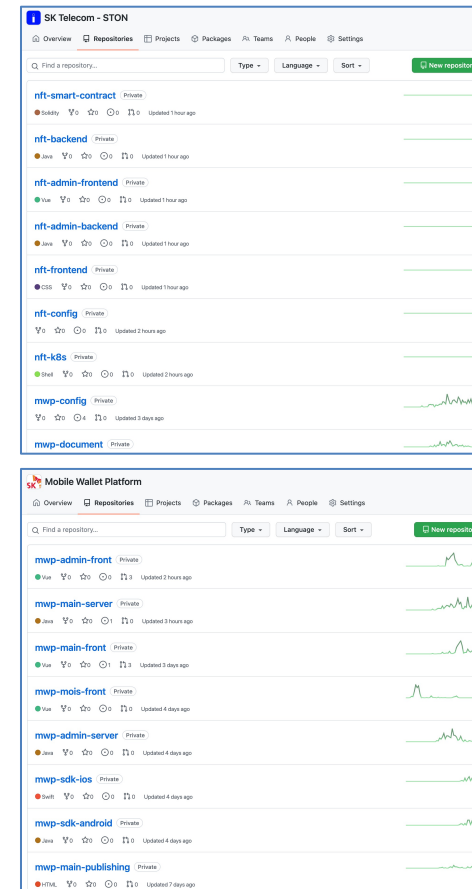
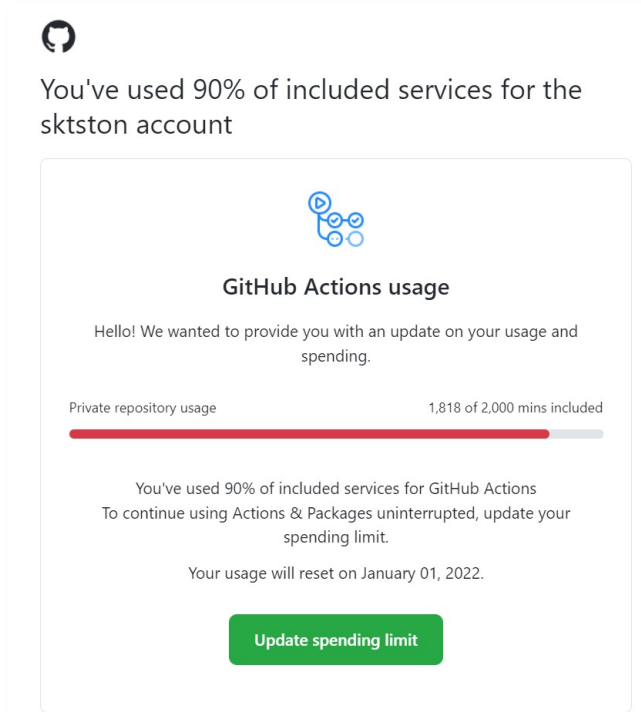
Actions Secrets

- Build에 필요한 Secret은 Key Store에 안전하게 관리



Organization 2개로 운영

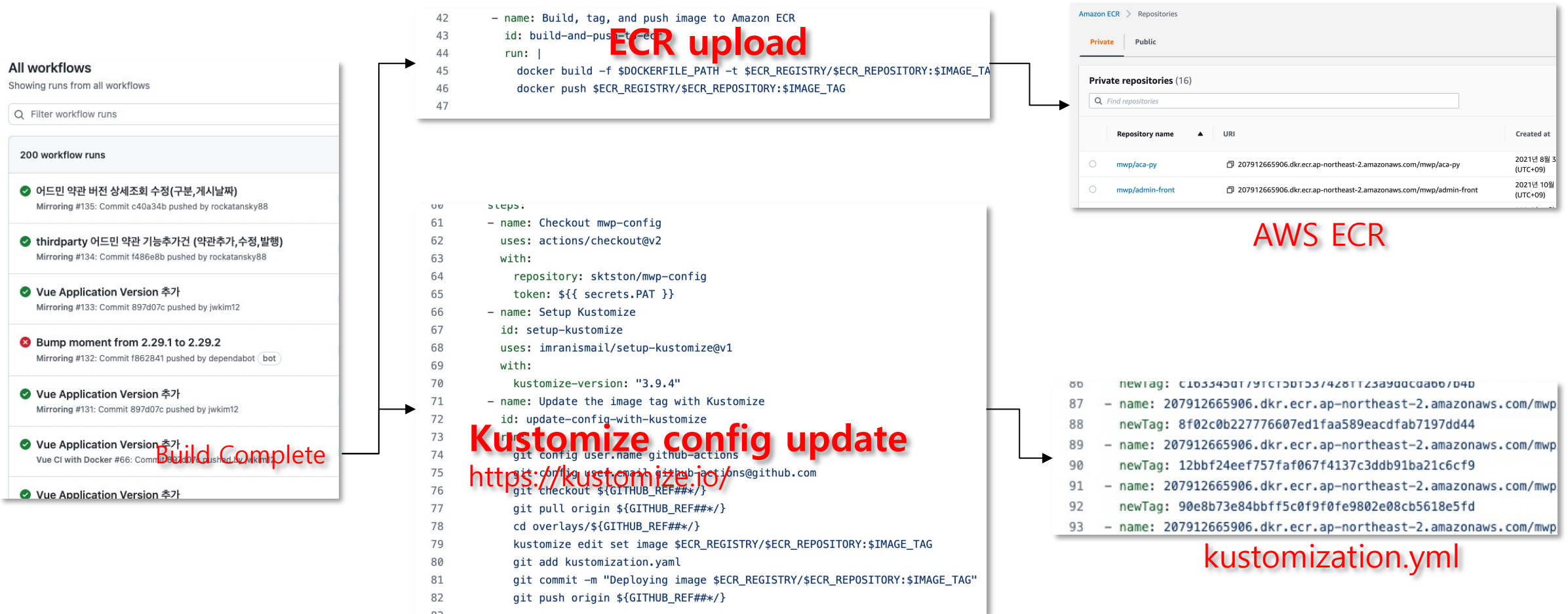
- GitHub Free Plan은 2000min/Month Build 가능
- iOS Build는 x10 build 시간 소모
- 개발 Schedule에 따라 월 2000min 초과 발생





Build 완료 후 AWS ECR로 image upload 하고 Tag 등록 → Kustomize e

- Config file에는 ECR에 update된 최신 image tag를 등록





ArgoCD Application에서 ECR image의 tag를 참조하여 배포

- 개발 환경별 image 배포 전략은 별도 수립

The screenshot displays the ArgoCD Application interface for 'mwp-dev'. The top navigation bar includes buttons for APP DETAILS, APP DIFF, SYNC, SYNC STATUS, HISTORY AND ROLLBACK, DELETE, and REFRESH. The application status is 'Healthy' and 'Synced'. The 'CURRENT SYNC STATUS' shows a successful sync to the 'develop' branch. The 'LAST SYNC RESULT' indicates a successful sync 3 days ago.

The 'Config' section is highlighted, showing a Config map with the following content:

```
86 newTag: C163343d1791C13D15374281123a9ddCda66/D4D
87 - name: 207912665906.dkr.ecr.ap-northeast-2.amazonaws.com/mwp/cloudagent-proxy
88   newTag: 8f02c0b22776607ed1faa589eacdfab7197dd44
89 - name: 207912665906.dkr.ecr.ap-northeast-2.amazonaws.com/mwp/cloudagent-datastore
90   newTag: 12bbf24eef757faf067f4137c3ddb91ba21c6cf9
91 - name: 207912665906.dkr.ecr.ap-northeast-2.amazonaws.com/mwp/main-server
92   newTag: 90e8b73e84bbff5c0f9f0fe9802e08cb5618e5fd
93 - name: 207912665906.dkr.ecr.ap-northeast-2.amazonaws.com/mwp/main-server
```

The 'Config' map is linked to various services in the application, including 'cloudagent-proxy', 'cloudagent-datastore', 'main-front', 'main-server', 'mois-front', 'mwp-asset-admin', and 'nydus'. Each service is shown with its deployment status and a link to the underlying pods.



Config의 overlays 세팅에 따라 동일한 image를 환경별 배포

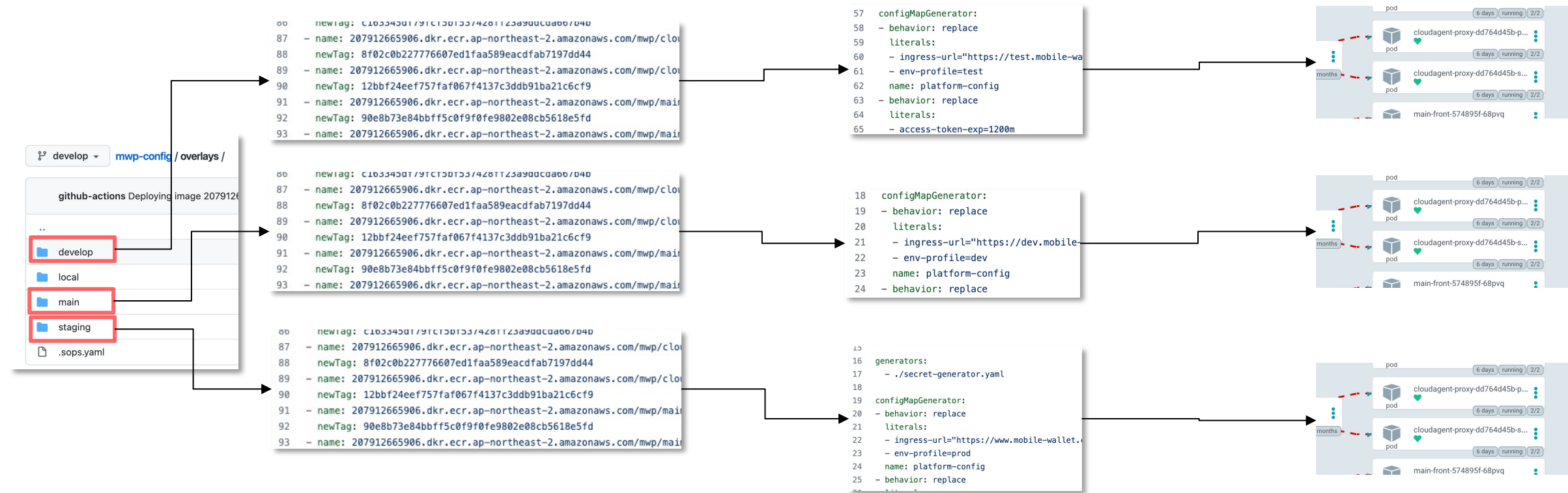


Image tag 확인

- Config/overlays/{{env}}/kustomization.yml
- {env}는 develop/staging/prd

환경설정 세팅

- Config/overlays/{{env}}/secret.enc.yaml
- Config/overlays/{{env}}/kustomization.yml
- {env}는 develop/staging/main(prd)

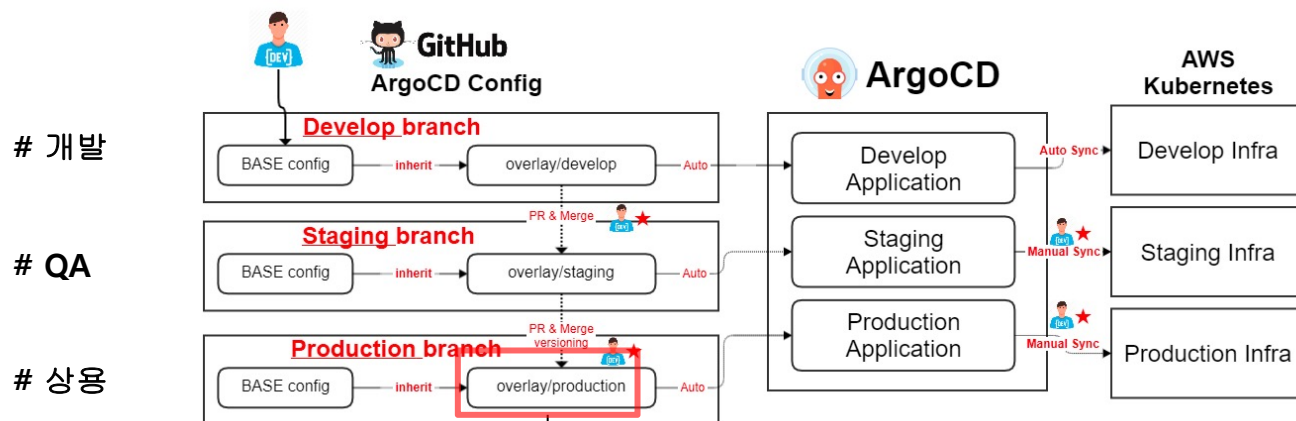
Deploy

- 각 환경에 맞게 배포



운영 관리자 확인 후 각 image를 선택적 배포

- Develop/Staging Test가 끝난 Service에 대해서 config update 하고 배포



Manual Sync

APP DIFF SYNC SYNC STATUS HISTORY AND ROLLBACK DELETE REFRESH

CURRENT SYNC STATUS

OutOfSync From main (adf28bb)

LAST SYNC RESULT

Sync OK Succeeded 4 days ago (Fri Apr 15 2022 10:38:59 GMT+0900)

Author: Mobile Wallet Platform Admin <92003017+stonskt@users...> Merge pull request #346 from sktston/staging

Comment: Merge pull request #346 from sktston/staging

CREATED_AT 09/27/2021 19:11:00

REPO URL git@github.com:sktston/staging

TARGET REVISION main

PATH overlays/main

k8s-istiosys-ingress-747dc01...

delete-user-27483078-2z4wr 16 days completed 0/2

delete-user-27490279-bs56q 11 days completed 0/2

delete-user-27496040-vrwnr 7 days completed 0/2

delete-user-27501801-mm4h6

ingress-alb

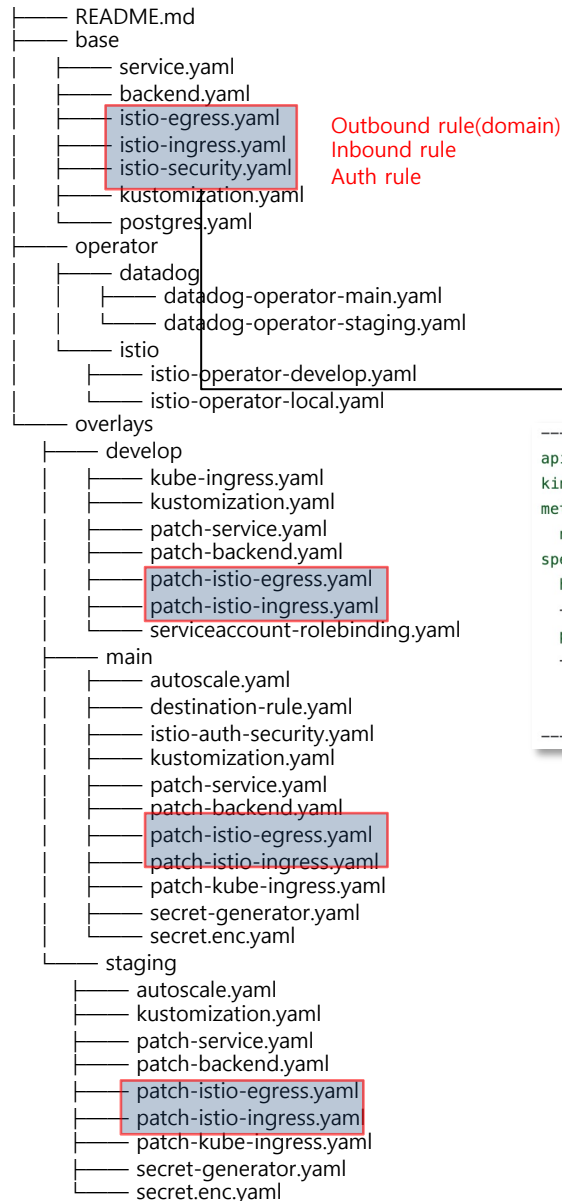
7 months

- ☐ /SERVICEACCOUNT/KUBE-SYSTEM/EKS-ADMIN ✓
- ☐ APPS/DEPLOYMENT/DEFAULT/ACA-PY ✓
- ☒ APPS/DEPLOYMENT/DEFAULT/ADMIN-FRONT ⚠
- ☒ APPS/DEPLOYMENT/DEFAULT/ADMIN-SERVER ⚠
- ☒ APPS/DEPLOYMENT/DEFAULT/AUTH ⚠
- ☐ APPS/DEPLOYMENT/DEFAULT/BATCH ✓
- ☒ APPS/DEPLOYMENT/DEFAULT/CLOUDAGENT-DATASTORE ⚠
- ☒ APPS/DEPLOYMENT/DEFAULT/CLOUDAGENT-PROXY ⚠
- ☒ APPS/DEPLOYMENT/DEFAULT/MAIN-FRONT ⚠
- ☒ APPS/DEPLOYMENT/DEFAULT/MAIN-SERVER ⚠
- ☒ APPS/DEPLOYMENT/DEFAULT/MOIS-FRONT ⚠
- ☐ APPS/DEPLOYMENT/DEFAULT/MWP-ASSET-ADMIN ✓
- ☐ APPS/DEPLOYMENT/DEFAULT/NYDUS ✓
- ☒ APPS/DEPLOYMENT/DEFAULT/ORCHESTRATOR ⚠
- ☐ APPS/DEPLOYMENT/DEFAULT/PARTNER-FRONT ✓
- ☐ APPS/DEPLOYMENT/DEFAULT/PARTNER-SERVER ✓

Git Config Tree Structure – Istio Proxy

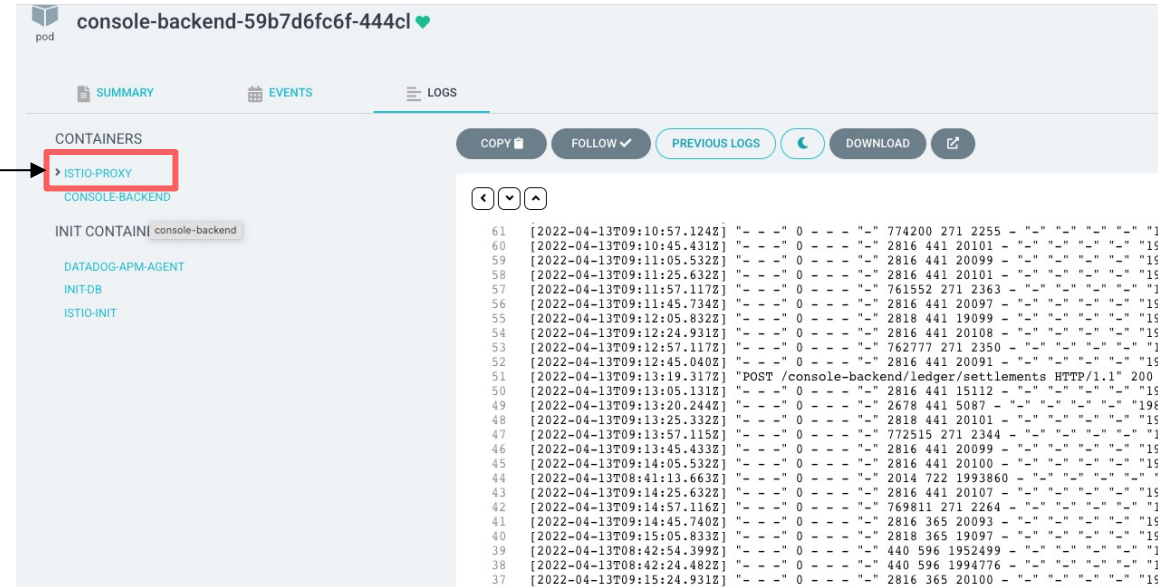


Service Mesh간 통신 처리를 위해 Istio(with circuit breaker) 적용



```
---
apiVersion: networking.istio.io/v1alpha3
kind: ServiceEntry
metadata:
  name: minwise-pass-ext
spec:
  hosts:
  - www.tauth.net
  ports:
  - number: 443
    name: https
    protocol: HTTPS
---
```

egress outbound sample



[AWS EKS-연재2] AWS EKS 에 Istio 설치 및 설정

AWS EKS 에 상용을 위한 Kubernetes 서비스 설정 시리즈 1. eksctl을 사용한 AWS EKS 생성 2. AWS EKS 에 Istio 설치 및 설정 3. AWS EKS 에서 Istio 와 Application Load Balancer 연결 4. Kubernetes 의 Resources 설정을 위한 VPA 및 Goldilocks 5. AWS EKS 의 Cluster Autoscaler 설정 6. Loki를 이용한 순서로 Kubernetes Logging 7. Kubernetes/Istio 를 위한 Datadog 설정 8. Argo CD 설치 및 설정 Isti...

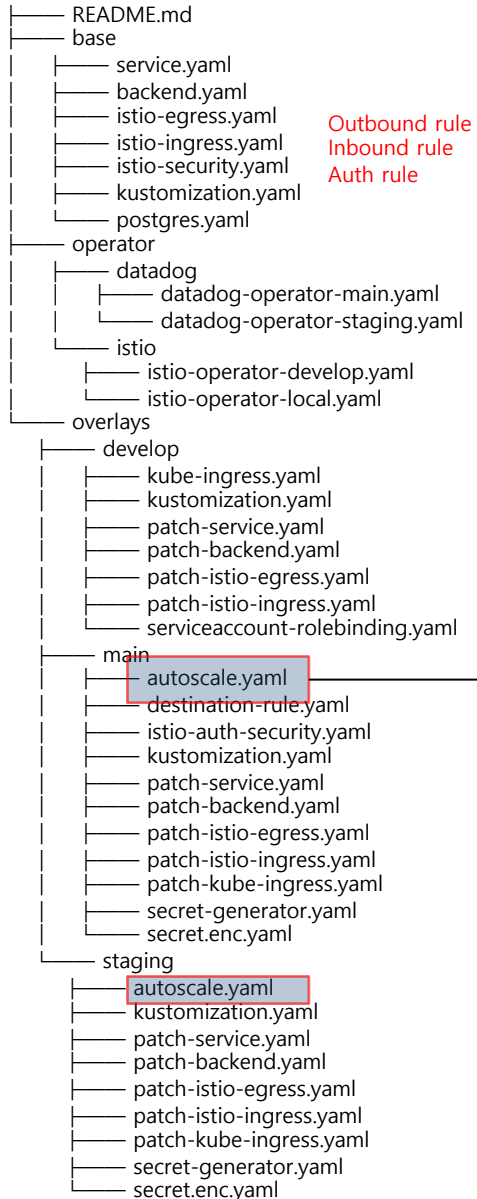
#Kubernetes #Amazon Web Service

Thomas | 22.03.02 | 398 | 5 | 0

참고 :

<https://devocean.sk.com/experts/techBoardDetail.do?ID=163655&page=&searchData=Kubernetes&subIndex=&idList=%5B163659%2C+163658%2C+163657%2C+163656%2C+163655%2C+163654%5D>

Git Config Tree Structure – Auto Scale

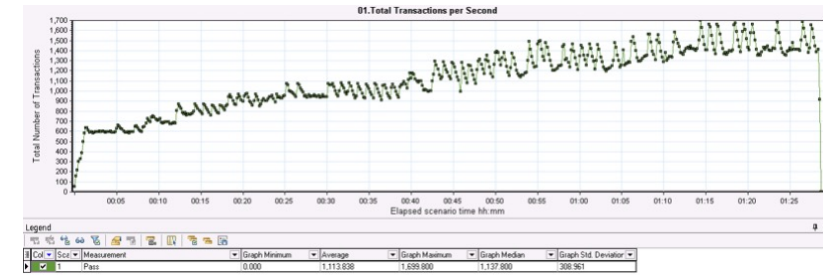


Outbound rule
Inbound rule
Auth rule

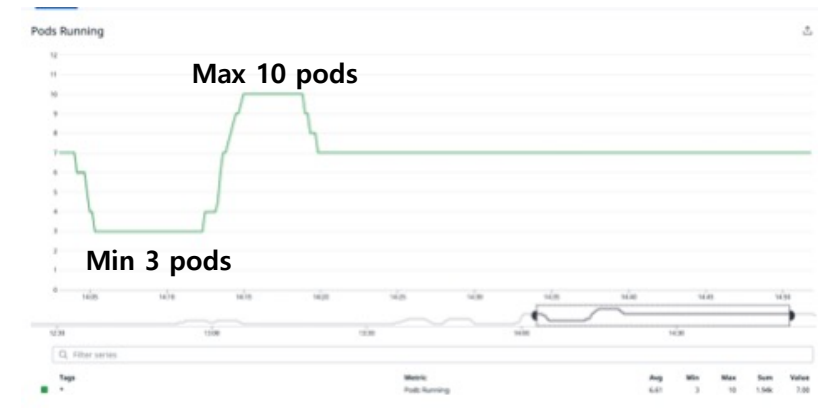
Traffic 증가 대응을 위한 Autoscale 적용

```
1  apiVersion: autoscaling/v2beta2
2  kind: HorizontalPodAutoscaler
3  metadata:
4    name: auth
5  spec:
6    scaleTargetRef:
7      apiVersion: apps/v1
8      kind: Deployment
9      name: auth
10   minReplicas: 3
11   maxReplicas: 10
12   metrics:
13     - type: Resource
14       resource:
15         name: cpu
16         target:
17           type: Utilization
18           averageUtilization: 100
19   behavior:
20     scaleDown:
21       stabilizationWindowSeconds: 300
22     policies:
23       - type: Percent
24         value: 50
25         periodSeconds: 30
26     scaleUp:
27       stabilizationWindowSeconds: 0
28     policies:
29       - type: Percent
30         value: 100
31         periodSeconds: 15
32       - type: Pods
33         value: 4
34         periodSeconds: 15
35     selectPolicy: Max
```

Total TPS



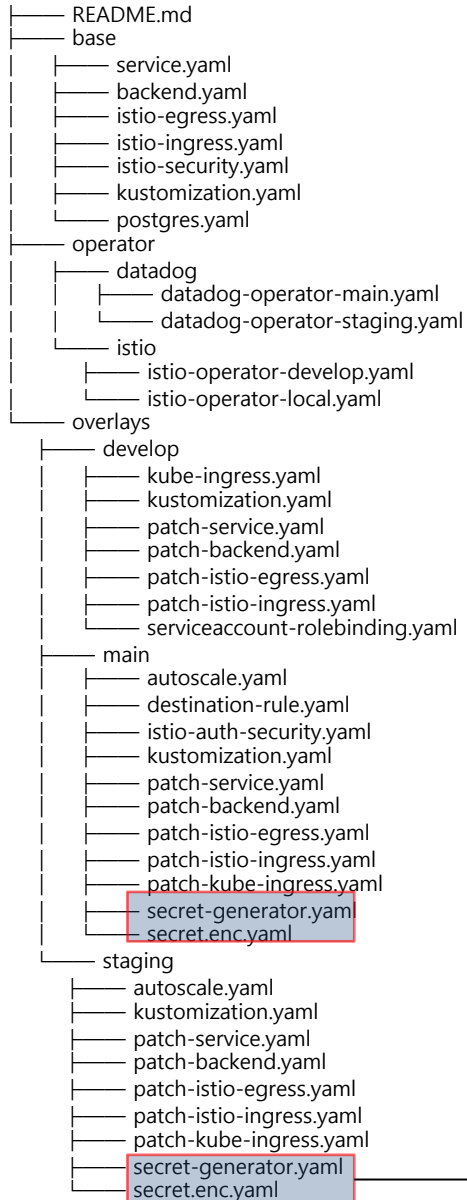
Pods Running (servicePod)



Git Config Tree Structure – KSOPS



A Flexible Kustomize Plugin for SOPS Encrypted Resource
(Kubernetes + AWS KMS + Kustomize + KSOPS)



```
27     version: 3.7.2
28     ---
29     apiVersion: v1
30     data:
31       db-pass: ENC[AES256GCM:...]
32       db-url: ENC[AES256GCM:...]
33       jwt-secret: ENC[AES256GCM:...]
34       wallet-key: ENC[AES256GCM:...]
35       redis-url: ENC[AES256GCM:...]
36     kind: Secret
37     metadata:
38       name: acapy-secrets
39     namespace: default
```

ArgoCD에서 Secret을 decryption 하여 배포



Datadog

- 좋은 Tool을 제공해 주신 회사에 감사

The screenshot shows the Datadog web interface. On the left is a dark sidebar with the Datadog logo and a list of navigation items. The main content area is titled 'All Dashboards' and lists 28 available dashboards. A search filter is at the top left of the dashboard list. The dashboards include various Kubernetes components and general system metrics.

Alarm

- Datadog Slack notification

This block contains a screenshot of a Slack notification from Datadog. The notification is titled '# monitoring' and shows a triggered alert. It includes a line graph of latency over time, with a peak at 13:15. Below the graph, it says 'Recovered: Service istio-ingressgateway has a high p90 latency on env:initial-prod'. The notification also includes a 'Mute Monitor' button and a 'Declare Incident' button.



Auto Integration test (Ops)

- Unit test는 자동화 적용하였지만, 시나리오 Test는 적용안됨 ➔ 서비스기반 자동화 검증 적용 예정 (~금년)

NFT 관련 신규 서비스 개발 (Dev)

- 새로운 기술/서비스 개발



연락주세요

- <https://devocean.sk.com/experts/view.do?page=&query=&ID=khujin&searchData=DevOps&idList=%5B163746%2C+163709%2C+163703%2C+163551%2C+163677%5D&subIndex=>

🏠 > 전문가



주재현

물고 답하고 **멘토링 신청**

이메일
***hyeon.ju@sk.com

등록일
20.09.23

게시글
3

🔖 #DevOps

SK텔레콤에서 2018년부터 블록체인의 기반 인증시스템을 Public Cloud에 개발 적용하고, 현재 상용화 완료하였습니다.
블록체인 Node 구축, Back-end, Front-end 개발등을 진행하였고, ArgoCD, K8s를 사용하여 DevOps 개발 진행하고 있습니다.

Tech 블로그

Total 3

- 

서비스중인 MSA에서 적용중인 CI/CD 정책

지난글 목록 MSA와 DevOps를 적용하여 상용 서비스 출시를 위해 힙(Hip)한 기술 모두 사용해보기 AWS 운영중인 MSA 서비스에 적용된 Kubernetes 정책 소개 이번에는 사실상 이번 시리즈의 핵심인 CI/CD 정책에 대해서 소개하고자 한다. 앞선 글에서 우리 플랫폼은 GitOps1.0을 참고 하고 있다고 하였다. GitOps의 방식 및 장단점은 구글링하면 수만개가 검색되니 별도의 설명은 생략합니다. 하지만 핵심은 아래와 같습니다...

#DevOps #Microservice Architecture #Kubernetes #Git #Amazon Web Service

 주재현 | 22.03.08 526 4 0
- 

AWS 운영중인 MSA 서비스에 적용된 Kubernetes 정책 소개

지난글 목록 MSA와 DevOps를 적용하여 상용 서비스 출시를 위해 힙(Hip)한 기술 모두 사용해보기 지난번 대략적인 소개에 이어 현재 AWS에서 서비스 운영중인 환경 및 정책에 대해서 간략히 소개하고자 한다. 그전에 현재 서비스의 조건에 대해서 정보를 드리면 아래와 같다. 대국민 서비스 (Any Open 필요) 개인정보 처리 시스템 (항분리 필요) 위 조건을 바탕으로 아래와 같은 Architecture가 결정 되었다. [Figure 1] AWS Simple...

#Kubernetes #DevOps #Amazon Web Service

 주재현 | 22.01.18 2143 6 3

Special Thanks To



김정기(Thomas)

5개

#Kubernetes

skt.1111747@sk.com

등록일 22.02.22 게시물 수 12 👤 1

관심분야: MSA, Kubernetes, Istio, AWS, CI/CD (Argo CD, Kustomize, Github Actions), Spring Boot, Hibernate

SK Telecom의 Backend 개발자입니다.

서비스 개발을 좋아하고 DevOps에 관심이 많습니다.

현재 Blockchain 플랫폼을 Kubernetes/Istio 를 활용하여 MSA 설계 및 개발하였습니다.

돌아가기



[AWS EKS-연재6] Loki를 이용한 손쉬운 Kubernetes Logging

AWS EKS 에 상용을 위한 Kubernetes 서비스 설정 시리즈 1. eksctl을 사용한 AWS EKS 생성 2. AWS EKS 에 Istio 설치 및 설정 3. AWS EKS 에서 Istio 와 Application Load Balancer 연결 4. Kubernetes 의 Resources 설정을 위한 VPA 및 Goldilocks 5. AWS EKS 의 Cluster Autoscaler 설정 6. Loki를 이용한 손쉬운 Kubernetes Logging 7. Kubernetes/Istio 를 위한 Datadog 설정 8. Argo CD 설치 및 설정 Lo...

#Kubernetes

Thomas | 22.04.14 👤 418 ♥ 3 □ 1



[AWS EKS-연재5] AWS EKS 의 Cluster Autoscaler 설정

AWS EKS 에 상용을 위한 Kubernetes 서비스 설정 시리즈 1. eksctl을 사용한 AWS EKS 생성 2. AWS EKS 에 Istio 설치 및 설정 3. AWS EKS 에서 Istio 와 Application Load Balancer 연결 4. Kubernetes 의 Resources 설정을 위한 VPA 및 Goldilocks 5. AWS EKS 의 Cluster Autoscaler 설정 6. Loki를 이용한 손쉬운 Kubernetes Logging 7. Kubernetes/Istio 를 위한 Datadog 설정 8. Argo CD 설치 및 설정...

#Kubernetes #Amazon Web Service

Thomas | 22.04.07 👤 270 ♥ 2 □ 0



[AWS EKS-연재4] Kubernetes 의 Resources 설정을 위한 VPA 및 Goldilocks

AWS EKS 에 상용을 위한 Kubernetes 서비스 설정 시리즈 1. eksctl을 사용한 AWS EKS 생성 2. AWS EKS 에 Istio 설치 및 설정 3. AWS EKS 에서 Istio 와 Application Load Balancer 연결 4. Kubernetes 의 Resources 설정을 위한 VPA 및 Goldilocks 5. AWS EKS 의 Cluster Autoscaler 설정 6. Loki를 이용한 손쉬운 Kubernetes Logging 7. Kubernetes/Istio 를 위한 Datadog 설정 8. Argo CD 설치 및 설정...

#Kubernetes

Thomas | 22.03.23 👤 287 ♥ 4 □ 0



[AWS EKS-연재3] Istio 와 Application Load Balancer 연결

AWS EKS 에 상용을 위한 Kubernetes 서비스 설정 시리즈 1. eksctl을 사용한 AWS EKS 생성 2. AWS EKS 에 Istio 설치 및 설정 3. AWS EKS 에서 Istio 와 Application Load Balancer 연결 4. Kubernetes 의 Resources 설정을 위한 VPA 및 Goldilocks 5. AWS EKS 의 Cluster Autoscaler 설정 6. Loki를 이용한 손쉬운 Kubernetes Logging 7. Kubernetes/Istio 를 위한 Datadog 설정 8. Argo CD 설치 및 설정 EK...

#Kubernetes #Amazon Web Service

Thomas | 22.03.14 👤 399 ♥ 4 □ 0



[AWS EKS-연재2] AWS EKS 에 Istio 설치 및 설정

AWS EKS 에 상용을 위한 Kubernetes 서비스 설정 시리즈 1. eksctl을 사용한 AWS EKS 생성 2. AWS EKS 에 Istio 설치 및 설정 3. AWS EKS 에서 Istio 와 Application Load Balancer 연결 4. Kubernetes 의 Resources 설정을 위한 VPA 및 Goldilocks 5. AWS EKS 의 Cluster Autoscaler 설정 6. Loki를 이용한 손쉬운 Kubernetes Logging 7. Kubernetes/Istio 를 위한 Datadog 설정 8. Argo CD 설치 및 설정 Isti...

#Kubernetes #Amazon Web Service

Thomas | 22.03.02 👤 405 ♥ 5 □ 0