최한수 Choi Hansu

DevOps · Cloud Native Engineer · 경력 4년+

comstering@gmail.com · github.com/comstering · www.linkedin.com/in/hansu-choi-05775a244

개발과 운영의 경계를 좁혀 비즈니스의 연속성을 만드는 엔지니어 최한수입니다.

단순히 시스템을 구축하는 것을 넘어, 동료들이 복잡한 인프라 구조에 구애받지 않고 비즈니스 로직에만 몰입할 수 있는 환경을 설계하는 데 가치를 둡니다.

안정적인 시스템 운영을 위한 아키텍처를 고민하며, 다양한 직군과의 원활하게 소통하기 위해 기술적 지식을 넘어 도메인 전반을 이해하려고 노력합니다.
최선의 해결책을 찾기 위해 기록하고 공유하며 팀과 함께 성장하는 문화를 지향합니다.

// experience

2024.06 ~ now

CJ Olive Young

DevOps · Cloud Native Engineer

OAuth2 Proxy 기반의 Zero Trust 인증 게이트웨이 구축 및 인프라 보안 강화 2026.03

인프라 레이어 통합 인증 프록시 구축을 통한 환경별(Dev/QA/Stg) 액세스 제어 최적화 및 보안 강화

- OAuth2 Proxy를 활용한 인증 게이트웨이 구축으로 사내 시스템 접근 제어 체계 강화
- Keycloak & Okta SSO와 연동하여 통합 인증 체계 구축 및 사용자 편의성 개선
- Istio AuthorizationPolicy와 연계하여 세밀한 리소스 기반 접근 제어 정책 구현

Argo Workflow 기반 CI/CD 파이프라인 고도화 2025.12 ~ 2026.02

GitOps 기반의 선언적 워크플로우 관리 체계를 구축하여 배치 워크플로우의 안정성 및 확장성을 대폭 개선

- Argo Workflows를 활용한 복잡한 배치 작업 및 데이터 파이프라인 오케스트레이션 구현
- 일평균 100+ 워크플로우 실행 처리 및 99.9% 이상 안정적 운영 달성
- Kubernetes-native 워크플로우 엔진 도입으로 리소스 활용률 40% 개선
- 워크플로우 템플릿 표준화를 통한 재사용성 향상 및 개발 생산성 증대
- GitOps 기반 배포 자동화로 배포 리드타임 단축
- AWS MWAA 제거를 통한 연간 $16,000 비용 절감 및 완전한 Kubernetes 네이티브 CI/CD 환경 구축

Cloudflare 전환을 통한 CDN/보안 인프라 최적화 2025.11 ~ 2026.02

글로벌 CDN 성능 향상 및 DDoS/WAF 보안 강화를 위해 AWS CloudFront에서 Cloudflare로 완전 전환

- Cloudflare CDN 도입으로 글로벌 콘텐츠 전송 속도 평균 35% 개선 (TTFB 기준)
- Cloudflare WAF 규칙 최적화를 통한 월평균 200만 건 이상의 악성 요청 차단
- Blue-Green 전환 전략 수립으로 다운타임 없이 CloudFront → Cloudflare 마이그레이션 완료
- Root domain의 Cloudflare 적용을 위한 DNS 관리 체계 개선 및 최적화

Okta SSO 기반 통합 인증 체계 구축 및 Keycloak IdP Broker 아키텍처 설계 및 구현 2025.11 ~ 2026.01

제로 트러스트 보안 모델 구현을 위해 Okta SSO를 중심으로 한 사내 통합 인증 인프라를 구축하고 Keycloak과의 통합을 통해 기존 시스템과 원활히 연동

- 10+ SaaS 및 Self-Hosted 애플리케이션에 Okta SSO 연동 완료, 평균 로그인 시간 70% 단축
- Keycloak과 Okta 간 OpenID Connect 기반 페더레이션 구성으로 레거시 시스템 통합 인증 달성
- Keycloak IdP Broker 패턴 적용으로 Okta 장애 시에도 Keycloak 자체 인증을 통한 서비스 연속성 확보
- AWS Identity Center 연동을 통한 AWS 계정 접근 통제 및 임시 자격 증명 기반 보안 강화
- 개인별 IAM User 250+ 계정을 Okta SSO 역할별 권한 관리 체계로 전환하여 IAM User 계정 완전 폐기 및 보안 강화
- Terraform IaC로 Okta 및 Keycloak 리소스 관리 자동화, 신규 애플리케이션 연동 시간 단축

Kubernetes Operator 활용 인프라 운영 자동화 2025.06

Operator Pattern을 활용하여 주요 인프라 컴포넌트의 라이프사이클 관리를 자동화하고 운영 효율성 향상

- Keycloak Operator를 활용한 인증 시스템 HA 클러스터 자동 구성 및 관리
- Strimzi Kafka Operator를 통한 Kafka Connect 클러스터 선언적 관리 및 운영 자동화
- Datadog Operator 도입으로 모니터링 에이전트 배포 및 설정 관리 자동화
- Kiali Operator를 통한 Istio Service Mesh 관찰성(Observability) 대시보드 구축
- Operator Pattern 기반 선언적 리소스 관리로 수동 배포 작업 80% 감소 및 운영 일관성 확보

Keycloak 기반 엔터프라이즈 인증/인가 시스템 구축 2025.06

마이크로서비스 아키텍처 환경에서 OAuth 2.0/OIDC 기반의 중앙 집중식 Identity & Access Management 시스템 구축

- Keycloak Operator를 활용한 HA(High Availability) 클러스터 구성으로 99.9% 가용성 달성
- Terraform 기반 IaC(Infrastructure as Code)로 Realm, Client, Role 설정 자동화
- JWT 토큰 기반 Stateless 인증 구조 설계로 초당 5,000+ TPS 인증 처리 성능 확보
- Fine-grained Authorization 정책 구현으로 서비스별 세밀한 접근 제어 체계 구축

VPC Peering 전환을 통한 네트워크 성능 최적화 및 비용 절감 2025.09 ~ 2025.10

Transit Gateway 기반 허브-스포크 아키텍처의 한계를 극복하기 위해 VPC Peering 기반 메시 네트워크로 전환

- VPC Peering 직접 연결을 통한 네트워크 홉(hop) 수 감소로 패킷 손실률 0.1% 미만 유지
- TGW 데이터 처리 비용 제거로 월간 네트워크 비용 약 50% 절감 (약 $3,000 → $1,500)
- Terraform 모듈화를 통한 Peering 연결 자동화로 신규 VPC 추가 시간 90% 단축

AWS EKS 네트워킹 아키텍처 고도화 2025.06

Pod IP 고갈 문제 및 VPC CNI의 iptables 성능 한계를 극복하기 위해 Custom Networking과 Cilium CNI 도입

- VPC Secondary CIDR(100.64.0.0/16)대역을 이용한 subnet 대역 할당 및 EKS Custom Networking 구성으로 Pod IP 풀 500개 → 16,000개로 32배 확장
- Cilium의 eBPF 기반 커널 바이패스 네트워킹으로 Pod 간 통신 레이턴시 30% 개선 및 CPU 사용률 20% 절감
- VPC CNI와 Cilium CNI Chaining 구성으로 AWS Native 네트워킹 기능과 Cilium의 고급 네트워크 정책 기능을 동시에 활용
- Cilium Network Policy를 활용한 세밀한 트래픽 제어 및 보안 강화
- Hubble UI 통합으로 네트워크 수준의 가시성 확보

Temporal 기반 분산 워크플로우 엔진 구축 2025.05 ~ 2025.09, 2026.01 ~ 2026.02

대규모 마이크로서비스 환경에서 복잡한 비즈니스 트랜잭션 및 Saga 패턴 구현을 위해 Temporal Workflow Engine 도입

- Self-hosted Temporal Cluster 구축으로 SaaS 대비 연간 약 $50,000 비용 절감
- PostgreSQL vs MySQL 벤치마킹 수행 (10,000 TPS 부하 테스트) 후 최적 데이터베이스 선정
- Rainbow Deployment 패턴 설계로 워크플로우 코드 버전 간 호환성 보장 및 무중단 배포 구현
- Workflow Versioning 및 Activity Retry 정책 최적화로 장기 실행 프로세스 안정성 확보

Istio Service Mesh 기반 Zero Trust 보안 아키텍처 구현 2025.05

마이크로서비스 간 East-West 트래픽에 대한 mTLS 암호화 및 세밀한 인가 정책 적용

- Istio Sidecar Proxy 기반 자동 mTLS 암호화로 서비스 간 통신 100% 암호화 달성
- AuthorizationPolicy를 통한 JWT Claims 기반 서비스 간 접근 제어 구현
- RBAC(Role-Based Access Control) 정책 적용으로 최소 권한 원칙(Principle of Least Privilege) 준수
- Istio Ingress Gateway 최적화로 외부 트래픽 TLS Termination 성능 개선

Kafka 에코시스템 안정성 및 운영 효율성 강화 2025.04 ~ 2025.10

Kafka 클러스터 및 Connect 시스템의 안정성을 높이고, 실시간 CDC 파이프라인 구축을 통한 데이터 동기화 자동화

- Strimzi Kafka Operator 도입으로 Kafka Connect Cluster 재시작 시간 80% 단축 및 자동 복구 체계 구축
- Kafbat UI(구 Kafka UI) 커스터마이징 및 오타 버그 수정 기여로 오픈소스 컨트리뷰션 완료
- Bitnami Confluent Schema Registry Helm Chart 개선 PR 제출 및 머지 (커뮤니티 기여)
- Debezium Kafka Connect를 활용한 CDC(Change Data Capture) 파이프라인 구축으로 실시간 데이터 동기화 지연 시간 3초 이내 달성
- Kafka Consumer Lag 모니터링 및 알람 체계 구축으로 데이터 유실 사전 방지

Event-Driven Architecture 기반 Kafka 메시징 시스템 구축 2024.07 ~ 2024.10

마이크로서비스 간 비동기 통신 및 이벤트 기반 아키텍처 구현을 위한 엔터프라이즈급 Kafka 플랫폼 구축, Transaction Outbox Pattern 적용

- AWS MSK(Managed Streaming for Kafka) 3-broker 클러스터 구성으로 초당 50,000+ 메시지 처리 성능 확보
- Kafka Connect 기반 CDC 환경 구축으로 데이터베이스 변경 이벤트 실시간 스트리밍
- Transactional Outbox Pattern 아키텍처 설계 및 구현으로 분산 트랜잭션 일관성 보장
- 커스텀 SMT(Single Message Transform) 개발로 Outbox 메시지 자동 변환 및 라우팅 구현
- CloudEvents 스펙 기반 Schema Registry 구축으로 이벤트 메시지 표준화 및 버전 관리 체계 확립
- 공통 Kafka Producer/Consumer 라이브러리 개발로 개발 생산성 향상 및 코드 중복 제거

Cloud Native 인프라 구축 및 MSA 전환 프로젝트 2024.07 ~ 2025.02

신규 마이크로서비스 아키텍처 시스템을 위한 Kubernetes 기반 클라우드 네이티브 인프라 설계 및 구축

- Multi-AZ VPC 네트워크 아키텍처 설계 (Public/Private/DataBase 서브넷 3-tier 구조)
- EKS 클러스터 구축 및 Karpenter 기반 Auto Scaling 구성으로 노드 프로비저닝 시간 단축 및 비용 최적화
- Istio Service Mesh 도입으로 서비스 간 통신 가시성 확보 및 트래픽 관리 체계 구축
- GitHub Actions 기반 CI/CD 파이프라인 구축
- ARC(Actions Runner Controller) Self-hosted Runner 운영으로 빌드 비용 월 99% 절감

보안 및 운영 효율성 개선 프로젝트 2024.07 ~ 2025.12

보안 강화 및 인프라 운영 최적화를 위한 다양한 개선 작업 수행

- AWS KMS Envelope Encryption 아키텍처 구현으로 민감 데이터 암호화 및 키 관리 자동화 (2025.12)
- External Secrets Operator 도입으로 Kubernetes Secret과 AWS Secrets Manager 동기화 자동화, 시크릿 로테이션 프로세스 구축 (2025.08)
- ECR Pull Through Cache 설정으로 Docker Hub Rate Limit 문제 해결 및 이미지 Pull 시간 단축, 네트워크 전송 비용 월 30% 절감 (2025.02)
- Braze 이메일 발송 인프라 구축 및 Route53 SPF/DKIM 레코드 관리로 이메일 전송 성공률 98% 이상 유지 (2025.01)

2022.05 ~ 2024.03

CLASS 101

Software Engineer

통합 인증 시스템 재구축 (Rebuild Auth) 2023.10 ~ 2024.03

레거시 모놀리식 인증 시스템을 MSA 환경에 적합한 중앙 집중식 인증 서비스로 전면 재설계

- OAuth 2.0/OIDC 기반 소셜 로그인 통합 인터페이스 설계 및 구현 (Google, Apple, Facebook, Kakao, Naver, Line 6개 제공자 지원)
- JWT 기반 Stateless 인증 아키텍처로 수평 확장성 확보
- Kubernetes 배포 파이프라인 구축 (Helm Chart 작성 및 Drone CI 연동)으로 배포 자동화 달성

프로모션 인증 코드 시스템 구축 2023.09 ~ 2023.10

난수 기반 일회성 프로모션 코드 발급 및 검증 시스템을 설계하여 마케팅 캠페인 지원

- 난수 기반 사전 프로모션 코드 생성
- 프로모션 코드 검증 API 개발 및 기존 시스템과 연동

회원 탈퇴 프로세스 개선 및 GDPR 준수 체계 구축 2023.06 ~ 2024.01

개인정보 보호 규정 준수 및 사용자 경험 개선을 위한 탈퇴 프로세스 전면 재설계

- 탈퇴 요청 비동기 처리 파이프라인 설계로 탈퇴 처리 시 개인정보 삭제시간 단축
- Kafka 이벤트 기반 데이터 삭제 워크플로우 구축으로 관련 시스템 자동 연동
- 탈퇴 사유 수집 및 분석 대시보드 구축으로 이탈률 개선 인사이트 제공

글로벌 성인 인증 시스템 리뉴얼 2023.04 ~ 2023.06

국내외 사용자 대상 연령 확인 프로세스를 재구축하여 법적 요구사항 충족 및 사용자 편의성 개선

- 국내 PG사 휴대폰 본인인증 API 연동으로 실시간 성인 인증 구현
- 해외 사용자 대상 연령 확인 시스템 구축

HR 연동 직원 관리 시스템 (Employee Service) 구축 2022.10 ~ 2023.01

FLEX HR 시스템과 연동하여 사내 직원 및 조직 정보를 실시간 동기화하는 내부 서비스 개발

- 정규화된 직원/부서 데이터 모델 설계 및 MongoDB document 스키마 구축
- FLEX API 연동 일배치 작업 개발로 직원 정보 자동 동기화 (오차율 0.1% 미만)
- Helm Chart 작성 및 Drone CI 배포 자동화로 배포 시간 단축

CLASS101+ 구독 앱 런칭 프로젝트 2022.07 ~ 2022.08

구독 비즈니스 모델 전환에 맞춘 React Native 기반 모바일 앱 인증 프로세스 재구축

- React Native 기반 OAuth 2.0 소셜 로그인 플로우 구현 (iOS/Android 동시 지원)

플랫폼 안정성 및 컴플라이언스 개선 작업

서비스 품질 향상 및 외부 정책 준수를 위한 인프라 개선 작업

- Gmail Sender Guidelines 준수를 위한 SPF/DKIM/DMARC 레코드 설정 및 이메일 발송 도메인 평판 관리 (2024.01)
- Spring Boot 2.3.3 → 3.2.0 메이저 업그레이드 및 JDK 11 → 17 마이그레이션 완료 (2023.11)
- 모노레포 내 OTP 인증 Web 프로젝트 추가 및 Drone CI 파이프라인 구축으로 빌드 자동화 달성 (2022.07)

// open source · side

Schema Registry helm chart contribution 2025.08

SASL_SSL 인증 과정에서 mTLS 인증이 필수인 부분을 선택적으로 변경하는 데 기여했습니다.

pull request (https://github.com/bitnami/charts/pull/35772)

kafbat-ui contribution 2025.07

kafbat-ui의 일부 문구를 수정하여 UX를 개선하는 데 기여했습니다.

pull request (https://github.com/kafbat/kafka-ui/pull/1205)

Lomeone 2024.01 ~ now

개인 Side Project를 진행하는 organization. Infrastructure를 AWS로 구축하고 Terraform을 사용하여 IaC를 적용해서 Infrastructure를 관리하고 있습니다.

github (https://github.com/lomeone)

// skills

platform	Kubernetes, EKS, Karpenter, Helm, Docker, Kubernetes Operator, Istio, Cilium, Hubble, Kiali
cloud · iac	AWS, Cloudflare, Terraform
ci/cd · workflow	GitHub Actions (ARC), Argo Workflows, Argo CD, GitOps, Temporal
data · messaging	Kafka, Kafka Connect, Debezium, Schema Registry, MongoDB, PostgreSQL, MySQL
auth · security	Keycloak, Okta, OAuth 2.0 / OIDC, JWT, OAuth2 Proxy, External Secrets Operator
backend	Kotlin, Spring Boot
observability	Datadog, OpenTelemetry, Prometheus

// certifications

HashiCorp Certified: Terraform Associate (004) 2026.04

HashiCorp

verify (https://www.credly.com/badges/476460b7-b5dc-4918-804e-ea3a169435d2/public_url)

정보처리기사 (Engineer Information Processing) 2021.06

HRDK

정보보안산업기사 (Industrial Engineer Information Security) 2021.06

KISA

// education · activity

공주대학교 (Kongju National University) 2016.03 ~ 2022.02

Computer Science, Bachelor of Engineering (B.E.). Data Structures, Algorithms, Database Design, Computer Networks, Software Engineering, Security

BoB 10기 2021.07 ~ 2022.03

보안제품개발트랙에 참여하여 Host 기반 네트워크 제어 솔루션(ARP Spoofing 기반 네트워크 관제도구)을 개발했습니다.
(Top 30)