박정훈

신입 클라우드 엔지니어

AWS 인프라 구축과 배포·부하 대응 검증

PetClinic 개인 프로젝트

동일 부하로 서버 사양 비교
실제 오류와 모니터링 지표 대조
HPA 확장·축소와 GitOps 배포 확인

github.com/gns140345-tech/petclinic-infra

프로젝트 목표와 수행 범위

기존 Spring Boot PetClinic을 활용해 AWS 운영 환경을 구축한 개인 프로젝트

직접 수행한 범위

AWS 인프라 구성 및 컨테이너 배포
GitHub Actions·Argo CD 배포 자동화
WhaTap·CloudWatch 모니터링
k6 부하 테스트와 운영 사양 비교

운영 시나리오

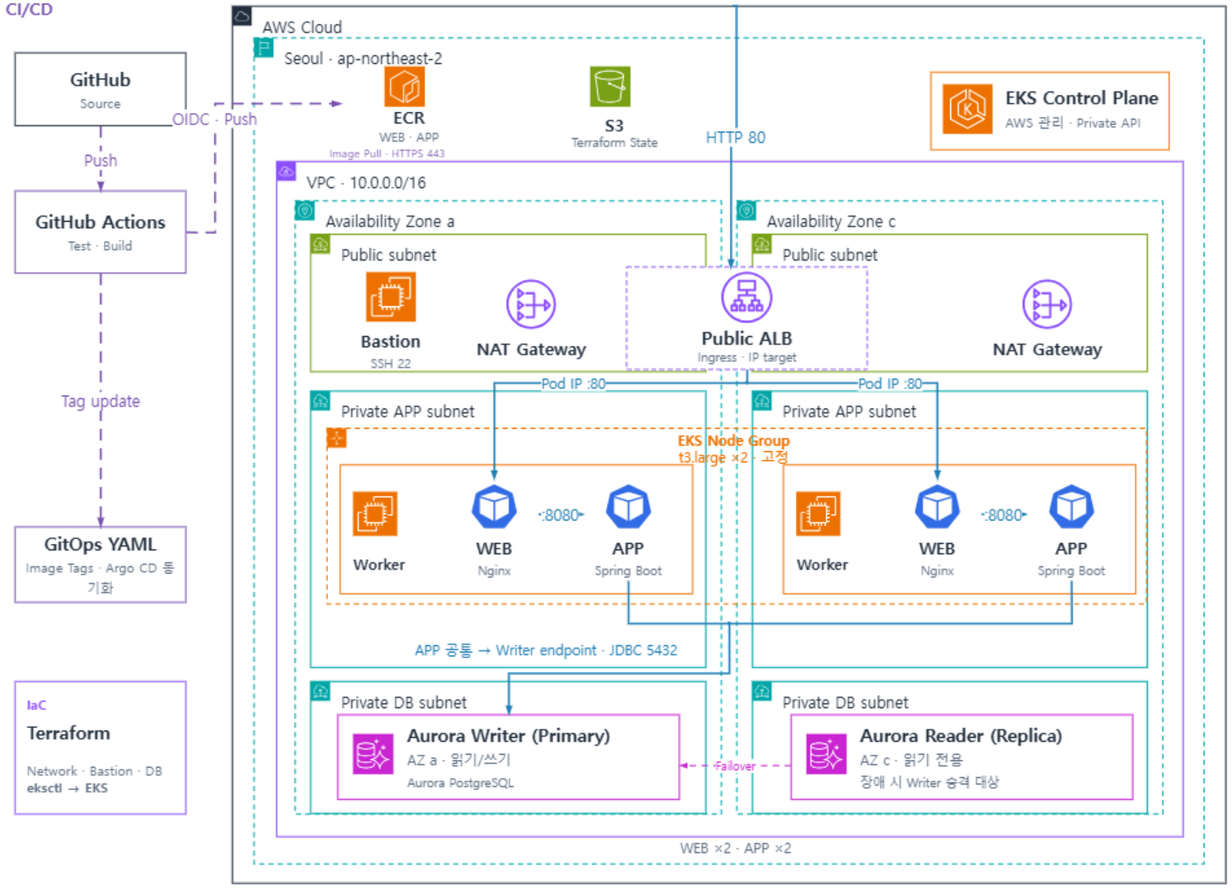
보호자·반려동물·방문 정보를 등록하고 조회하는
업무를 반복해 부하에 따른 서비스 반응 확인



실제 서비스 화면

전체 아키텍처

서비스 요청, 배포, 관리 접근의 경로를 구분



서비스 요청

ALB를 통해 WEB·APP에 접근
APP에서 Aurora로 연결

관리 접근

Bastion 경유 Private EKS API
IAM·Access Entry로 권한 제어

구성도는 초기 large 2대 기준
사양 비교 후 medium 2대 선택

인프라 구성을 코드로 관리

Terraform과 eksctl의 관리 범위를 나눠 반복 구축

선택 이유

수동 구축 시 발생할 수 있는 설정 누락과 재구축 환경의 차이를 줄이기 위해 적용

도구	관리 범위
Terraform	네트워크·보안 그룹·Bastion·Aurora
eksctl	EKS 클러스터·Managed Node Group
Kubernetes·Controller	워크로드와 관련 동적 리소스

운영 시 확인: 생성 주체와 의존 관계에 맞춰 Ingress·EKS 등 리소스 정리 순서를 기록

Private EKS의 GitOps 배포

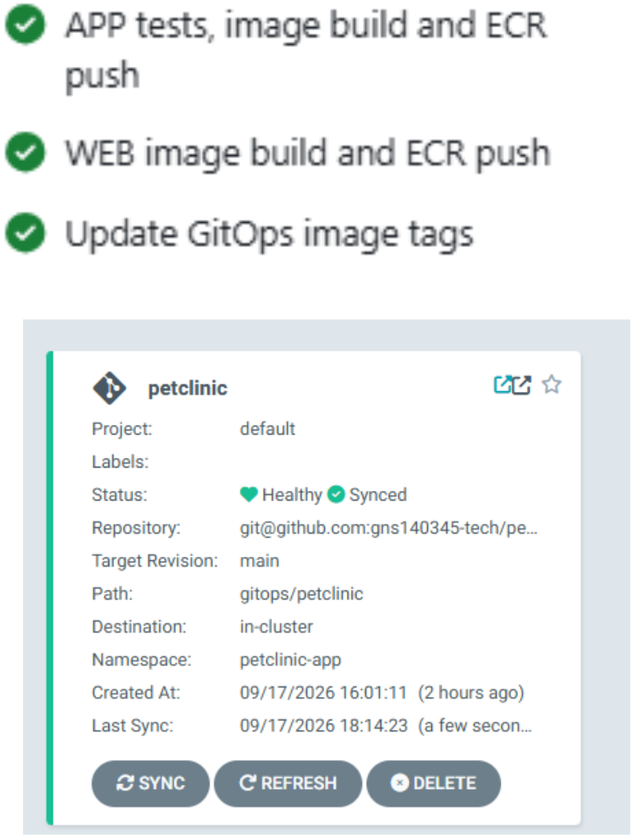
외부 CI는 이미지를 만들고, 내부 Argo CD가 Git 선언을 배포

배포 흐름

01 GitHub Actions에서 검증·이미지 빌드
02 OIDC 인증 후 ECR 업로드
03 GitOps Manifest 이미지 태그 갱신
04 Argo CD 동기화 후 EKS Rollout

선택 이유와 검증

외부 CI의 Private API 직접 접근 경로를 추가하지 않음
CI 성공, Rollout 완료, 실제 UI 반영을 각각 확인



실제 오류와 모니터링 지표 대조

WhaTap으로 APP·EKS를, CloudWatch로 ALB·Aurora를 관찰

발생시킨 상황	실습 기록에서 확인한 반응
HTTP 500 요청 30회	ALB Target 5XX 합계 30건·WhaTap 오류 증가
HTTP 404 요청 10회	ALB Target 4XX 10건
배치 조건 불일치 Pod 생성	Pending Pod 1개
실패하는 Readiness 설정	Unavailable Pod 1개

대조 방법

CloudWatch 수집 시차를 고려해 같은 테스트 시간대와 건수를 확인했습니다.
확인 범위는 지표 수집·반영이며, 경보 발송·수신은 별도 검증이 필요합니다.

동일 부하 비교로 운영 사양 선정

Node 2대·AZ 배치·업무 흐름을 유지하고 large와 medium 비교

k6 / 각 50·100·150 VU를 5분씩 실행 / HPA 적용 전 / 업무 1회 HTTP 5건

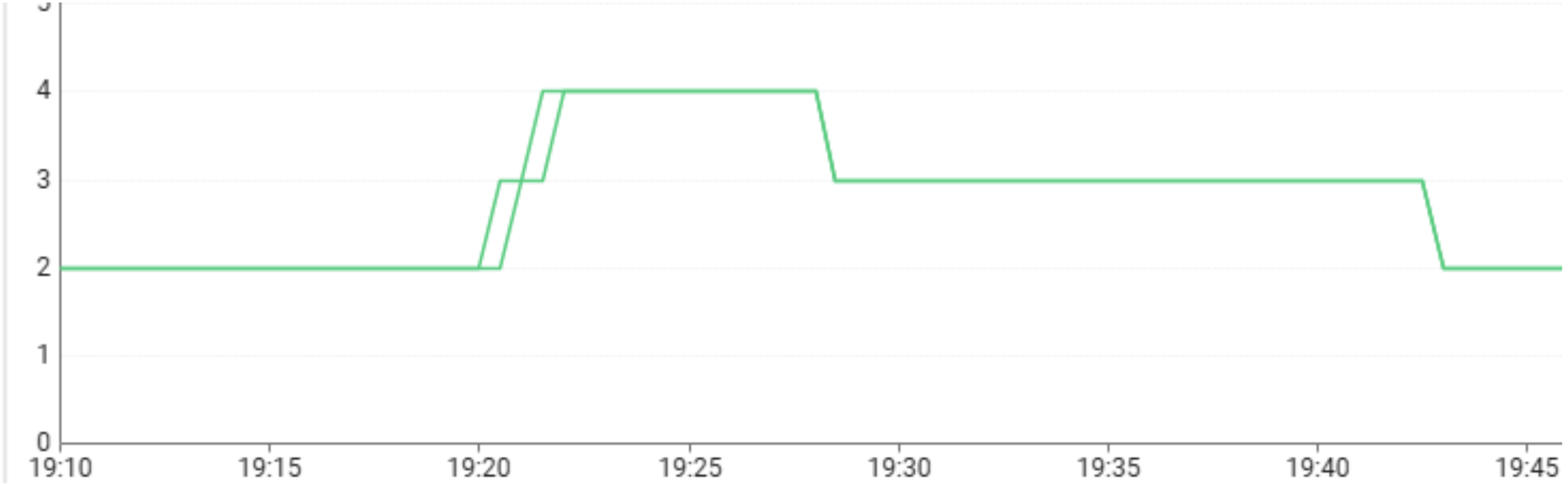
가상 사용자	large RPS	medium RPS	large P95	medium P95
50 VU	30.24	30.56	48.66ms	68.26ms
100 VU	61.14	60.25	40.06ms	48.42ms
150 VU	90.32	90.21	58.34ms	55.31ms

모든 비교 구간에서 양쪽 HTTP 실패율 0%

처리량이 유사하고 지속 포화가 관측되지 않아 medium 2대 선택.
각 시험은 5분으로, 장시간 안정성·CPU 크레딧 소모·비용 절감률은 미검증.

HPA 확장·축소와 Argo CD 역할 조정

Node 2대를 유지하면서 실제 APP Replica 변화를 확인



CPU Request 500m / HPA 목표 70% / min 2 · max 4

Replica 2 → 4 → 3 → 2를 관측했고, APP 4개가 기존 Node 2대에 정상 배치됐습니다.
Argo CD의 replicas 차이 무시 설정으로 HPA의 수량 조절과 충돌을 방지했습니다.

회고와 지원 역량

측정 조건과 검증 범위를 남기며 운영 결과를 설명

프로젝트에서 익힌 접근

조건 유지 후 한 가지 변경으로 결과 비교
발생시킨 오류와 지표의 시간·건수 대조
설정 상한과 실제 관측 결과 구분
CI 성공과 사용자 화면 반영을 각각 확인

추가 검증 과제

장시간 CPU 크레딧·메모리 변화
경보 발송·수신 및 장애 복구 시간
실제 운영 시간에 따른 전체 비용

교육

베스핀글로벌 멀티클라우드 부트캠프 4기
2026.04.02 ~ 10.08 / 1,000시간

자격·어학

AWS DevOps Engineer – Professional
AWS Solutions Architect – Associate
Azure Fundamentals
리눅스마스터 2급·네트워크관리사 2급
TOEIC 840