

클라우드 기초_최종정리본

📁 과목(Class)	클라우드 기초
🔍 키워드	

클라우드 기초 (Cloud Computing & Docker)

학습 개요

- **기간:** 2025.03 ~ 2025.07
- **환경:** Rocky Linux 9.0, Ubuntu, Docker CE, Docker Compose
- **핵심 역량:** 클라우드 보안 아키텍처 이해, Docker 컨테이너 인프라 설계 및 IaC 기반 배포 자동화

☁️ 상세 학습 내용 (Technical Skills & Concepts)

1. 클라우드 아키텍처 및 보안 (Cloud Architecture & Security)

- **서비스 모델 분석:** IaaS, PaaS, SaaS의 차이점 이해 및 ****책임 공유 모델(Shared Responsibility Model)****에 따른 보안 관리 포인트(IAM, 데이터 암호화 등) 식별.
- **계층형 보안 설계:** 보안성, 가용성을 고려한 **계층 분리(Tiered Architecture)** 원칙 학습 및 DB 직접 접근 차단을 통한 공격 표면(Attack Surface) 최소화 전략 수립.

2. 컨테이너 엔지니어링 (Container Engineering)

- **이미지 관리 전략:** `commit` (수동), `export/import` (파일시스템 추출), `save/load` (레이어 백업) 방식의 차이점 이해 및 상황별(백업 vs 이관) 이미지 관리 전략 적용.
- **Dockerfile 최적화:** `FROM`, `RUN`, `COPY` 지시어를 활용한 인프라 코드화 및 **Multi-stage Build** 기법을 적용하여 이미지 용량 경량화와 보안성 강화.

3. 스토리지 및 네트워크 인프라 (Storage & Networking)

- **데이터 영속성 확보:** 데이터 보존을 위한 **Docker Volume**과 개발 환경(소스 실시간 반영)을 위한 **Bind Mount**의 차이점을 이해하고 적재적소에 활용.
- **클러스터 스토리지:** NFS를 연동하여 멀티 호스트 간 데이터 공유 환경을 구축하고, 컨테이너 이동성(Mobility)을 보장하는 스토리지 아키텍처 구현.

4. 인프라 자동화 (Orchestration Basics)

- **Docker Compose:** `docker-compose.yml` 작성을 통해 웹(Web)과 데이터베이스(DB) 등 다중 컨테이너 환경을 코드로 정의(IaC)하고, 의존성에 맞춰 일괄 배포 및 관리.

실전 트러블슈팅 (Troubleshooting Log)

Case 1. 이미지 삭제 오류와 의존성 관리

- **문제:** `docker rmi` 명령어로 이미지를 삭제 시도했으나, "image is being used by running container" 에러 발생.
- **해결:** 이미지가 실행 중인 컨테이너에 의존성이 있음을 확인. `docker stop` → `docker rm` → `docker rmi` 순서로 라이프사이클을 준수하여 리소스를 안전하게 정리함.

Case 2. NFS 권한 문제와 보안 적용

- **문제:** NFS로 공유된 볼륨에 컨테이너가 데이터를 쓰지 못하는 'Permission Denied' 발생.
- **해결:** 실습 단계에서는 `chmod 777` 로 해결했으나, 실제 운영 환경에서는 **UID/GID 매핑** 이나 **Root Squash** 옵션을 통해 최소 권한 원칙을 적용해야 함을 보안적 관점에서 분석함.

Case 3. Dockerfile 빌드 효율화

- **문제:** 소스 코드가 조금만 바뀌어도 이미지 빌드 시간이 오래 걸리는 현상.
- **해결:** 변경이 잦은 소스 코드 복사(`COPY`) 명령어를 변경이 적은 라이브러리 설치(`RUN`) 명령어보다 뒤에 배치하여 **Docker Layer Caching** 기능을 극대화함.

학습 회고 (Learning Reflection)

"기존의 무거운 가상머신(VM) 방식에서 벗어나, 컨테이너 기술을 통해 리소스를 효율적으로 사용하는 법을 체득했습니다. 특히 수동으로 서버를 구축하던 방식에서 Dockerfile 과 Docker Compose를 활용해 '**인프라를 코드로 관리(IaC)**'하는 방식으로 전환하며 운영 효율성을 경험했습니다. 또한, NFS 스토리지 연동과 네트워크 설계를 직접 수행하며, 향후 DevSecOps 파이프라인 구축 시 보안을 어떻게 내재화할지에 대한 기술적 토대를 마련했습니다."