

# 이재철 | Server Developer (Platform)

“플랫폼 안정성과 개발자 생산성을 동시에 높이는 엔지니어”

---

## ☑ 연락처

- Email: [qws941@kakao.com](mailto:qws941@kakao.com)
  - Phone: 010-5757-9592
  - GitHub: [github.com/qws941](https://github.com/qws941)
  - Portfolio: [resume.jclee.me](https://resume.jclee.me)
- 

## ☑ 지원 동기

토스 커머스의 **Server Developer (Platform)** 포지션에 지원합니다.

7년간 금융·제조·교육 산업에서 **대규모 인프라 운영, 플랫폼 안정화, 자동화 시스템 구축**을 경험했습니다. 특히 토스 Platform 팀이 추구하는 **공통 서비스 제공**(Distributed Locks, Logging, Metrics), **Gateway 운영**(Authentication, Security), **안정적인 서비스 운영** 철학에 깊이 공감합니다.

토스 커머스 Platform 팀에 기여할 수 있는 핵심 역량:

- 대용량 시스템 운영 경험**
  - 초당 10만 이벤트 처리 가능 아키텍처 설계 (Splunk-FortiNet Integration)
  - 80대 방화벽 실시간 중앙관리 플랫폼 운영
  - Redis 7 기반 실시간 데이터 처리 시스템 구축
- Monitoring & Alert 전문성**
  - Grafana/Prometheus/Loki/Tempo 통합 Full-Stack Observability 구축
  - Universal Observability 아키텍처 설계 및 운영
  - 실시간 로그 스트리밍 파이프라인 (Promtail → Loki → Grafana)
  - 장애 대응 시간 40% 단축 (평균 45분 → 27분)
- API Gateway & Load Balancing**
  - Cloudflare Workers Edge API 설계 및 운영
  - FortiManager/FortiAnalyzer API 직접 연동
  - F5, HAProxy, Nginx 실무 경험
  - 전국 동시 접속 처리 가능 아키텍처 검증 (SafeWork 플랫폼)
- 플랫폼 안정성 & 장애 복구**
  - 99.9% 가용성 달성 (연간 다운타임 8.7시간 이하)
  - 자동 롤백 시스템 구현 (배포 실패 시 자동 복구)
  - MTTR 70% 개선 (AI 기반 장애 분석)
  - 3-Port 배포 전략으로 서비스 중단 0건 달성
- 컨테이너 오케스트레이션**
  - Docker, Kubernetes, Portainer API 실무 경험
  - Private Registry 운영 (registry.jclee.me)
  - Multi-Port Deployment, Blue-Green, Canary 배포 전략
  - Watchtower 기반 자동 업데이트 및 무중단 배포
- 개발자 생산성 향상**
  - Python/Shell 자동화로 업무 시간 50~95% 단축
  - CI/CD 파이프라인 완전 자동화 (GitHub Actions, GitLab CI)
  - 150+ 자동화 명령어 체계 구축 (SlashCommand 시스템)

- 반복 업무 자동화로 운영 효율 80% 향상

---

## ☑ 핵심 역량 요약

Platform

	Redis 7	10
API Gateway	Cloudflare Workers, FortiManager API	
Monitoring & Alert System	Grafana/Prometheus/Loki , 24/7	
ELK Stack	Wazuh+Kibana, ELK	
HAProxy / Load Balancing	F5, HAProxy, Nginx	
	MTTR 70% ,	
	Flask, Node.js	Production-Ready 5

실적 기반 증명: - ☑ 안정성: 99.9% 가용성 달성 (프로덕션 시스템) - ☑ 확장성: 75,000% 확장 여유 검증 (엔터프라이즈급 아키텍처) - ☑ 성능: 장애율 35% 감소, 대응 시간 40% 단축 - ☑ 자동화: 업무 시간 50~95% 단축, 운영 효율 80% 향상

---

## ☑ 경력사항

(주)아이티센 CTS | 정보보안 운영 엔지니어

2025.03 ~ 현재 (7개월) | 넥스트레이드 운영SM

### 플랫폼 안정성 & 모니터링

- 15종 보안 솔루션 통합 운영: Fortigate, Palo Alto, NAC, DLP, EDR, APT, WAF, IPS, SIEM 등
- 24/7 실시간 모니터링: 보안 이벤트 중앙 집중식 모니터링 및 장애 대응
- 장애율 35% 감소: 보안 솔루션 튜닝 및 최적화 (월 8건 → 5건)
- 대응 시간 40% 단축: 자동화된 이벤트 대응 프로세스 구축 (평균 45분 → 27분)

### 규제 준수 & 컴플라이언스

- 금융감독원 정기 감사: 지적사항 0건 달성 (2회 연속)
  - 보안 정책 최적화: 오탐률 50% 감소 (일 200건 → 100건)
- 

(주)가온누리정보시스템 | 프리랜서 인프라 엔지니어

2024.03 ~ 2025.12 (1년 10개월) - 관택투자일임과 겸직 | 넥스트레이드 구축 프로젝트

### 플랫폼 자동화 & 효율화

- 방화벽 정책 자동화: Python 스크립트로 100건+ 일괄 배포
- 작업시간 50% 단축: 수동 작업 대비 8시간 → 4시간
- 에이전트 최적화: EPP/CPP-DLP 충돌 해결, CPU 사용률 30% 개선

## 보안 아키텍처 설계

- 망분리 구축: 금융위원회 규제 준수, 정보 유출사고 0건 (12개월)
  - 15종 보안 솔루션 통합: DDoS, IPS, WAF, VPN, NAC, DLP, SSL 복호화, APT 등
- 

## (주)관텍투자일임 | 인프라·정보보호팀 엔지니어

2022.08 ~ 2025.12 (3년 5개월) - 가온누리정보시스템과 겸직 | AI 기반 주식투자 서비스

### 클라우드 플랫폼 운영

- AWS 클라우드 보안: VPC, IAM, CloudTrail, GuardDuty 구성
- 150대+ 서버 운영: 금융보안데이터센터(FSDC) 운영
- 자동화 효과: Python 스크립트로 장애율 40% 감소 (월 10건 → 6건)

### 성능 최적화

- DB 쿼리 튜닝: CPU 사용률 30% 개선 (75% → 52%)
  - POC 성공: PB 플랫폼 성능 검증, 목표 대비 120% 달성
- 

## (주)편앤씨 | DevOps 엔지니어

2022.05 ~ 2022.07 (3개월) | 클라우드 인프라

### CI/CD & 자동화

- AWS 아키텍처 구축: EC2, Auto Scaling, VPC, Route 53, S3
  - 백업/복구 자동화: MTTR 50% 단축 (평균 2시간 → 1시간)
  - 보안 강화: CI/CD 파이프라인 보안 스캔 추가, 취약점 조기 발견율 80% ↑
- 

## (주)조인트리 | 인프라·시스템 엔지니어

2021.09 ~ 2022.04 (8개월) | 국민대학교 차세대 정보시스템

### 네트워크 & 보안 플랫폼

- Fortigate UTM, VMware NSX-T: 네트워크 세분화 및 SDN 구성
  - 고가용성 아키텍처: 이중화 설계로 99.9% 가용률 유지
  - OSS 모니터링 구축: Wazuh + Kibana 기반 보안 모니터링
- 

## (주)메타엠 | 인프라·시스템 엔지니어

2020.08 ~ 2021.09 (1년 2개월) | 대규모 콜센터 인프라

## 대규모 시스템 운영

- 1,000명 규모 재택근무 환경: Fortigate SSL VPN, NAC 통합
  - Ansible 자동화: NAC 예외정책 자동 배포, 처리시간 90% 단축 (건당 30분 → 3분)
  - Python 자동화: Cisco 스위치 점검 자동화, 주당 75% 시간 단축 (8시간 → 2시간)
- 

## (주)엠티데이터 | 서버·시스템 엔지니어

2018.10 ~ 2019.11 (1년 2개월) | 한국항공우주산업(KAI)

### 보안 운영 & 정책 관리

- Linux 서버 50대+ 운영: 보안 패치 및 취약점 관리
  - 방화벽 정책 최적화: 중복 룰 30% 제거 (3,000개 → 2,100개)
  - 망분리 구축: 제조망-개발망 물리적 분리, 유출사고 0건 (21개월)
- 

## ☑ 주요 프로젝트

### 1. Splunk-FortiNet Integration Platform (2024 ~ 현재)

Production-Ready | 대규모 실시간 이벤트 처리 플랫폼

#### 기술스택

- Backend: Node.js, Cloudflare Workers (Edge API)
- Architecture: DDD Level 3, 9개 도메인 분리 설계
- API Integration: FortiManager/FortiAnalyzer JSON-RPC API 직접 연동
- Monitoring: Splunk 통합 로그 분석 및 정책 검증

#### 대용량 트래픽 처리 성과 ☑

- ☑ 초당 10만 이벤트 처리 검증 (100,000 events/sec)
- ☑ 75,000% 확장 여유 확보 (엔터프라이즈 규모 대응 가능)
- ☑ 80대 방화벽 동시 관리 (실시간 정책 동기화)
- ☑ API 응답 시간 < 100ms (p99 latency 유지)
- ☑ 무중단 운영 365일 (Zero Downtime)

#### 토스 Platform 팀 기여 포인트

- 대용량 메시지 발송 시스템 설계 경험: 초당 10만 이벤트 처리 아키텍처
- API Gateway 패턴: Cloudflare Workers Edge Computing
- 실시간 Monitoring: Splunk 기반 로그 분석 및 메트릭 수집

☑ GitHub: [github.com/qws941/splunk](https://github.com/qws941/splunk) ☑ Live Demo: [splunk.jclee.me](https://splunk.jclee.me)

---

### 2. REGTECH Blacklist Intelligence Platform (2024 ~ 현재)

Production | 고가용성 & 자동 복구 아키텍처

## 기술스택

- **Backend:** Python Flask
- **Database:** PostgreSQL 15, Redis 7 (Caching & Pub/Sub)
- **Container:** Docker, Portainer API
- **CI/CD:** GitHub Actions, AI 기반 자동 배포
- **Monitoring:** Prometheus + Grafana

## 안정성 & 장애 복구 성과 ☑

- ☑ **99.9% 가용성 달성** (연간 다운타임 < 9시간)
- ☑ **MTTR 70% 개선** (평균 60분 → 18분)
- ☑ **자동 롤백 시스템:** 배포 실패 시 < 30초 내 자동 복구
- ☑ **AI 기반 장애 분석:** 로그 패턴 인식 및 자동 대응
- ☑ **무중단 배포:** Blue-Green Deployment (Portainer API)

## 토스 Platform 팀 기여 포인트

- **안정적인 서비스 운영:** 99.9% 가용성 검증
- **장애 복구 자동화:** MTTR 70% 개선 경험
- **Profiling & 성능 최적화:** Redis 캐싱, DB 쿼리 튜닝

☑ **GitHub:** [github.com/qws941/blacklist](https://github.com/qws941/blacklist) ☑ **Live Demo:** [blacklist.jclee.me](https://blacklist.jclee.me)

---

## 3. FortiGate Policy Orchestration Platform (2024 ~ 현재)

### Production | 3-Port 고가용성 배포

## 기술스택

- **Backend:** Python Flask
- **API:** FortiManager JSON-RPC
- **Container:** Docker, Portainer, GitHub Actions
- **Deployment:** 3-Port 배포 전략 (7777/7778/7779)

## 핵심 성과

- ☑ **정책 검증 시간 80% 단축** (자동 검증)
- ☑ **서비스 중단 0건** (3-Port HA 아키텍처)
- ☑ **감사 대응 시간 90% 절감** (변경 이력 자동 추적)
- ☑ **무중단 배포** (Production/Backup/Development 동시 운영)

## Platform 팀 기여 포인트

- 3-Port 고가용성 배포 전략 설계
- FortiManager API 직접 연동 경험
- 정책 변경 추적 시스템 구축

☑ **GitHub:** [github.com/qws941/fortinet](https://github.com/qws941/fortinet) ☑ **Live Demo:** [fortinet.jclee.me](https://fortinet.jclee.me)

---

## 4. Full-Stack Observability Platform (2024 ~ 현재)

### Production | 토스 요구사항 100% 충족 | 11개 대시보드 운영

기술스택 (토스 요구사항)

- Prometheus ☑: 메트릭 수집 및 알림 (토스 필수)
- Loki ☑: 로그 집계 시스템 (토스 필수 ELK Stack 대체)
- Grafana: 통합 대시보드 및 시각화 (11개 대시보드)
- Tempo: 분산 트레이싱
- Promtail: 로그 수집 에이전트
- Traefik: Reverse Proxy, SSL 자동화
- Docker Compose: Container Orchestration

Monitoring & Observability 성과 ☑

- ☑ Prometheus: 5개 Production 시스템 실시간 메트릭 수집
- ☑ Loki: 초당 10만+ 로그 이벤트 처리 (15개 AI Agent 포함)
- ☑ Grafana: 11개 대시보드 운영 (총 121개 패널), 24/7 모니터링
- ☑ Alertmanager: 실시간 알림 (Slack, Email 통합)
- ☑ Tempo: 분산 트레이싱 (마이크로서비스 성능 분석)

11개 운영 대시보드 세부사항

대시보드	패널 수	목적
1. Agent Activity	14	15개 AI Agent 실시간 모니터링
2. CI/CD Pipeline	15	(Dashboard/Deploy/Infra/Log/Monitoring) Guardian Protocol 5-phase 파이프라인 (7 jobs, 98% 성공률)
3. Infrastructure Health	16	Container 상태, Core 라이브러리, 시스템 리소스
4. Performance	4	실시간 성능 메트릭 (CPU, Memory, I/O)
5. System Performance	13	15개 AI Agent 인프라 성능
6. Portfolio	16	DevOps 포트폴리오 메트릭
7. Security & Compliance	16	CLAUDE.md v10.1 Constitutional Compliance (28 checks, 100% 달성)
8. Test & Code Quality	16	Jest 73.8% 커버리지, ESLint 9.x, Prettier
9. Claude Tasks	7	Background 작업 모니터링
10. Compliance	4	Constitutional Compliance 요약
11. Project Health	10	전체 프로젝트 건강 상태
총 합계	121 패널	종합 Observability

Constitutional Compliance 검증

- ☒ 28개 자동 검증 항목: Project Structure (5), Environmental (3), Observability (3), Guardian Protocol (4), Testing (5), Documentation (4), Core Libraries (3)
- ☒ 100% 컴플라이언스 점수: CLAUDE.md v10.1 헌법 준수
- ☒ Zero 위반: Local Grafana 금지, Environmental Map 필수 참조 등 엄격한 규칙 적용

#### 토스 Platform 팀 기여 포인트

- **Prometheus 운영 경험:** 토스 핵심 모니터링 도구 숙련 (5개 시스템 실전 운영)
- **Logging 시스템:** Loki 기반 통합 로깅 (초당 10만 이벤트, 15개 Agent 로그 통합)
- **Metrics 수집:** 실시간 메트릭 파이프라인 설계 및 운영 (121개 패널 관리)
- **대시보드 설계:** 11개 Production 대시보드 직접 설계 및 운영 (JSON 직접 작성)
- **CI/CD 모니터링:** Guardian Protocol 5-phase 파이프라인 가시성 확보

☒ GitHub: [github.com/qws941/grafana](https://github.com/qws941/grafana) ☒ Live Demo: [grafana.jclee.me](https://grafana.jclee.me)

---

### 5. SafeWork Industrial Health Platform (2024 ~ 현재)

#### Production | Edge Processing & Hybrid Architecture

##### 기술스택

- **Backend:** Flask 3.0, PostgreSQL 15, Redis 7
- **Edge API:** Cloudflare Workers
- **Architecture:** Hybrid (Monolithic + Microservices)

##### 핵심 성과

- ☒ 전국 동시 접속 처리 (Edge API)
- ☒ 집계 오류 100% 제거 (디지털 전환)
- ☒ 다수 중소기업 실운영 (SaaS 플랫폼)
- ☒ 실시간 분석 (Real-time Analytics)

#### Platform 팀 기여 포인트

- Edge Processing 아키텍처 설계
- Hybrid Architecture 경험 (폐쇄망 지원)
- 실시간 데이터 처리 파이프라인

☒ GitHub: [github.com/qws941/safework](https://github.com/qws941/safework) ☒ Live Demo: [safework.jclee.me](https://safework.jclee.me)

---

### ☒ 기술 스택

#### 토스 Platform 팀 요구사항 기준 매칭

##### 1. Container & Orchestration ☒☒☒☒☆ (85%)

Docker	- Production	5
Kubernetes	- POC	(PB )
Istio	-	(Service Mesh )
Portainer API	-	
Private Registry	-	registry.jclee.me

## 2. Monitoring & Observability 📊📊📊📊 (100%)

Prometheus	-	( )
ELK Stack	-	Wazuh + Kibana ( )
Grafana	-	Full-Stack Observability
Loki	-	
Tempo	-	
Pinpoint	-	APM ( : Grafana Tempo)
Splunk	-	

## 3. API Gateway & Load Balancing 📊📊📊☆ (90%)

Cloudflare Workers	-	Edge API
Spring Cloud Gateway	-	( : Cloudflare Workers)
Traefik	-	Reverse Proxy, SSL
HAProxy	-	
Nginx	-	Reverse Proxy, Load Balancing
F5	-	LB

## 4. Automation & CI/CD 📊📊📊☆ (80%)

Python	-	Flask, (3)
Shell Script	-	,
Ansible	-	, IaC
Terraform	-	AWS
GitHub Actions	-	CI/CD
GitLab CI	-	
Jenkins	-	( : GitHub Actions)

## 5. Database & Caching 📊📊📊☆ (85%)

Redis	-	Redis 7 Production ( )
PostgreSQL 15	-	Production DB
MySQL	-	(PostgreSQL )
MongoDB	-	(NoSQL)
DB	-	, (CPU 30% )

## 6. Cloud & Virtualization

AWS	-	VPC, IAM, EC2, S3, CloudTrail, GuardDuty
VMware vSphere	-	
VMware NSX-T	-	SDN,

## 7. Security & Compliance

Fortigate	-	, API
Palo Alto	-	
WAF, IPS, DDoS	-	
NAC, DLP, EDR, APT	-	
SIEM, SOAR	-	

## 8. Backend Development 📊📊☆☆ (70%)



Kotlin	- ( )
Java	- ,
Spring Boot	- (Flask )
Spring Webflux	- (Reactive Programming)
Netty	- ( )
Node.js	- Backend API ( )
Flask 3.0	- Python Web Framework
JavaScript/TypeScript	- Frontend & Backend
RESTful API	- API

9. Messaging & Queue ☒☆☆☆ (40%)

Kafka	- ( )
Redis Pub/Sub	-

☒ 자격증

자격증명	발급기관	취득일
CCNP	Cisco Systems	2020.08
RHCSA	Red Hat	2019.01
CompTIA Linux+	CompTIA	2019.02
LPIC Level 1	Linux Professional Institute	2019.02

☒ 학력

한양사이버대학교 | 컴퓨터공학과 2024.03 ~ 재학중

☒ 토스 Platform 팀에서 이루고 싶은 것

1. 공통 서비스 제공 (Distributed Locks, Logging, Metrics)
  - **Distributed Locks:** Redis 기반 분산 락 메커니즘 구현 및 최적화
  - **Logging:** Grafana Loki + Promtail 기반 통합 로깅 시스템 운영 경험 활용
  - **Metrics:** Prometheus 기반 메트릭 수집 및 실시간 알림 체계 강화
2. Gateway 운영 (Authentication, Security, Degrade)
  - **Authentication:** 기존 IAM, VPN, NAC 통합 인증 경험을 토스 Gateway에 적용
  - **Security Modules:** 15종 보안 솔루션 통합 경험을 활용한 보안 모듈 강화
  - **Degrade Features:** 고가용성 아키텍처 경험을 바탕으로 서비스 저하 기능 구현
3. 성능 개선 (Profiling, Hardware Acceleration)
  - **Profiling Tools:** Grafana Tempo, Pinpoint 등 APM 도구를 활용한 성능 병목 분석
  - **Performance Optimization:** DB 쿼리 튜닝 경험 (CPU 30% 개선) 활용
  - **Hardware Acceleration:** Redis, SSD 최적화 경험 적용

#### 4. 안정적인 서비스 운영

- **99.9% 가용성:** Production 환경에서 검증된고가용성 아키텍처 적용
- **MTTR 최소화:** 장애 대응 시간 40% 단축 경험을 토스 Platform에 기여
- **자동 복구:** AI 기반 장애 분석 및 자동 롤백 시스템 구축 경험 공유

#### 5. 빠른 기술 습득 및 적응

- **Kotlin/Spring Boot:** 공식 문서 및 실습을 통한 빠른 학습 (주 10시간)
- **Istio Service Mesh:** Kubernetes 기반 Service Mesh 아키텍처 이해 (주 5시간)
- **Kafka:** 메시징 시스템 설계 및 운영 학습 (주 3시간)

---

### ☑ 마지막 한 마디

토스의 **Server Developer (Platform)** 포지션은 제가 7년간 쌓아온 **Monitoring & Observability, Gateway 운영, 플랫폼 안정화** 전문성을 모두 발휘할 수 있는 최적의 자리라고 확신합니다.

**현재 강점:** - ☑ Prometheus, ELK Stack, Redis: 토스 요구사항 100% 충족 - ☑ Docker, Kubernetes POC: Container Orchestration 경험 - ☑ 99.9% 가용성, MTTR 70% 개선: 검증된 안정성

**빠른 학습 및 적응:** Kotlin, Spring Boot, Istio 등 토스 핵심 기술은 현재 학습 중이며, 기존 Flask/Node.js 경험을 바탕으로 빠르게 습득하고 있습니다. “**복잡한 인프라를 단순하게, 반복 업무를 자동화로**” 라는 철학으로 토스 Platform 팀의 **공통 서비스 제공, Gateway 운영, 안정적인 서비스 운영**에 기여하고 싶습니다.

감사합니다.

---

**제출일:** 2025년 9월 30일 **포트폴리오:** <https://resume.jclee.me> **GitHub:** <https://github.com/qws941>