도커 2020253091\_김호중

도커는 리눅스 컨테이너에 여러 기능을 추가함으로써 애플리케이션을 컨테니어 로서 좀더 쉽게 사용할 수 있게 만들어진 오픈소스입니다. 도커에 대해 살펴보기 전 하이퍼바이저에 대해 간략하게 살펴보겠습니다. 여러 운영체제를 하나의 호스트에서 생성해 사용할 수 있게 해준 기술이로 가상머신을 생성하고 실행하는 프로세스입니다. 다시 말해 단일 하드웨어에서 여러 다른 가상 머신을 호스팅할 수 있는 프로그램입니다. 하이퍼바이저에는 타입 1 타입 2가 존재하는데요. 타입 1부터 살펴보면 하이퍼바이저가 하드웨어에 직접 실행되며 게스트 os는 하드웨어에서 level 2로 실행됩니다. 타입 2는 일반적인 상황과 같이 os에서 실행되며 vm내부에서 실행되는 게스트 운영체제는 level 3에서 실행됩니다.

다음은 하이퍼바이저와 도커의 차이점에 대해서 알아보겠습니다. 하이퍼 바이저느 각종 시스템을 가상화하고 독립된 공간을 생성하기에 일반 호스트에 비해 성능의 손실이 발생합니다. 그뿐만 아니라 vm은 게스트 운영체제를 사용하기 위한 라이브러리 커널 등 전부 포함하기에 이미지의 크기또한 커지게되죠. 이런 문제를 해결 할 수 있는 스택이 바로 도커입니다. 도커는 가상화된 공산을 생성함에 있어 리눅스 자체기능을 사용하고 프로세스 단위의 격리 환경을 만들기에 성능의 손실이 거의 없습니다. 컨테이너에 필요한 커널은 호스트의 커널을 공유해주고 컨테이너 안 프로세스 에 필요한 라이브러리 및 실행 파일만 존재하기 때문에 이미지의 크기 또한 작죠.

커널에 대해서 모르시는 분이 계실 수 있으니 간략하게 개념만 잡고 가겠습니다. 커널은 메모리 관리 프로세스 관리 시스템 호출 및 보안 장치드라이버 등의 역할을 수행하며 os에서 핵심적인 역할을 맡은 파츠입니다. 쉽게 말해 커널은 하드웨어와 스포트웨어 사이를 이어주는 하나의 파츠죠 입출력을 관리하고 사용자의 요청을 하드웨어가 처리할 수 있도록 역할을 분배하는 것이 커널의 존재이유입니다.

다음은 리눅스에서 제공하는 namespace cgroup chroot 에 대해 알아 보겠습니다. 먼저 chroot라고 불리는 change root directory는 말그대로 루트 디렉토리를 바꾼다는 뜻으로 트리구조에서 새로운 루트노드를 만들어 사용한다 생각하시면 편합니다. 그림을 보면 알 수 있듯이 이런 동작과정으로 process k의 루트는 더 이상 /가 아니게 되며 k의 루트 디렉토리는 더 이상 /와 B,C에 접근 할 수 없습니다. Namespace는 프로세사나 프로세스 그룹별로 자원을 불리하는 방법으로 하드웨어 뿐만 아니라 소프트웨어 요소에도 적용됩니다. 네트워크 조회 등에서도 사용되죠 Cgroup은 프로세스가 사용할 수 있는 자원의 양을 제한합니다..

도커의 기본적인 기능들과 개념에 대해서 살펴보겠습니다. 우선 이미지입니다. 이미지는 도커에서 서비스 운영에 필요한 서버프로그램 소스코드 및 라이브러리 컴파일된 실행파일을 묶는 형태를 이미지라 합니다. 도커 이미지는 가상 머신의 이미지와 달리 커널을 포함하지 않고 있기에 이미지 크기가 그다지 크지 않습니다. 보통 100mb-nGB정도의 용량으로 vm에 비하면 굉장히 작죠

다음은 레이어 입니다.

레이러는 기존이미지에 추가적인 파일이 필요할시에 해당파일을 추가하기 위한 개념으로 이미지레이어는 읽기 속성만 가지고 있습니다 read only layer. 도커를 실행할떄 이미지의 최상단에 컨테이너 레이어가 쌓이게되며 이 레이어는 읽고 쓰기가 가능하며 Read write Layer라고 합니다. 컨테이너가 종료되어도 메모리에서 삭제되지 않고 남아있으며 컨테이너를 삭제하는 경우 함께 모두 사라집니다. 여기에서 말하는 도커 컨테이너는 이미지를 실행하기위한 인스턴스로 어플리케이션을 패키징하여 격리된 공간에서 프로세스를 동작시키는 기술입니다.

도커는 이와 같이 컨테이너 layer에 원래 이미지에 대한 변경점들을 저장합니다. 아시다 싶이 도커의 컨테이너 생성과 삭제는 매우쉽기 때문에 실수로라도 Mysql과 같은 데이터 베이스가 담긴 컨테이너를 지우는 일이 종종 존재합니다. 이를 방지하기 위해 컨테이너의 데이터를 영속적 데이터로 활용할 수 있는 방법이 있는데 그중 하나가 바로 도커 볼륨입니다. 볼륨을 활용하는 방법에는 3가지가 있습니다. 볼룸을 호스트와 공유하는 방법 볼륨컨테이너를 사용하는 방법 도커가 사용하는 볼륨을 생성하는 방법 이 3가지 입니다. 도커는 이밖에도 스웜 컴포즈 네트워크 데몬 등등 다양한 기능들을 제공합니다. 이런 기능들에 대해서 이번스터디를 통해 이해하고 사용해보고자 합니다.

도커의 사용 예

Namespace