

Container기반 Web Application

개요

컨테이너는 어플리케이션의 실제 구동 환경을 가상화하여 제공합니다. 개발자는 어떤 환경에서든 어플리케이션을 쉽고 지속적으로 배포할 수 있습니다. 가상머신은 하드웨어 스택을 가상화하지만, 컨테이너는 운영 체제 수준에서 가상화하기 때문에 OS 커널을 공유하고, 메모리를 훨씬 적게 사용하고, OS 재부팅보다 빠르게 구동할 수 있습니다.

Kubernetes는 수많은 컨테이너 어플리케이션을 자동화하여 관리하기 때문에 분산 시스템 환경을 탄력적으로 실행 및 관리할 수 있도록 제공되는 플랫폼입니다. Kubernetes는 어플리케이션의 확장과 장애 조치의 용이성을 제공하고, 쉽고 빠르게 지속적으로 배포할 수 있는 환경을 제공하여, 웹 어플리케이션 시스템을 성공적으로 구축/운영할 수 있도록 도와 줍니다.

아키텍처 다이어그램

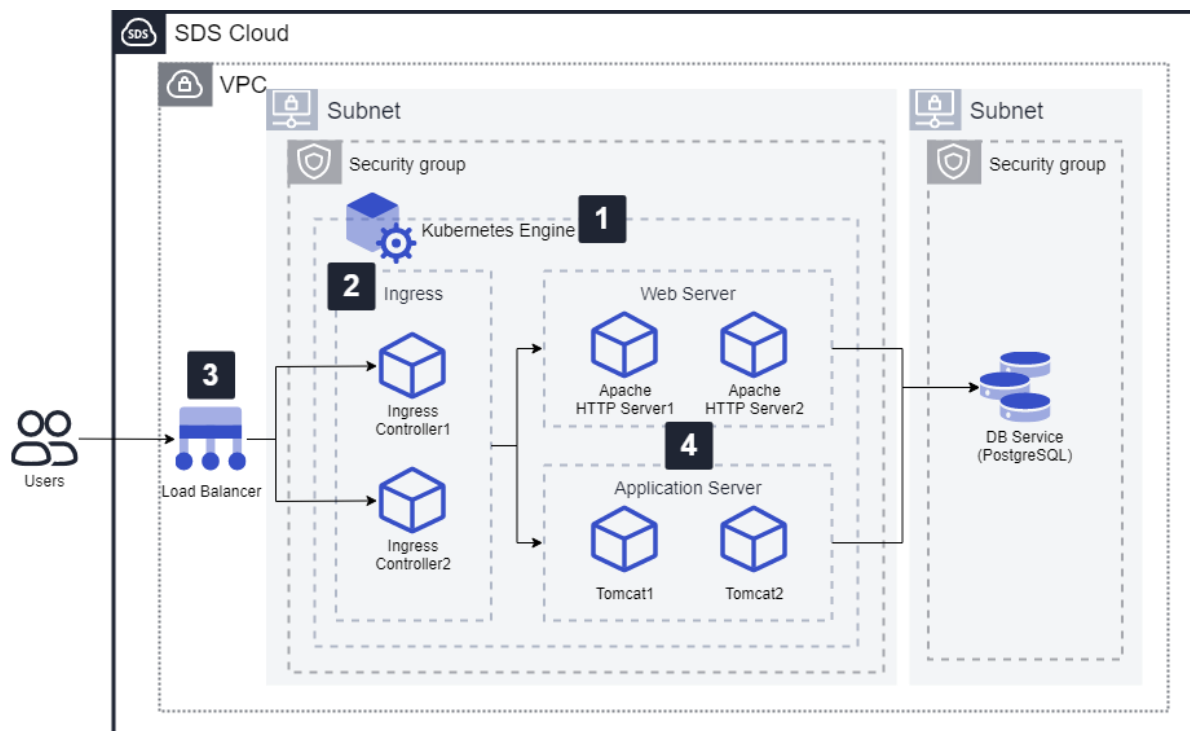


Figure 1. Container기반 Web Application 구성

1. 컨테이너 런타임 관리와 오케스트레이션을 위해 **Kubernetes Engine** 상품 서비스를 프로비저닝한다.
2. 클러스터 외부로 내부 서비스의 HTTP와 HTTPS 경로를 노출하기 위해서 Kubernetes의 **Ingress Controller**가 필요하다. **Kubernetes Apps** 상품 서비스에서 **Ingress Controller**를 신청하여 프로비저닝한다.
3. Ingress Controller의 가용성을 확보하기 위해서 **Load Balancer** 상품을 신청하고 Ingress Controller와 연결하기 위한 Worker Node IP를 등록한다.
4. WEB/WAS 어플리케이션을 구성하기 위해서 **Kubernetes Apps** 상품 서비스에서 Apache HTTP Server, Tomcat 등을 신청하여 컨테이너 프로비저닝 한다.

사용 사례

A. 경영 정보 시스템의 컨테이너 전환

컨테이너 기술 및 표준 오픈 소스 S/W를 적용하면 일반적으로 H/W 자원을 20% 내외로 절감할 수 있고, S/W의 라이선스 비용도 80% 내외로 줄어듭니다. 시스템 별 적용 기술과 S/W, 버전 등이 달라서 동일 기능에 대한 개발 및 투자 비용이 발생하던 것을 컨테이너로 전환함으로써, 중복 개발을 최소화하고, 전환 기간을 단축할 수 있습니다.

B. 사용자의 폭증에 의한 자원 증설이 필요한 시스템의 컨테이너 전환

사용자 폭증이 예상되는 시간에 앞서 WEB/WAS 컨테이너 자원을 Scale-Out하여 사용자 폭증에 대비할 수 있습니다.

사용자 폭증을 예측할 수 없는 경우를 대비하여 Kubernetes의 Auto-Scaling을 적용하여 CPU/Memory의 자원 임계 값 이상으로 사용할 때에는 자동으로 컨테이너 자원을 늘릴 수 있습니다. 반대로 임계 값 이하로 떨어지는 경우에는 자동으로 컨테이너 자원을 반납하여 비용 효율적으로 시스템 운영을 할 수 있습니다.

선결 사항

없음

제약 사항

Kubernetes Engine의 Worker Node 생성 최대 개수는 서브넷에서 사용 가능한 IP 범

위 내로 제한되며, 최대 50개까지만 생성 가능합니다.

고려 사항

컨테이너 기반의 WEB/WAS 등의 구성은 어플리케이션의 특성에 따라서 최적화되어야 합니다. **Kubernetes Apps**에서 제공하는 컨테이너 이미지는 **Kubernetes Engine**에서 구동하기 위한 기본적인 구성으로만 제공되기 때문에 최적화 구성 및 WEB/WAS 등의 환경 설정은 각 컨테이너 이미지 제공 Community를 참고하여 설정하는 것을 권고합니다.

Ingress Controller를 활용하는 경우에는 Ingress Controller의 부하 분산 및 가용성을 보장하기 위해서 **Load Balancer**를 통해서 서비스되도록 구성하는 것을 권고하며, Ingress Controller는 도메인 기준으로 어플리케이션 컨테이너를 연결 관리하기 때문에 도메인 및 DNS 등록 후에 사용하는 것을 권고합니다.

WEB/WAS 컨테이너에서 DB 시스템과 연결하기 위해서는 컨테이너가 실행되는 Worker Node의 IP로 연결됩니다. **Security Group**이나 DB 접근 제어를 사용하는 경우에는 컨테이너가 구동되는 Worker Node IP가 등록되어야 DB 시스템과 연결될 수 있습니다.

관련 상품

- Kubernetes Engine
- Kubernetes Apps
- VPC
- Virtual Server
- Load Balancer
- File Storage

관련 문서

- Container기반 CI/CD Pipeline