

4강. IaaS 필수 분석 : 컨테이너 인프라 분석





목차

- ✓ 리눅스 컨테이너 기술 분석
- ✓ LXC vs Docker
- ✓ 문제

리눅스 컨테이너 기술 분석



가상화 분야

데이터 가상화

다수의 이기종 데이터를 단일한 소스로 접근 가능

네트워크 기능 가상화

NFV (Network Functions Virtualization)

서버 가상화

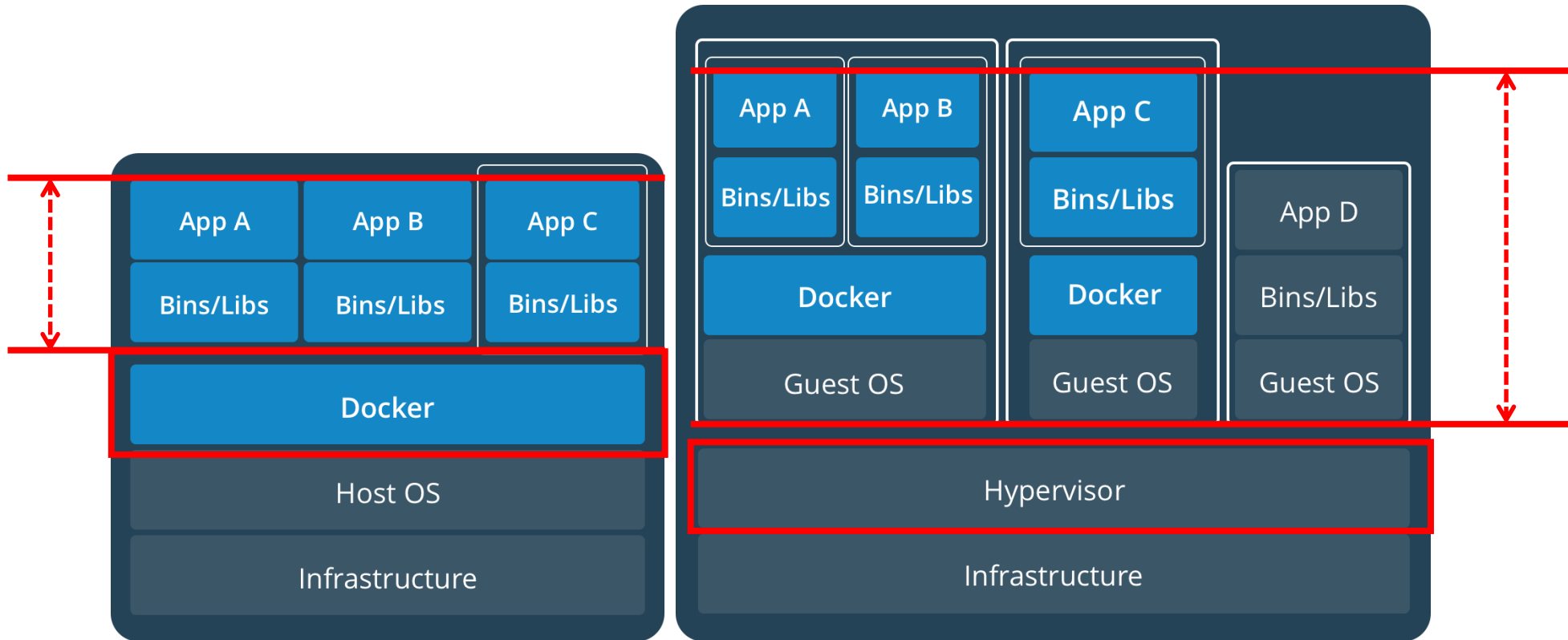
하이퍼바이저 (Hypervisor)

운영체제 가상화

컨테이너 (Container)



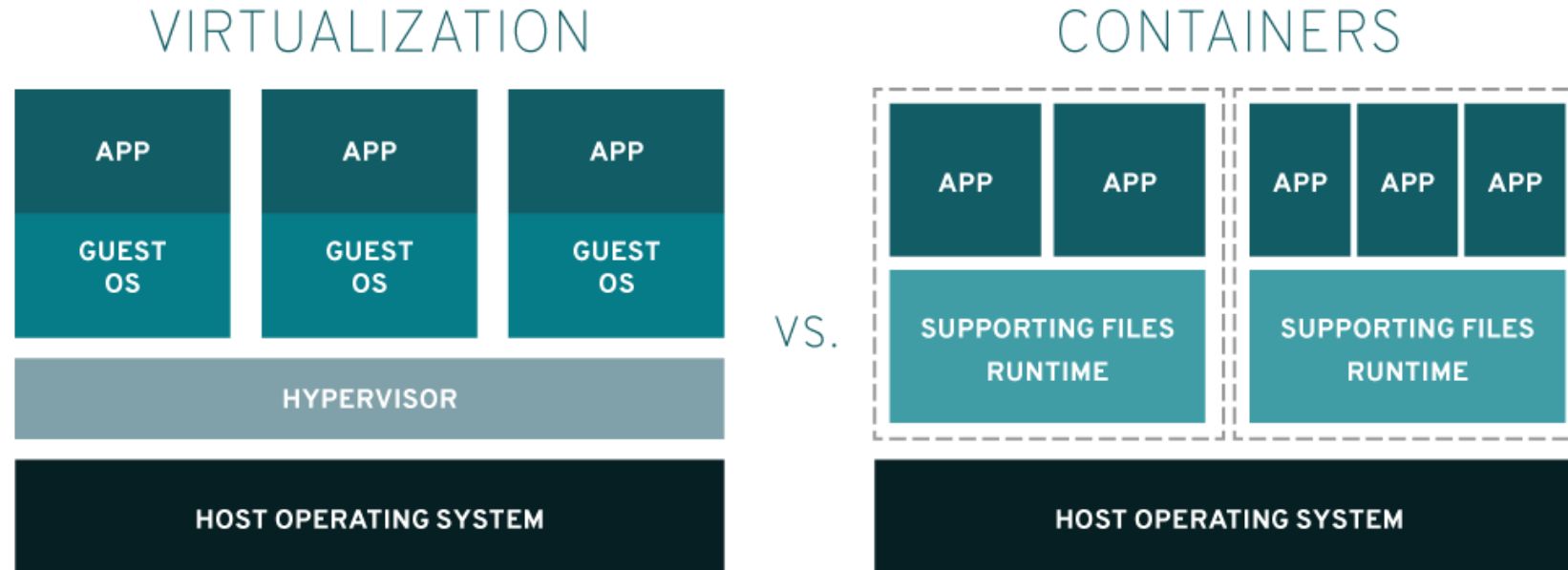
하이퍼바이저 vs 리눅스 컨테이너



※ 이미지 출처 : Docker



하이퍼바이저 vs 리눅스 컨테이너



※ 이미지 출처 : 레드햇

하이퍼바이저

단일 시스템에서 여러 운영체제가 동시에 실행

리눅스 컨테이너

단일 커널에서 여러 컨테이너가 동시에 실행

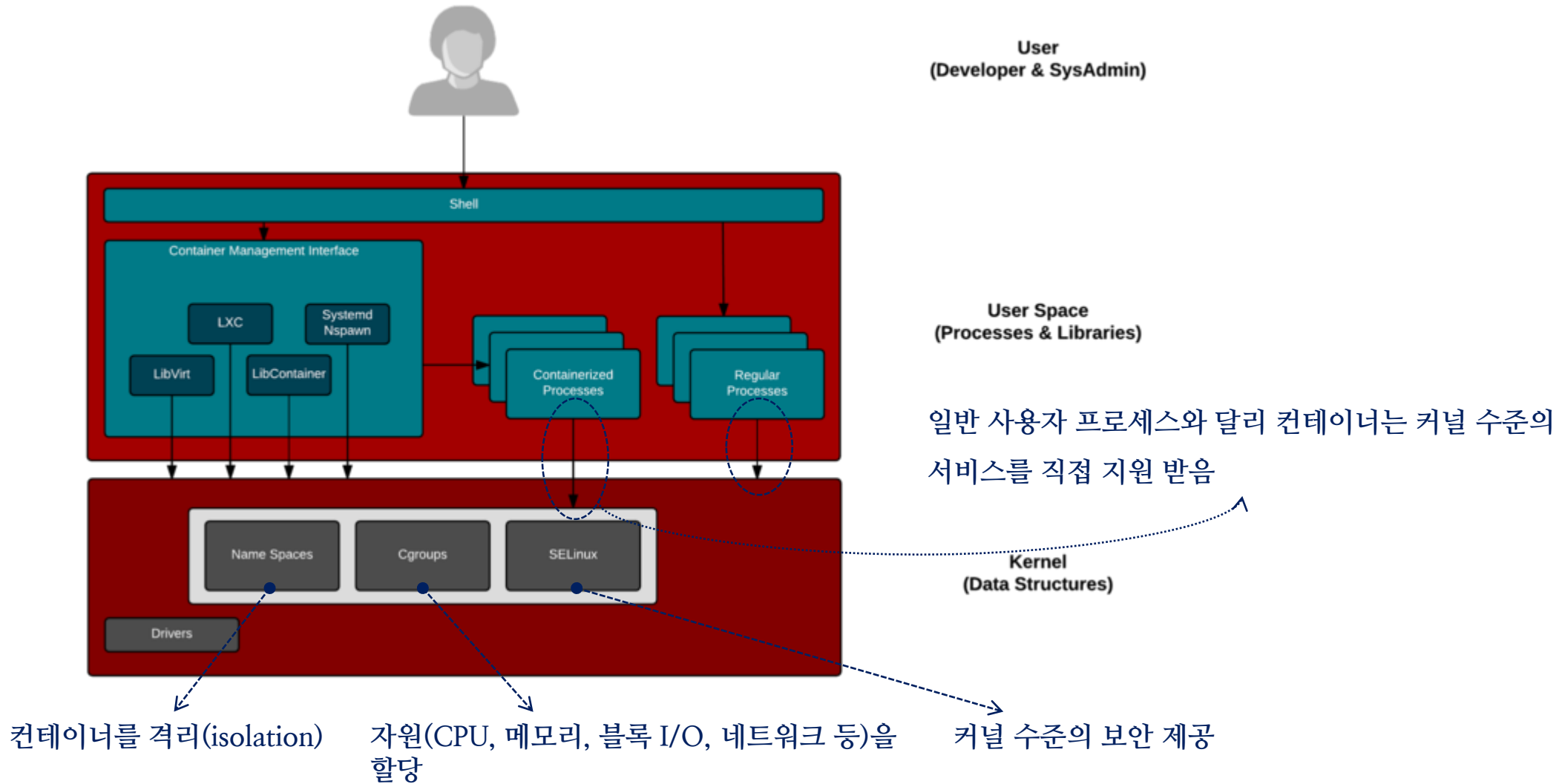


리눅스 컨테이너

- 사용자 프로세스 가상화 기술(Lightweight process virtualization)은 새로운 것이 아님
 - ✓ FreeBSD Jails
 - ✓ Solaris Zones
 - ✓ AIX WPARs (Workload Partitions)
 - ✓ Linux-based containers projects : 예) Linux-VServer (linux-vserver.org)
- 왜 요즘 뜨는가?
 - ✓ 리눅스 커널 레벨에서 컨테이너 기능을 지원하기 시작했기 때문



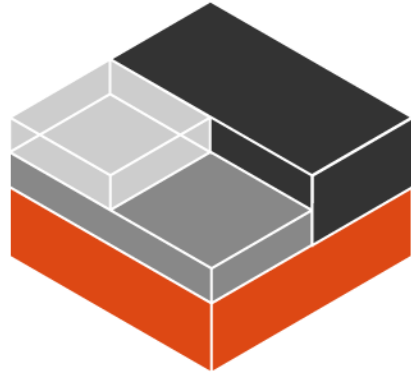
리눅스 컨테이너



LXC vs Docker



LXC (Linux Containers)



What's LXC?

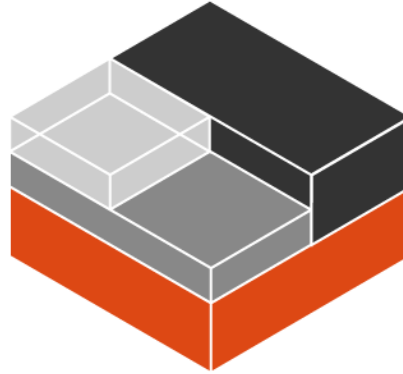
LXC is a userspace interface for the Linux kernel containment features.

Through a powerful API and simple tools, it lets Linux users easily create and manage system or application containers.

※ 출처 <https://linuxcontainers.org/lxc/introduction/>



LXC (Linux Containers)



- Components (구성요소)

- ✓ The “liblxc” library
- ✓ Several languages bindings for the API
 - : python3, lua, Go, ruby, python2, Haskell
- ✓ A set of standard tools to control the containers
- ✓ **Distribution container templates**
 - : lxc-busybox, lxc-centos, lxc-Debian, lxc-opensuse, lxc-ubuntu

※ 출처 <https://linuxcontainers.org/lxc/introduction/>



LXC vs LXD

What's LXD?

LXD is a next generation system container manager. It offers a user experience similar to virtual machines but using Linux containers instead.

It's image based with pre-made images available for a wide number of Linux distributions and is built around a very powerful, yet pretty simple, REST API.

※ 출처 <https://linuxcontainers.org/lxd/introduction/>



LXD Features

- Secure by design (unprivileged containers, resource restrictions and much more)
- Scalable (from containers on your laptop to thousand of compute nodes)
- Intuitive (simple, clear API and crisp command line experience)
- Image based (with a wide variety of Linux distributions published daily)
- Support for Cross-host container and image transfer (including live migration with CRIU)
- Advanced resource control (cpu, memory, network I/O, block I/O, disk usage and kernel resources)
- Device passthrough (USB, GPU, unix character and block devices, NICs, disks and paths)
- Network management (bridge creation and configuration, cross-host tunnels, ...)
- Storage management (support for multiple storage backends, storage pools and storage volumes)

※ 출처 <https://linuxcontainers.org/lxd/introduction/>

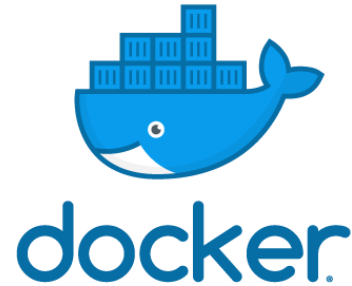
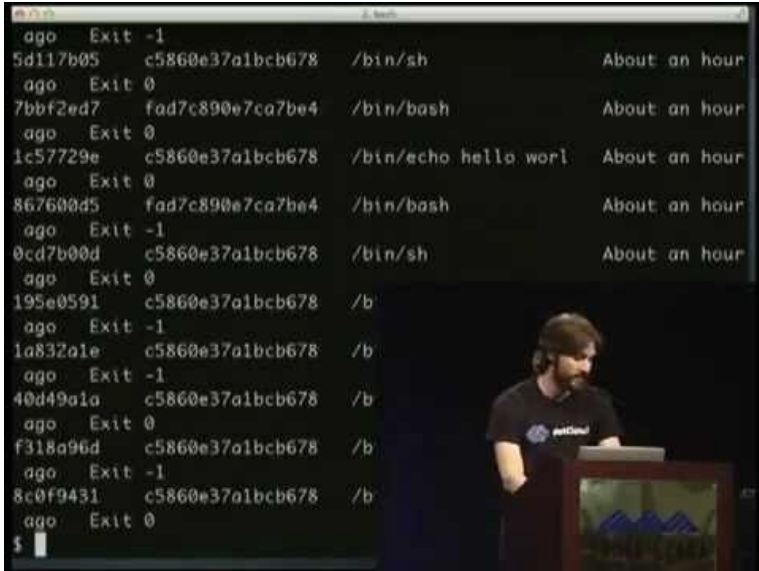


LXC vs LXD

- LXC는 컨테이너를 관리하는 명령어 툴
- LXD는 liblxc 를 통해 컨테이너를 관리하는 데몬 서비스



도커의 역사



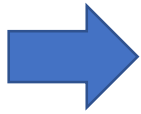
- 2013.3. Python Conference
dotCloud 창업자, Solomon Hykes,
“The future of Linux Containers” 세션에서 처음 발표
- 2013.10, 회사 이름을 Docker Inc. 변경
- 2014.6, Docker 1.0 발표



OS 가상화 -> App 가상화

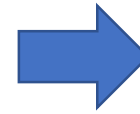
OS 전체를 가상화하는 방법
(사용이 간단하나 무겁고 느림)

- VMware
- VirtualBox



OS 전체를 가상화하지 않는 방법
(성능 개선으로 클라우드 컴퓨팅 기반이 됨)

- KVM (CPU 가상화 기술)
- Xen (반가상화 기술)

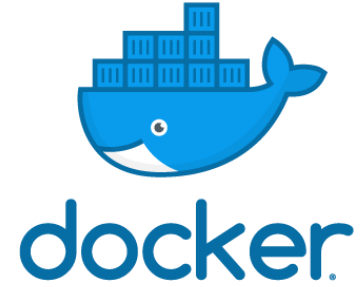
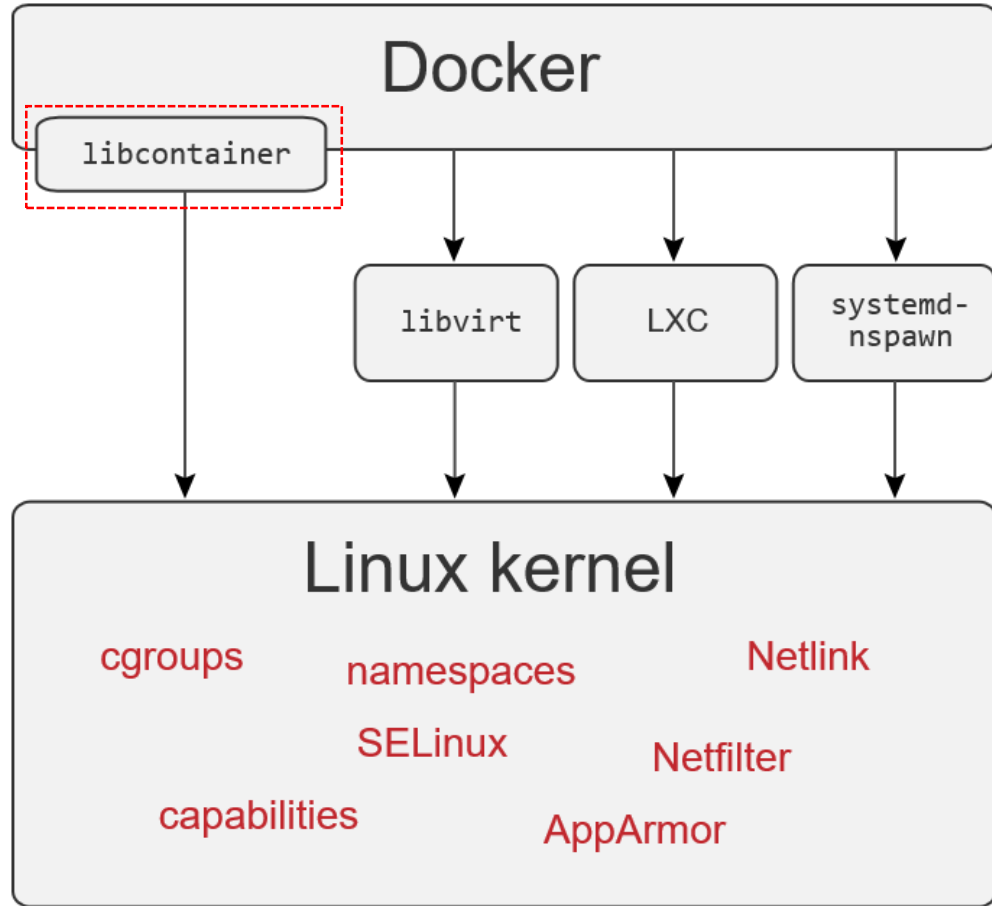


프로세스를 격리하는 방법
(가볍고 빠름)

- LXC
- Docker



도커 내부 구조

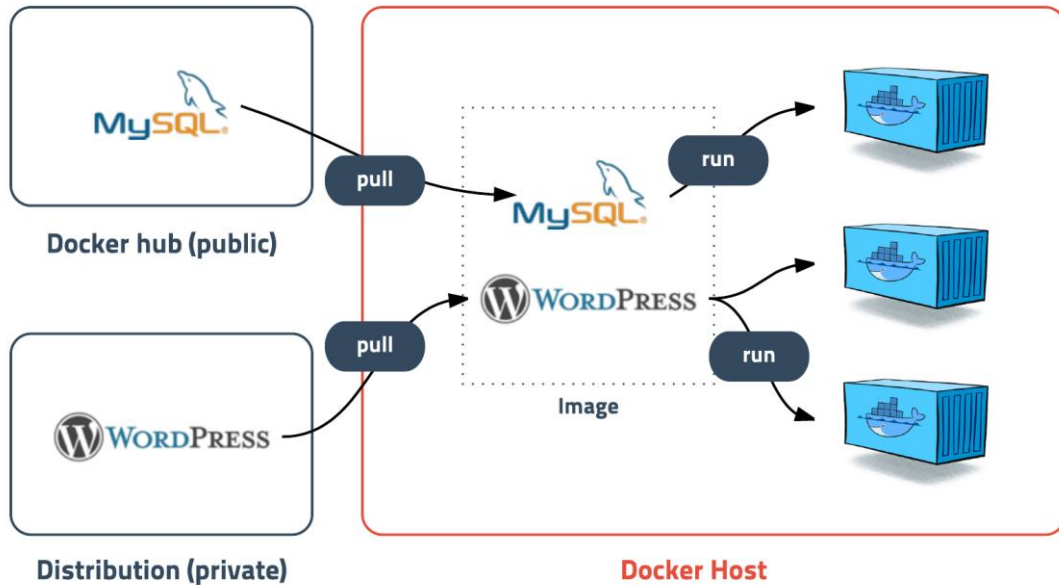


- 도커 버전 0.9 이전까지는 LXC 사용, 0.9 버전부터 “libcontainer” 사용



도커, 이미지(Image)

도커 이미지



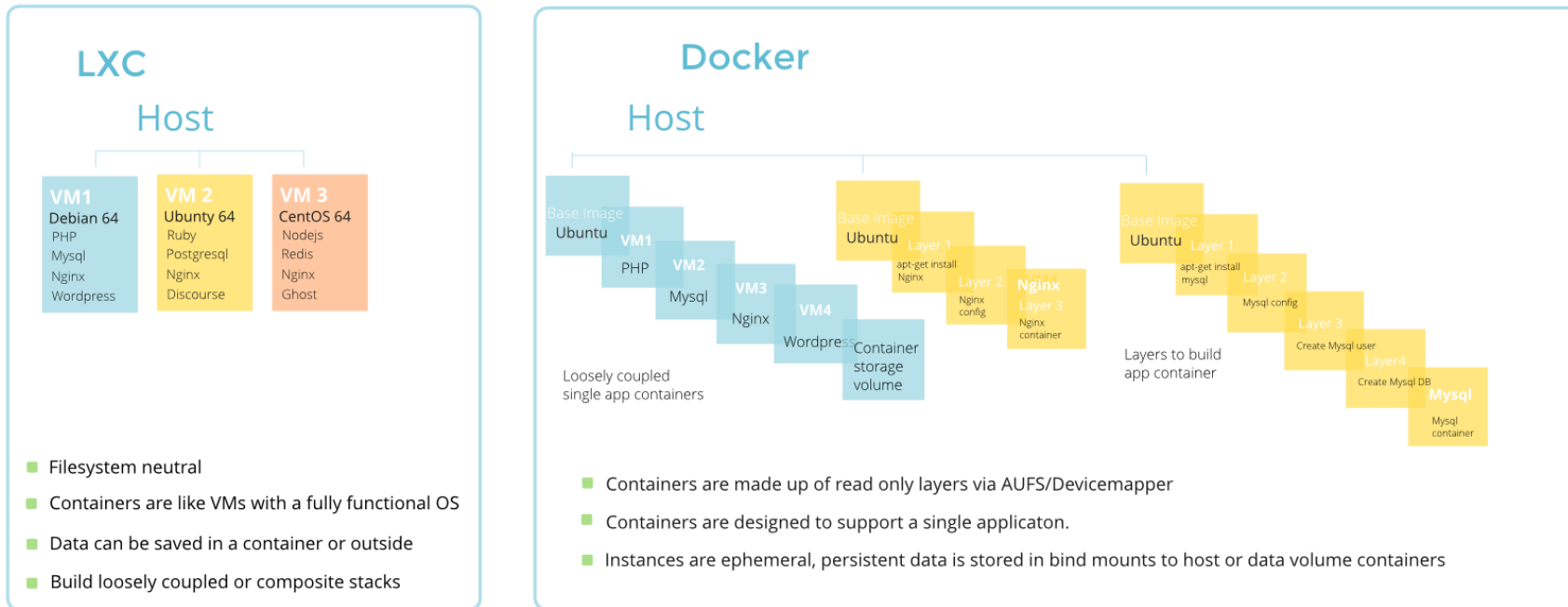
- 컨테이너 실행에 필요한 파일과 설정 값등을 포함
- 컨테이너는 이미지를 실행한 상태
- 변경할 수 없는 것 (immutable)
- 추가되고 변경되는 값들은 컨테이너에 저장됨
- 같은 이미지에서 여러 개의 컨테이너 생성 가능
- 컨테이너가 삭제되더라도 이미지는 그대로
- Docker Hub에 등록하거나 Docker Registry를 구축하여 관리 가능



LXC vs Docker

flockport

Key differences between LXC and Docker



flockport.com

- 도커는 컨테이너가 단일 프로세스로 실행, LXC 컨테이너는 “init” 프로세서가 있어 여러 프로세스 실행 가능

※ 이미지 출처 : <https://archives.flockport.com/lxc-vs-docker/>

복습 문제



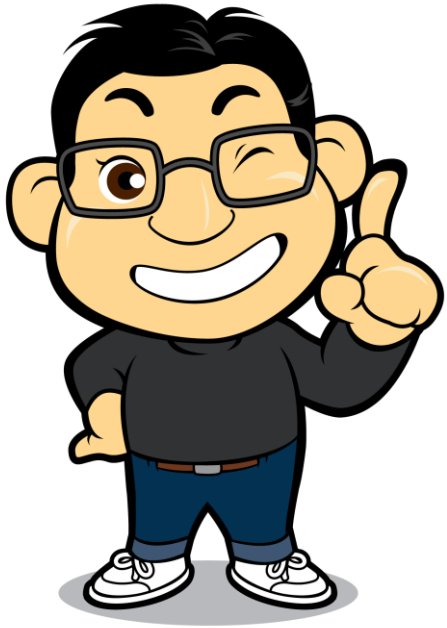
4강. 복습 문제

문제 <1> 하이퍼바이저 대신 컨테이너가 등장한 배경은?

문제 <2> OS 가상화와 App 가상화(격리)의 차이점은?

문제 <3> LXC 와 Docker 의 가장 큰 차이점은?

감사합니다!



엑세스랩(주), 유 명 환 funfun.yoo@gmail.com