

이재철 | Server Developer

Phone 010-5757-9592 Email qws941@kakao.com GitHub github.com/qws941 Portfolio resume.jclee.me

☒ Summary

7년간 대규모 인프라 운영에서 **Python 자동화로 반복 업무를 시스템화**하여 효율을 **50~95% 개선**했습니다. **초당 10만 이벤트**를 처리하는 실시간 플랫폼을 설계·운영하며, **예외 상황을 구조화**하고 안정적으로 처리하는 시스템 설계에 집중합니다.

핵심 역량

- Python, Node.js 기반 자동화 시스템 개발
- 대규모 이벤트 처리 (Kafka, Redis, Elasticsearch)
- Platform Engineering (Prometheus, Grafana, Docker, Kubernetes)
- 자동화 파이프라인 (Ansible, GitHub Actions)
- 데이터베이스 (PostgreSQL, MySQL, MongoDB, Redis)

주요 성과

- Python 자동화로 운영 업무 **50~95% 효율 개선**
 - 초당 10만 이벤트 처리 실시간 플랫폼 구축
 - **99.9% 가용성** 달성, **MTTR 70% 개선**
 - 1,000대 규모 인프라 **3일 내 자동 배포**
 - 금융권 규제 요건 **100% 충족** 시스템 설계
-

☒ Work Experience

(주)아이티센 CTS / (주)가온누리정보시스템

정보보안 운영 엔지니어 / 인프라 엔지니어 | 2024.03 ~ 현재 (1년 7개월)

넥스트레이드 (금융권 혁신금융서비스) 구축 및 운영

담당 업무

- 금융위원회 본인가 대비 보안 체계 구축 및 운영 안정화
- 15종 이상 보안 솔루션 통합 운영 및 자동화
- Python 기반 운영 자동화 시스템 개발 및 적용

- 보안 이벤트 모니터링 및 장애 대응 (24/7)

주요 성과

- Python으로 방화벽 정책 100건+ 자동 배포, 작업시간 50% 단축
 - 수작업 정책 배포를 Python 스크립트로 자동화
 - 예외 상황(중복 정책, 충돌) 자동 감지 및 처리
 - 배포 전 검증 로직 추가로 장애 0건 유지
- 보안 솔루션 충돌 해결로 단말 CPU 30% 개선
 - EPP/DLP 간 충돌 원인 분석 및 프로세스 우선순위 조정
 - 예외 처리 로직 구현으로 안정성 확보
- 보안 정책 재설계로 오탐률 50% 감소, 장애율 35% 감소
 - 운영 데이터 분석 기반 정책 최적화
 - 규칙 기반 자동 분류 시스템 구현
- 금융규제 요건 100% 충족, 본인가 심사 통과
- 내부정보 유출사고 0건 유지 (19개월)
- 금융감독원 정기 감사 지적사항 0건

Tech Stack Python, Ansible, Fortigate, NAC, DLP, APT, VPN

핵심 포인트: 금융권 규제 환경에서 자동화를 통해 복잡한 운영 프로세스를 시스템화하고, 예외 상황을 안정적으로 처리하는 구조 설계

(주)관텍투자일임

인프라·정보보호팀 인프라 엔지니어 | 2022.08 ~ 2024.03 (1년 7개월)

AI 기반 주식투자 서비스 인프라

담당 업무

- 금융보안데이터센터(FSDC) 운영 및 자동화
- 150대 이상 서버 및 스토리지 형상관리 자동화
- Python 기반 장애 예방 및 대응 시스템 개발
- AWS 클라우드 보안 구성 및 자동화

주요 성과

- Python 자동화로 장애율 40% 감소
 - 서버 상태 모니터링 자동화
 - 이상 징후 사전 감지 및 알림 시스템 구축

- DB 접근제어 쿼리 튜닝으로 CPU 30% 개선
- 금융감독원 정기 감사 통과, 개인정보 유출사고 0건

Tech Stack AWS, Python, Terraform, MySQL, PostgreSQL

(주)메타엠

인프라 엔지니어 | 2019.12 ~ 2021.08 (1년 8개월)

대규모 재택근무 환경 구축

담당 업무

- 1,000명 규모 VPN 재택근무 환경 긴급 구축
- Ansible 기반 대규모 자동 배포 시스템 개발
- Python 기반 점검 자동화 스크립트 개발

주요 성과

- **Ansible로 VPN 1,000대 자동 배포 (3일 내 완료)**
 - 수작업 시 2주 소요 → 자동화로 3일 단축
 - Playbook 기반 일관성 있는 배포 프로세스 구축
- VPN-백신 충돌 문제 해결, 장애 문의 40% 감소
- **Python 점검 스크립트로 점검시간 75% 단축**

Tech Stack Fortigate SSL VPN, NAC, Ansible, Python, Cisco

☒ Projects

REGTECH Blacklist Intelligence Platform

Production | 2024 ~ 현재 blacklist.jclee.me | github.com/qws941/blacklist

REGTECH 포털 자동 수집 및 위협 인텔리전스 플랫폼

Tech Stack Python Flask, PostgreSQL 15, Redis 7, Docker, GitHub Actions, Prometheus, Grafana

주요 성과 - 99.9% 가용성 달성 - MTTR 70% 개선 - 자동 롤백 시스템 (30초 내) - AI 기반 장애 분석

토스 관련성: 운영 업무 자동화 및 예외 상황 처리 시스템 설계

Splunk-FortiNet Integration Platform

Production | 2024 ~ 현재 splunk.jclee.me | github.com/qws941/splunk-fortinet

초당 10만 이벤트 처리 가능한 Splunk-FortiManager 통합 플랫폼

Tech Stack Python Flask, Splunk API, FortiManager JSON-RPC, Cloudflare Workers, Redis 7, Docker

주요 성과 - 초당 10만 이벤트 처리 검증 - 75,000% 확장 여유 확보 - 80대 방화벽 동시 관리 - API 응답 시간 < 100ms

토스 관련성: 대규모 이벤트 처리 및 실시간 시스템 설계

FortiGate Policy Orchestration Platform

Production | 2024 ~ 현재 fortinet.jclee.me | github.com/qws941/fortinet

FortiManager 정책 관리 자동화 플랫폼

Tech Stack Python Flask, FortiManager JSON-RPC, Docker, GitHub Actions

주요 성과 - 정책 검증 시간 80% 단축 - 서비스 중단 0건 - 감사 대응 시간 90% 절감

토스 관련성: 반복적인 정책 관리 업무를 자동화 가능한 흐름으로 구조화

Full-Stack Observability Platform

Production | 2024 ~ 현재 grafana.jclee.me | github.com/qws941/grafana

Prometheus + Grafana + Loki + Tempo 통합 모니터링 플랫폼

Tech Stack Prometheus, Grafana, Loki, Tempo, Docker Compose

주요 성과 - 11개 대시보드 운영 (121개 패널) - 초당 10만+ 로그 이벤트 처리 - 24/7 실시간 모니터링

토스 관련성: 시스템 안정성 확보 및 장애 대응 자동화

☒ Education

한양사이버대학교 | 컴퓨터공학과 | 2024.03 ~ 재학중

☒ Certifications

- CCNP (Cisco Certified Network Professional) | Cisco Systems | 2020.08
 - RHCSA (Red Hat Certified System Administrator) | RED HAT | 2019.01
 - CompTIA Linux+ | CompTIA | 2019.02
 - LPIC Level 1 | LINUX | 2019.02
-

☒ 토스 AI-자동화 포지션 적합성

요구사항 1: 반복적인 운영 업무를 자동화 가능한 흐름으로 정의하고, 시스템으로 구현

☒ Python으로 방화벽 정책 100건+ 자동 배포 (작업시간 50% 단축) ☒ Ansible로 VPN 1,000대 3일 내 자동 배포 ☒ Python 점검 스크립트로 점검시간 75% 단축

요구사항 2: 예외와 판단이 필요한 흐름을 안정적이고 유연하게 구성

☒ EPP/DLP 충돌 해결로 단말 CPU 30% 개선 ☒ 보안 정책 재설계로 오탐률 50% 감소 ☒ VPN-백신 충돌 문제 해결로 장애 문의 40% 감소

요구사항 3: 재사용 가능한 구조 설계, 도메인 지식과 기술 함께 고려

☒ 초당 10만 이벤트 처리 플랫폼 (75,000% 확장 여유) ☒ 99.9% 가용성 달성, MTTR 70% 개선 ☒ 금융규제 요건 100% 충족 시스템 설계

기술 스택 매칭

☒ Python, Node.js (핵심 자동화 언어) ☒ PostgreSQL, MySQL, MongoDB, Redis ☒ Docker, Kubernetes, Prometheus, Grafana ☒ Kafka, Elasticsearch (유사 기술 경험) ☒ Kotlin/Spring (학습 중)