

Dockek 01

[도커 기본]

따라하며 배우는 도커

John Ahn

2023.01.10

도커를 쓰는 이유

어떠한 프로그램 다운 받는 것을 간단하게 만들기 위한 방법

프로그램 다운 받는 방식이 서버, 패키지 버전, 운영체제에 따라 달라진다.

그렇다면 도커란?

- 컨테이너를 사용하여 응용프로그램을 더 쉽게 만들고 배포하고 실행할 수 있도록 설계된 도구이며 컨테이너 기반의 오픈소스 가상화 플랫폼이며 생태계이다.

컨테이너??

프로그램, 실행환경을 컨테이너로 추상화하고 동일한 인터페이스를 제공하여 프로그램의 배포 및 관리를 단순한게 해주는 것, 프로그램을 손쉽게 이동 배포하게 해준다.

도커 이미지 & 컨테이너

컨테이너 이미지 ?

- 코드, 런타임, 시스템 도구, 시스템 라이브러리 및 설정과 같은 응용 프로그램을 실행하는데 필요한 모든 것을 포함하는 가볍고 독립적인 실행 가능한 소프트웨어 패키지

도커이미지가 프로그램을 실행할때 필요한 설정, 종속을 가지고 있다.

도커 사용

1. 도커 CLI 커맨드 입력한다
2. 도커 서버 (도커 Daemon)이 그 커맨드를 받아서 그것에 따라 이미지를 생성하든 컨테이너를 실행하든 모든 작업을 하게된다.

도커 CLI → 도커 서버에서 필요한 것을 가져오거나 실행함.

컨테이너 (가상화 기술에 관하여)

한대의 서버 하나의 용도로만으로 사용 → 남는 공간을 **VM** 가상 환경 서버 이용가능 (하이퍼 바이저) - [호스트 시스템에서 다수의 게스트 OS를 구동할 수 있게 하는 소프트 웨어]

- 멀티 OS 사용하기 위해 네이티브 하이퍼 바이저 사용하거나
- VM으로 사용하는 것은 호스트형 하이퍼 바이저로 사용

하이퍼 바이저 vs 도커

공통점

- 도커 컨테이너와 가상 머신은 기본 하드웨어에서 격리된 환경 내에서 애플리케이션을 배치하는 방법

차이점

- 어플리케이션 실행 할 때 도커 → 호스트 OS에 실행 패키지인 이미지만 배포하면 된다
- 반대로 VM은 내부에서 OS가 실행되어야 한다.

가상 머신은

각 os별로 독립적으로 돌아가기 때문에 격리 되어있다.

도커는

도커는 호스트 OS에 실행 이미지를 배포하고 실행 하기 때문에 동일한 커널을 공유한다.

그러면 도커 컨테이너 격리는 어떻게 진행되나?

리눅스의 Cgroup/ 네임스페이스에 대해 알아야 한다.

Cgroup 다른 프로세스 사이에서 벽을 만드는 리눅스 커널 기능들이다.

너무많은 사용량을 가지는 어플이 있다면 CPU사용량을 제한 할 수 있는 기능

네임스페이스

하나의 시스템에서 프로세스를 격리시킬 수 있는 가상화 기술 - 경량 프로세스 가상화 기술

이미지로 컨테이너를 만드는 과정

이미지는 응용프로그램을 실행하는데 필요한 모든 것을 포함고있다.

1. 시작시 실행 될 명령어
2. 파일 스냅샷

→ 두가지를 이미지가 가지고 있다. 파일 스냅샷은 디렉토리나 파일을 카피 한것.

이미지로 컨테이너를 만드는 순서

1. Docker 클라이언트에 `docker run <이미지>` 입력
2. 파일 스냅샷을 컨테이너 하드 디스크 부분에 설치 함
3. 시작시 실행 될 명령어를 컨테이너 에서 실행
4. 커널을 통해 카카오톡 실행 파일을 실행 시킴.

이미지 → 컨테이너에 설치하는 환경 조성 & 실행 까지 저장되어 있는것.

커널은 운영체제 중 항상 메모리에 올라가 있는 운영체제의 핵심 부분으로서 하드웨어와 응용 프로그램 사이에서 인터페이스를 제공하는 역할을 하며 컴퓨터 자원들을 관리하는 역할을 한다. 즉, 커널은 인터페이스로써 응용 프로그램 수행에 필요한 여러가지 서비스를 제공하고, 여러가지 하드웨어(CPU, 메모리) 등의 리소스를 관리하는 역할을 한다.

Cgroup과 네임스페이스를 컨테이너에서 사용할 수 있는 이유

네임스페이스(프로세스를 격리시는 것) Cgroup (자원사용량을 격리 시킬 수 있는 기능)

`docker version` 치면

OS/Arch: linux/amd64 가 나온다

즉, 도커가 실행 될 때

Windows 위에 리눅스 VM에서 실행되는 것.
