

PROJECT 01. 인프라 보안 구축 (Drop The Port!)

이름: 허준혁

기간: 2025.08 - 2025.11 (4개월)

인원: 4명 (팀 프로젝트)

역할: 네트워크 엔지니어

기여도: 25%

💡 3줄 요약

- ✓ 실제 기업 환경을 모사하여 **DMZ/Internal Zone**이 분리된 보안 인프라 설계
- ✓ 방화벽 정책과 접근 제어(ACL)를 통해 **최소 권한 원칙**의 보안 환경 구현
- ✓ ELK Stack을 구축하여 분산된 로그를 중앙에서 **실시간 관제 및 시각화**

🔧 사용 기술 (Tech Stack)

Network (Cisco C3650)

Firewall (AhnLab TrusGuard 50B)

Server (Ubuntu 22.04, Win11)

Monitoring (ELK Stack, Wireshark)

Virtualization (Docker)

WEB/DB (Nginx, PHP, PostgreSQL)

💻 핵심 문제 해결 (Problem & Solution)

Q. 단일망 사용으로 인한 보안 취약성 문제

A. 망 분리 및 계층적 보안 아키텍처 적용

L3 스위치와 방화벽을 연동하여 DMZ(Web), Internal(DB/Log), 관리자망을 물리/논리적으로 격리하고 **VLAN**으로 트래픽 분산 처리

Q. 비인가 외부 접속 및 내부 정보 유출 위험

A. 강력한 접근 제어 및 NAT 정책 수립

기본 정책을 'Deny All'로 설정 후, 웹 서비스(80/443) 등 **필수 포트만 허용(Whitelist)**하고, **NAT**를 적용하여 내부 IP 구조 은닉

Q. 장비별 로그 분산으로 인한 장애 대응 지연

A. 중앙 집중형 로그 통합 모니터링 구축

Rsyslog와 Filebeat로 모든 장비의 로그를 수집 서버로 전송하고, **ELK Stack(Kibana)** 대시보드에서 위험 탐지 시각화 구현

🚀 성과

- ✓ 모의 해킹(악성 IP 접속, SSH Brute Force) 시도 시 방화벽 및 서버 iptables에서 **즉각 차단 성공**
- ✓ 로그 중앙화를 통해 장애 발생 시 **원인 파악 시간 단축 및 가시성 확보**

PROJECT 02. 마인크래프트 클러스터링 (네모의 꿈)

이름: 허준혁

기간: 2025.08 - 2025.10 (3개월)

인원: 2명 (팀 프로젝트)

역할: 팀장

기여도: 50%

💡 3줄 요약

- ✓ 단일 서버 과부하 해결을 위해 **Velocity** 프록시 기반 **3-Node** 클러스터링 구축
- ✓ 라즈베리파이 클러스터와 **다층 방어 모델(격리-인증-감시)**로 보안성/효율성 강화
- ✓ CLI 관리의 어려움을 해소하고자 **Python** 기반 통합 관리 **GUI** 도구 자체 개발

🔧 사용 기술 (Tech Stack)

Core App (Minecraft Paper)

Infrastructure (Raspberry Pi 5)

Network (Velocity Proxy)

Database (MariaDB)

Management (Python GUI)

API (Discord)

💻 핵심 문제 해결 (Problem & Solution)

Q. 사용자 증가에 따른 단일 서버 과부하

A. 프록시 기반 부하 분산(Load Balancing)

Velocity 프록시를 도입하여 오버월드/네더/관리 서버로 역할을 분리하고, 3대의 라즈베리파이에 트래픽을 분산시켜 성능 최적화

Q. 무분별한 비인가 접속 및 테러 위험

A. MFA 도입 및 Zero Trust 기반 다층 방어 모델 적용

1차: 대기실 격리 → 2차: 화이트리스트 검증 → 3차: 디스코드 연동 2FA(이중 인증) 완료 시에만 게임 서버 진입 허용

Q. 복잡한 리눅스 CLI 명령어로 인한 관리 비효율

A. Python GUI 관리 도구 (BlockOps) 개발

서버 상태(CPU/RAM) 실시간 모니터링, 원클릭 백업, 화이트리스트 관리가 가능한 **GUI 프로그램**을 개발하여 운영 편의성 증대

🚀 성과

- ✓ 자체 개발 인증 시스템 적용 후 비인가 접속 시도 원천 차단 및 데이터 안정성 확보
- ✓ 게임 서버(오버월드)의 트래픽과 로그 처리를 분리하고, 플레이어의 행동 데이터를 관리 서버로 실시간 전송·DB화하여 중앙 집중형 데이터 관리 환경 구현