

I. 자기소개 및 지원 동기

안녕하세요. 저는 인프라와 보안을 중심으로 5년 넘게 금융, 교육, IT 서비스 등 다양한 환경에서 시스템을 다듬고 안정화해온 이재철입니다. Linux, Unix, Windows 서버 운영 노하우와 AWS 기반 자동화 경험, 그리고 NAC, DLP, EDR 같은 보안 솔루션을 통합하며 효율성과 신뢰성을 동시에 챙겨온 점이 제 강점입니다.

Toss의 System Engineer 포지션에 끌린 이유는 단순히 인프라를 관리하는 데서 끝나는 게 아니라, **프로덕트의 가치를 극대화하는 시스템을 만드는 역할**에 있다는 점입니다. 대규모 트래픽을 안정적으로 처리하면서도 빠르게 혁신하는 Toss의 환경은, 제가 Kubernetes, CI/CD, IaC로 고가용성을 실현해온 경험과 잘 맞아떨어진다고 생각했습니다. 특히, 최근 대체거래소 매매체결시스템(넥스트레이드) 프로젝트와 Fortinet 방화벽 자동화 경험을 바탕으로, Toss의 빠르고 안전한 핀테크 서비스에 기여하고 싶습니다. SRE 관점에서 시스템의 신뢰도를 높이고, 팀과 협업하며 사용자에게 믿음을 주는 서비스를 만드는 데 기여하고 싶어 지원했습니다.

II. 경력 및 주요 성과

1. (주)가온누리정보시스템 | 프리랜서 인프라 엔지니어 (2024.03 ~ 2025.02)

- **업무:** 대체거래소 매매체결 시스템(넥스트레이드) 구축 프로젝트 참여하여 Legacy 환경에서 Linux/Windows 서버 운영, NAC·DLP 솔루션 관리, Fortinet FW 정책 자동화(Python활용)
- **성과:**
 - 넥스트레이드 방화벽 정책 자동화로 배포 시간 70% 단축
 - 넥스트레이드 프로젝트에서 DR 테스트 개선, 백업 성공률 95% → 99% 달성

2. (주)관텍투자일임 | 인프라·정보보호팀 엔지니어 (2022.08 ~ 2024.03)

- **업무:** AWS(EC2, RDS, S3, IAM) 기반 서버 운영, NAC 정책 튜닝, Python/Shell로 로그 관리 자동화
- **성과:**

- 단말 인증 실패율 30% 감소, 금융감독원 감사에서 유출 사고 0건 달성
- AWS 리소스 모니터링 체계화로 연간 비용 20% 절감

3. (주)편앤씨 | DevOps 엔지니어 (2022.05 ~ 2022.07)

- **업무:** Jenkins-GitLab CI로 CI/CD 파이프라인 구축, Docker 운영, Terraform으로 스테이징 환경 자동화
- **성과:**
 - 무중단 배포 구현, 배포 주기 7일 → 3일 단축
 - 장애 대응 시간 50% 줄여 서비스 안정성 강화

4. (주)조인트리 | 인프라·시스템 엔지니어 (2021.09 ~ 2022.04)

- **업무:** VMware/NSX-T 기반 차세대 시스템 구축, NAC·APT 솔루션 통합, LDAP 계정 관리 자동화
- **성과:**
 - 가상화 자원 최적화로 비용 15% 절감, 가동률 99.9% 유지
 - 장애 발생률 25% 감소

5. (주)메타넷애플랫폼 | 인프라·시스템 엔지니어 (2020.08 ~ 2021.08)

- **업무:** 1,000명+ 규모 재택 환경 구축(Fortigate VPN, NAC), Python으로 로그·백업 자동화
- **성과:**
 - 백업 작업 시간 주당 6시간 → 2시간 단축
 - EDR 충돌 이슈 해결로 장애 문의 40% 감소

6. (주)엠티데이터 | 서버·시스템 엔지니어 (2018.10 ~ 2019.10)

- **업무:** Unix/Windows 서버 운영, DR 설계, DB 접근제어 솔루션 초기 구축
 - **성과:**
 - DR 복구 성공률 90% → 99%로 개선
 - 보안 패치 처리 속도 30% 단축
-

Ⅲ. 보유 기술 및 자격

- 기술 스택:

- OS/서버: Linux, Unix, Windows Server
- 클라우드: AWS(EC2, RDS, S3, EKS), VMware
- 자동화: Terraform, Ansible, Python, Shell Script
- DevOps: Kubernetes, Docker, Jenkins, GitLab CI, Prometheus, ELK
- 보안: NAC, DLP, EDR, APT, Fortinet 방화벽, DB 접근제어

- 자격증: CCNP, RHCSA, LPIC1, 사무자동화산업기사

사례 1. 로그 관리 및 사고 대응 자동화

- 문제:

로그 관리와 사고 대응이 수동으로 이루어져 주당 약 6시간의 작업이 필요했습니다. 이는 팀의 생산성을 저하시키고, 사고 탐지 및 대응 속도가 느려 시스템 보안에 영향을 미쳤습니다.

- 해결 과정:

저는 Python과 Shell 스크립트를 사용하여 로그 수집, 분석, 사고 대응을 자동화했습니다.

- Python 스크립트를 작성하여 여러 서버에서 로그를 수집하고 파싱했습니다.
- Shell 스크립트를 통해 로그를 분석하여 잠재적인 보안 사고나 시스템 오류를 탐지했습니다.
- 자동 알림 시스템을 설정하여 사고 발생 시 즉각적인 대응이 가능하도록 했습니다.

- 사용 기술:

- Python: 로그 수집 및 분석
- Shell: 스크립트 작성 및 실행
- 모니터링 도구: Nagios

- **결과:**

- 주당 수동 작업 시간이 6시간에서 2시간 미만으로 단축되었습니다.
- 사고 탐지 및 대응 속도가 향상되어 시스템 보안과 신뢰성이 강화되었습니다.
- 팀은 더 전략적인 작업에 집중할 수 있게 되어 전체적인 생산성이 향상되었습니다.

- **프로젝트 전후 개선된 점:**

- 이전: 수동 프로세스로 시간이 많이 소요되고, 사고 대응 지연
- 이후: 자동화로 시간 절약, 빠르고 정확한 사고 대응, 보안 강화

사례 2: Fortinet 방화벽 정책 자동화

- **문제:**

수동으로 관리하던 Fortinet 방화벽 정책은 시간이 많이 걸리고 오류가 잦았습니다. 정책 배포 과정에서 인적 오류로 인해 네트워크 다운타임이 발생하고, 보안 취약점이 노출될 위험도 있었습니다.

- **해결 과정:**

저는 Python과 Ansible을 사용하여 방화벽 정책 배포를 자동화하는 솔루션을 개발했습니다.

- Python 스크립트를 사용하여 정책 구성 파일을 파싱하고 유효성을 검증했습니다.
- Ansible을 활용해 정책을 Fortinet 방화벽에 자동으로 배포하도록 설정했습니다.
- 자동화 프로세스를 테스트하고, 배포 전후의 성능을 모니터링했습니다.

- **사용 기술:**

- Python: 정책 파싱 및 유효성 검증
- Ansible: 자동화 배포 및 구성 관리
- Fortinet 방화벽: 네트워크 보안 장비

- **결과:**

- 정책 배포 시간이 70% 단축되었습니다.
- 운영 오류가 50% 감소하여 네트워크 다운타임이 줄어들고, 보안 신뢰성이 향상되었습니다.
- 이 자동화는 네트워크 보안과 성능을 동시에 개선하여 팀의 효율성을 크게 높였습니다.

- **프로젝트 전후 개선된 점:**

- 이전: 수동 배포로 시간이 오래 걸리고 오류 빈도 높음
- 이후: 자동화로 빠르고 신뢰성 있는 정책 배포, 보안 강화

IV. 입사 후 포부

Toss에서 제가 그리고 싶은 그림은 세 가지입니다.

첫째, **대규모 트래픽을 견디는 인프라를 만듭니다.** Kubernetes(EKS)와 Terraform으로 스케일링과 자동화를 극대화하고, 넥스트레이드 같은 고가용성 시스템 경험을 살려 안정적인 운영 기반을 다지고 싶습니다.

둘째, **보안과 속도의 조화를 이루겠습니다.** Fortinet 자동화처럼 보안 정책을 효율적으로 관리하며, CI/CD에 보안 점검을 녹여 DevSecOps를 실현하겠습니다.

셋째, **데이터로 문제를 앞서 해결하겠습니다.** Prometheus와 ELK로 실시간 모니터링을 강화하고, 장애의 근본 원인을 파악해 재발을 막는 시스템을 정착시키고 싶습니다.

Toss의 빠르고 안전한 프로덕트를 뒷받침하는 엔지니어로서, 팀과 함께 사용자에게 신뢰를 주는 서비스를 만들겠습니다. 감사합니다.