GKE 아키텍처 가이드

Cloud_서비스/세일즈 툴킷

Exported on 06/19/2023

Table of Contents

Public Cloud에서 컨테이너 형태로 어플리케이션을 개발 및 운영하고자 하는 경우 활용할 수 있는 방법을 정리합니다.

각 Cloud Service Provider가 제공하는 관리형 Kubernetes 서비스를 대상으로 플랫폼 관리 영역과 어플리케이션 개발 영역으로 나눠 서비스의 기능을 비교하고

어플리케이션을 개발 및 관리하는 방법을 가이드합니다.

1 GKE 1. 테스트 아키텍처

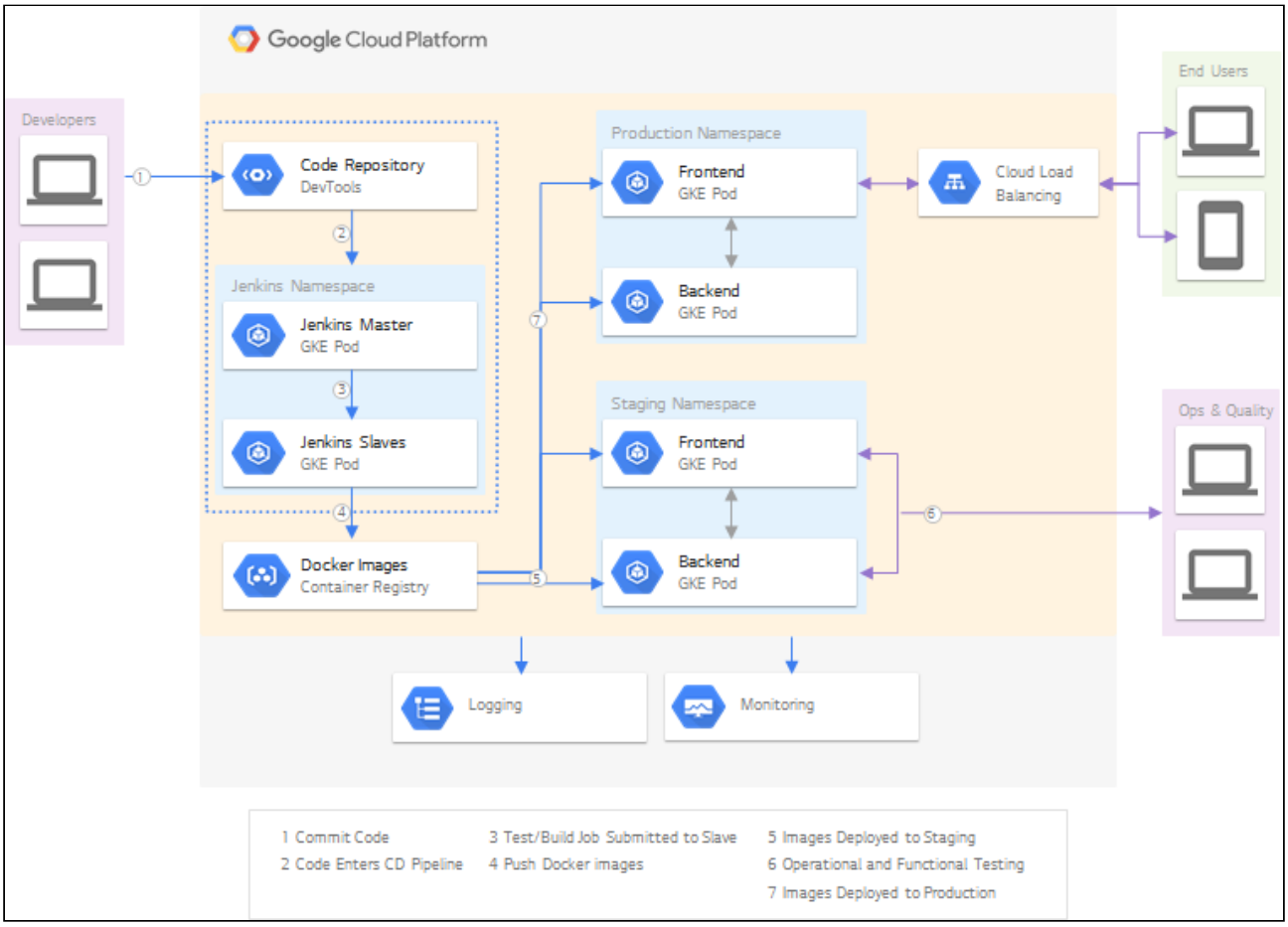
Google Kubernetes Engine을 활용하여 어플리케이션을 개발하고 관리할 때 주로 활용하게 되는 Google Cloud Platform의 서비스와 아키텍처 구성에 대해 개괄적으로 설명합니다.

1.1 GKE 개요

GKE(Google Kubernetes Engine)는 GCP(Google Cloud Platform) 기반에 컨테이너식 애플리케이션 배포를 위한 관리형 환경입니다. GKE 클러스터는 Kubernetes 명령 및 Resource를 사용하여 응용 프로그램을 배포 및 관리 하고 자동 배포를 위한 배포 정책 설정과 어플리케이션 상태 모니터링 서비스를 제공합니다.

1.2 아키텍처 구성

형상관리, CI/CD, 컨테이너 레지스트리, 런타임 어플리케이션, 데이터베이스, 로깅/모니터링 등을 어플리케이션 라이프 사이클 관점에서 구성하는 방법을 정의한다.



1.3 GKE와 GAE 비교

컨테이너를 실행할 수 있는 서비스를 비교하여 아키텍트가 각자의 환경에 적합한 서비스를 활용할 수 있도록 한다.
GKE(Google Kubernets Engine)는 Kubernetes를 기반으로 하는 컨테이너식 어플리케이션을 관리형 환경이다. 개발자 생산성, 리소스 효율성, 자동화된 작업, 오픈소스 유연성에 혁신을 가져와 제품 출시 시간을 단축해준다. OS 위에 컨테이너가 구동되는 형태로 애플리케이션과 서비스를 손쉽게 배포, 업데이트, 관리할 수 있으며, 컨테이너 복제, 모니터링, 복구를 사용하여 서비스 가용성을 높여 사용자에게 원활한 환경을 제공할 수 있다. 또한, 리소스를 최적화 하여 사용할 수 있으며, 수요에 맞게 확장/축소가 유용하다. GKE를 사용하는 개발자는 서비스를 컨테이너로 구성하고 구동하는 부분만 하면 된다.
GAE(Google App Engine)은 완전 관리형 서버리스 어플리케이션 플랫폼이다. 기본 인프라를 걱정하지 않고, 어플리케이션을 원할하게 확장할 수 있다. 개발자는 서버 관리와 구성 배포를 고민하지 않고 어플리케이션을 개발하고 사용한 리소스에 대해 비용만 지불하면 된다. 자바, PHP, Node.js, Python, C#, .Net, Ruby, Go 등 일반적인 개발 언어 및 개발자 도구를 지원하여 개발자들의 높은 생산성과 민첩성에 도움이 된다. 어플리케이션 확장, 축소, 패치와 같은 인프라 작업은 Google에서 관리한다.

구분	Kubernetes Engine	App Engine	

		Flexible Environment	**Standard Environment**
지원되는 언어	any	any(Python, Java, PHP, Go, Node.js, Ruby, .NET, Custom Runtimes)	Python, Java, PHP, Go
지원되는 버전	any	any	언어별 지원되는 버전이 고정됨
이용모델	IaaS, PaaS	PaaS	PaaS
Runtime 환경	컨테이너 형태	컨테이너 형태	VM 형태
	사용자가 만든 docker image를 실행함	소스를 기반으로 플랫폼이 docker image를 생성하여 실행하거나 사용자가 만든 docker image를 실행함	소스를 기반으로 빌드하여 사전 정의된 실행환경(sandbox)의 디렉터리에 배포하여 실행함
어플리케이션 관리	사용자가 로깅, 모니터링, health check, scaling 등 컨테이너 환경을 자유로이 설정할 수 있음	컨테이너에 대한 기본 설정(로깅, 모니터링, health check, scaling 등)은 제공하며, 사용자가 환경변수를 통해 제한적인 설정변경 가능함	컨테이너에 대한 기본 설정(로깅, 모니터링, health check, scaling 등)은 제공하며, 사용자가 환경변수를 통해 제한적인 설정변경 가능함
웹 취약점 점검	미지원	자동으로 웹 어플리케이션 취약점을 점검함	자동으로 웹 어플리케이션 취약점을 점검함
Cluster 자동 확장	수동으로 Cluster 확장	부하에 따라 자동으로 Server 확장됨	부하에 따라 자동으로 Server 확장됨
3'rd party 바이너리 설치	가능	가능	불가
SSH 접속	가능	가능(설정 필요)	불가
Local Disk 쓰기	가능(ephemeral)	가능(ephemeral)	불가
비용	할당된 Cluster의 시간당 (CPU, 메모리, 디스크 사용량)	할당된 자원의 시간당 사용량 (CPU, 메모리, 디스크 사용량)	일일 무료 사용 후 인스턴스 실행시간

장점	• On Premise에서 동일한 Kubernetes 환경을 구성할 수 있음 • 컨테이너 오케스트레이션에 대해 통제 가능(설정 변경 등)	• 로깅 설정, scaling, 버전 upgrade, 트래픽 분산등을 자동으로 처리하므로 개발자가 컨테이너를 설정할 필요없음	• 빠르게 Scale Up/Down 가능(Milliseconds) • 트래픽이 없는 경우 0으로 Scale되어 비용이 발생하지 않음
적용모델	Container Workload	확장가능한 웹 어플리케이션 및 모바일 backend 어플리케이션, Container Workload	확장가능한 웹 어플리케이션 및 모바일 backend 어플리케이션

1.4 Kubernetes와 GKE 비교

오픈소스 Kubernetes 대비 Google Cloud가 제공하는 차별적인 기능에 대해 소개한다.

- 액세스 관리 : Google 계정 및 역할 권한으로 클러스터의 액세스 권한을 관리함
- 네트워킹 : Google Cloud VPN를 이용해 컨테이너 클러스터에 하이브리드 네트워킹을 구성이 가능함
- 보안 및 규정 : Google 보안팀이 Kubernetes Engine을 지원하며 HIPAA 및 PCI DSS 3.1 규정을 모두 준수
- 통합 로깅 및 모니터링 : Stackdriver Logging과 Stackdriver Monitoring을 사용 설정하면 애플리케이션 실행 상태를 손쉽게 확인이 가능함
- 자동 업그레이드 : Kubernetes의 최신 출시 버전으로 클러스터를 자동으로 최신 상태로 유지함
- 자동 복구 : 자동 복구를 사용 설정하면 노드 상태 확인에 실패할 경우 Kubernetes Engine이 실패한 노드의 복구 프로세스를 시작함
- 완전 관리형 서비스 : Google SRE가 Kubernetes Engine을 완전하게 관리하므로 클러스터의 가용성이 보장되고 최신 상태로 유지됨
- Private Container Registry : Google Container Registry와 통합하면 손쉽게 Private Docker 이미지를 저장하고 액세스가 가능함
- 신속한 빌드 : Google Container Builder를 사용하면 인증을 설정할 필요 없이 Kubernetes Engine에서 안정적으로 컨테이너를 배포가 가능함
- 사용 편의성 : 콘솔 화면 내에 터미널 접속 기능이 있어, 별도의 CLI 설치 작업 없이 손쉽게 사용이 가능함

- GKE 개요(see page 4)
- 아키텍처 구성(see page 4)
- GKE와 GAE 비교(see page 5)
- Kubernetes와 GKE 비교(see page 7)

2 GKE 2. 플랫폼 테스트 결과

2.1 테스트 결과 요약

Google Cloud Platform에서 서비스중인 GKE(Google Kubernetes Engine)의 플랫폼 기능을 사용해보고 테스트 결과를 설명합니다. Google GKE는 다른 퍼블릭 클라우드 및 Kubernetes 기반 PaaS 플랫폼과 비교하여 가장 완성도가 높으며 오랜 운영 노하우를 바탕으로 필요한 옵션 기능을 플랫폼 기본 기능에 포함시켜 놓았다. 특히 Kubernetes auto upgrade와 auto repair와 같은 자동화 기능이 우수하며 Stackdriver와 같은 모니터링 서비스와의 연계 기능도 잘 되어 있다.

2.2 테스트 결과 상세

구분	항목	설명	배점	가중치	Google Kubernetes Engine	
					평가결과	평가의견
설치/변경	Host OS 설치/구성 자동화	VM 의 OS 설치 및 구성이 자동화 되는가	20	1	2	- 기능 지원 - GKE 사용을 위한 host cluster 생성시 zone/ region 옵션을 선택 가능 - Master 버전은 default 버전과 상위버전 3개를 추가로 제공 - Node 버전은 더 많은 버전을 제공함(region 별로 일부 지원 버전의 차이 있음) - 설치되는 호스트 OS는 cos(containner-optimized os) 버전과 ubuntu 제공 - GKE host cluster 생성시 다양한 cluster option 제공

구분	항목	설명	배점	가중치	Google Kubernetes Engine	
					평가결과	평가의견
	Host OS 업그레이드 자동화	VM 의 OS Upgrade 가 플랫폼 중단 없이 자동화 되는가		1	0	- 지원 안됨 - 제공되는 host cluster OS 버전은 선택 불가함 - OS는 cos->ubuntu 변경은 가능하나 ubuntu 선택 시 플랫폼 사용 기능 제약 (auto-repair, auto-upgrade 사용 불가)
	Host OS 다운그레이드 자동화	VM 의 OS Downgrade 가 플랫폼 중단 없이 자동화 되는가		1	0	- 지원 안됨 - 제공되는 host cluster OS 버전 선택 및 다운그레이드 기능은 제공 되지 않음
	플랫폼 설치	플랫폼 최초 설치 (플랫폼 권장하는 버전으로) 되는가		3	3	- 기능 지원 - GKE 플랫폼 default 버전 및 상위 3개 버전 설치 가능 버전 제공
	플랫폼 설치 정의서	정의서(예: Manifest YAML) 를 통해 설치가 자동화 되는가.		2	2	- 기능 지원 - YAML 또는 등록된 플랫폼 템플릿을 통한 설치 기능 지원
	플랫폼 설치 정의서의 민감정보 분리 보관	정의서에서 민감한 정보 (예: 계정, 비번, 인증서, Private Key) 을 분리 보관할 수 있는가.		2	2	- 기능 지원 - Google 계정 및 역할 권한으로 클러스터의 액세스 권한을 관리합니다.
	플랫폼 설치 정의서의 민감정보 암호화	분리된 민감 정보를 안전하게 보관하는 메커니즘이 제공되는가		2	2	- 기능 지원 - Google 계정 및 역할 권한으로 클러스터의 액세스 권한을 관리합니다.
	설치 정의서의 템플릿화	정의서의 변경을 통해 설정 변경이 가능한가.		2	2	- 기능 지원 - GKE host cluster 구성 시 선택한 옵션을 템플릿화 하여 재사용 가능 - 기존에 운영중인 host cluster 동일하게 구성하는 clone 기능 제공

구분	항목	설명	배점	가중치	Google Kubernetes Engine	
					평가결과	평가의견
	설치/관리 GUI 조회 제공 여부	설치된 플랫폼의 구성 정보를 확인하기 위한 GUI 가 제공되는가		2	2	- 기능 지원 - GCP(Google Cloud Platform) 콘솔을 통해 기능 제공
	설치/관리 GUI 변경 제공 여부	GUI 에서 Ansible Inventory 정보 수정/배포 기능이 있는가		2	2	- 기능 지원 - GCP(Google Cloud Platform) 콘솔을 통해 기능 제공
	플랫폼 검증 자동화	설치후 시스템 정상 여부 확인 자동화가 가능한가		2	1	- 기능 지원 - GEK host cluster 생성 후 cluster summary tab 에서 기본 제공됨 (conditions 조건에 플랫폼의 동작의 필수 조건 정의 및 verify 수행)
업그레이드	플랫폼 업그레이드	호환성 유지 버전 업그레이드 기능 확인이 가능한가	15	4	4	- 기능 지원 - 수동/자동 Upgrade를 지원하나 최신 버전은 적용할 수 없음 (플랫폼에서 제공되는 버전으로만 업그레이드 가능함) - Master/Worker node 업그레이드 각각 수행 가능 - 자동 업그레이드는 cluster host os 버전을 cos 선택시에만 가능함. - 업그레이드 되는 동안 해당 node에서 서비스 및 클러스터 구성 변경 불가

구분	항목	설명	배점	가중치	Google Kubernetes Engine	
					평가결과	평가의견
	롤백	업그레이드 도중 장애 발생시 롤백될 수 있는가		4	4	- 기능 지원 - 지원함, 단 성공적으로 Upgrade된 경우에는 적용할 수 없음 - Upgrade 과정 중에 Rollback, Cancel 가능하며, 이전 버전으로 복원됨 - Cloud shell을 이용한 CLI 로만 지원하며 GUI 콘솔에는 업그레이드 기능만 제공함
	플랫폼 다운그레이드	호환성 유지 버전 다운그레이드 기능 확인이 가능한가		4	2	- 기능 지원 - Cloud shell을 이용한 cli로만 지원하며 롤백 기능으로 Downgrade를 지원함 - Master node의 downgrade는 제공하지 않음 - Cloud shell에서 CLI로만 기능 제공(콘솔에는 다운그레이드 기능 없음)
	자동 플랫폼 업그레이드	자동 버전 업그레이드 기능을 제공하는가		3	3	- 기능 지원 - 사용자가 정의한 maintenance window 기간에 자동으로 updgrade 진행함. - 특정 node의 자동 upgrade를 disable 가능함 - COS((containner-optimized os)만 지원하며 ubuntu os인 경우 제공하지 않음
내결함성	Master Node 장애	마스터 노드 장애시 플랫폼 가용성 확인이 가능한가	15	3	3	- 기능 지원 - GKE cluster host 생성시 Highly available 옵션 제공 (Master/node가 multi zone에 자동 replication 구성됨)

구분	항목	설명	배점	가중치	Google Kubernetes Engine	
					평가결과	평가의견
	Workder Node 장애	워커 노드 장애시 플랫폼 가용성 확인이 가능한가		3	3	- 기능 지원 - 호스트 장애 상황 확인이 어려우나 가용존 이중화로 무중단 서비스 구성 가능
	Infra Node(운영 관리 Node) 장애	인프라 노드 장애시 플랫폼 가용성 확인이 가능한가		2	2	- 기능 지원 - 호스트 장애 상황 확인이 어려우나 가용존 이중화로 무중단 서비스 구성 가능
	Router Node 장애	라우터 노트 장애시 플랫폼 가용성 확인이 가능한가		3	3	- 기능 지원 - 호스트 장애 상황 확인이 어려우나 가용존 이중화로 무중단 서비스 구성 가능
	PV 스토리지 내결함성	스토리지 장비 또는 서비스 장애시 플랫폼 내결함성 범위 확인이 가능한가		2	2	- 기능 지원 - 디스크 장애 확인은 불가하나 Persistent Disk with Multiple readers, SSD Persistent Disk, Regional Persistent Disk(beta) 스토리지 서비스 제공
	자동 노드 복구	Worker Host 장애를 자동 인지하고 자동 복구 기능을 제공하는가		2	2	- 기능 지원 - Node 장에에 자동 repair 수행하며 특정 node 단위로 설정변경가능함 - COS(Container-Optimized) OS인 경우 제공함

구분	항목	설명	배점	가중치	Google Kubernetes Engine	
					평가결과	평가의견
백업 / 복구	플랫폼 백업/복구	플랫폼 삭제 및 동일 버전 재설치 후 데이터 복구 가능 여부 확인이 가능한가	5	5	1	- 지원 안됨 - 플랫폼의 기능을 백업 받기 보다는 template등을 통해 재배포 하는 방식 권고 - OS snapshot 기능을 이용하여 백업/복구 구현 가능
스케일링	Master Node 스케일링	운영자원 효율화를 위해 마스터 노드를 증설하거나 감소할 수 있는가	5	1	1	- 기능 지원 - Master node는 GCP 플랫폼에서 관리함
	Workder Node 스케일링	운영자원 효율화를 위해 워커 노드를 증설하거나 감소할 수 있는가		2	2	- 기능 지원 - cluster node 확장하는 방식으로 지원 가능
	Infra Node(운영관리 Node) 스케일링	운영자원 효율화를 위해 인프라 노드를 증설하거나 감소할 수 있는가		1	1	- 기능 지원 - cluster node 확장하는 방식으로 지원 가능
	Router Node 스케일링	운영자원 효율화를 위해 라우터 노드를 증설하거나 감소할 수 있는가		1	1	- 기능 지원 - Master node는 GCP 플랫폼에서 관리함
플랫폼 관리	운영자 관리 GUI 제공	운영자가 PaaS 플랫폼을 관리 할 수 있는 GUI 관리 화면을 제공하는가	20	4	4	- 기능 지원 - 웹 형태의 Console 및 cloud shell 제공함
	운영자 계정 관리	LDAP 계정과 연계한 운영자 계정 관리가 가능한가		4	4	- 기능 지원 - IAM & admin 서비스에서 사용자 및 권한 통합 관리 가능하며 LDAP 연동 가능
	유관시스템 계정 연계(모니터링/로그/미터링)	LDAP 계정과 연계한 모니터링/미터링 계정 관리가 가능한가		5	5	- 기능 지원 - IAM & admin 서비스에서 사용자 및 권한 통합 관리 가능하며 LDAP 연동 가능

구분	항목	설명	배점	가중치	Google Kubernetes Engine	
					평가결과	평가의견
	Maintenance Window 지정	업무 부하가 작은 시간대를 정하여 Maintenance 가능한가		3	3	- 기능 지원 - 자동/수동 Maintenance 윈도우 설정이 가능하며 자동 업그레이드 기능 제공
	PV 스토리지 자원 관리 기능	PV 자원의 할당 / 회수 자동화 기능이 가능한가		4	3	- 기능 지원 - 디스크 장애 확인은 불가하나 Persistent Disk with Multiple readers, SSD Persistent Disk, Regional Persistent Disk(beta) 스토리지 서비스 제공
모니터링	Host 자원 모니터링	각 VM의 자원 모니터링을 통합하여 제공하는가	10	2	2	- 기능 지원 - Stackdriver 모니터링 기능으로 제공하며, 로그는 30일동안 보관됨
	컨테이너 배치 모니터링	각 컨테이너의 자원 모니터링을 통합하여 제공하는가		3	3	- 기능 지원 - Stackdriver 모니터링 기능으로 제공하며, 로그는 30일동안 보관됨
	데이터 보관기간 설정	모니터링 데이터 보관 기능을 조정할 수 있는가		2	1	- 지원 안됨 - 별도의 모니터링 데이터 보관 기간 조정을 불가함 - bigquery로 장기 데이터 보관 및 분석 가능함
	알람 기능 제공	플랫폼 운영자 관점에서 알람기능을 제공하는가		3	3	- 기능 지원 - Stackdriver 모니터링 기능으로 제공하며, 로그는 30일동안 보관됨
로그통합	Host 로그통합	각 VM의 로그를 통합하여 제공하는가	10	2	2	- 기능 지원 - Stackdriver 로깅 기능으로 제공하며, 로그는 30일동안 보관됨

구분	항목	설명	배점	가중치	Google Kubernetes Engine	
					평가결과	평가의견
	컨테이너 로그통합	컨테이너 Lifecycle 및 Event 로그를 통합하여 제공하는가		2	2	- 기능 지원 - Stackdriver 로깅 기능으로 제공하며, 로그는 30일동안 보관됨
	데이터 보관기간 설정	로그 데이터 보관 기능을 조정할 수 있는가		2	1	- 지원 안됨 - 별도의 모니터링 데이터 보관 기간 조정을 불가함 - bigquery로 장기 데이터 보관 및 분석 가능함
	Audit 로그 제공	보안 관점에서 분석 가능한 audit logs를 제공하는가		1	1	- 기능 지원 - Kube-apiserver를 통해 audit logging 기능 제공 (beta) 가능 (Kube-apiserver는 GCP 포함되어 endpoint만 제공하는 방식)
미터링	사용자 과금 정책 적용	사용자 정의 과금정책 적용 가능한가		1	0	- 지원 안함
	인스턴스 단위 미터링	컨테이너수 단위 미터링 가능한가		1	1	- 부분 기능 지원 - stackdriver의 logging 메뉴를 사용해 로깅은 가능하나 별도의 과금 체계 개발 필요
	자원사용량 단위 미터링	CPU, MEMORY, Storage 사용량 단위 미터링 가능한가		1	1	- 부분 기능 지원 - stackdriver의 logging 메뉴를 사용해 로깅은 가능하나 별도의 과금 체계 개발 필요

- 테스트 결과 요약(see page 8)
- 테스트 결과 상세(see page 8)

3 GKE 3. 어플리케이션 테스트 결과

3.1 테스트 결과 요약

컨테이너 빌드/배포/관리를 위한 유용한 도구를 제공하며, GCP와 결합하여 보안 설정 및 모니터링 부분도 효과적으로 관리가 가능하다.

Container를 처음 사용하는 사용자도 쉽게 접근할 수 있도록 별도의 Client 환경 구성 없이 CLI를 사용할 수 있는 Cloud shell과 같은 유용한 도구를 제공하며, Stackdriver를 이용해 모니터링, Tracing, 로깅, Debugging 등을 효과적으로 관리할 수 있다. 또한, Container Registry를 구성해서 별도의 Private Registry를 설치 하지 않아도 사용이 가능하다. 그 외에 Istio, Spinnaker 등 Kubernetes 진영의 MSA 도구와 유기적으로 연동되었다.

3.2 테스트 결과 상세

구분	항목	설명	배점	가중치	Google Kubernetes Engine	
					평가결과	평가의견
컨테이너 빌드/배포	소스코드 빌드/배포	소스 코드를 이용한 빌드/배포 기능	15	5	0	- 제공되지 않음
	DockerHub, 사용자 개발 컨테이너 이미지 배포	컨테이너 이미지를 이용한 빌드/배포 기능		5	5	- 기능 확인함 Container Registry를 이용하여 이미지 배포 및 관리가 가능함
	Recreate 배포기능 확인	Recreate 배포 방식 지원		1	1	- 기능확인함
	Rolling 배포기능 확인	Rolling 배포 방식 지원		1	1	Deployment config 베이스로 배포정책이 설정되고 수행됨
	Rollback 배포기능 확인	Rollback 기능 지원		1	1	- 기능확인함. kubectl rollout undo deployment/hello

구분	항목	설명	배점	가중치	Google Kubernetes Engine	
					평가결과	평가의견
	BlueGreen 배포기능 확인	BlueGreen 배포 방식 지원		1	0	- 기본제공 하지 않으며 spinnaker를 활용하여 제공 가능
	Canary 배포기능 확인	Canary 배포 방식 지원		1	0	- 기본제공 하지 않으며 spinnaker를 활용하여 제공 가능
컨테이너 레지스트리 관리	Container Registry Service 확인	Container Registry 제공 여부, 설치 편의성	5	2	2	- Container Registry를 제공함 - CLI를 통해 이미지를 Push하거나 웹콘솔에서 생성하면 사용이 가능함
	Container Regsitry 내 이미지 관리 편의성	이미지 버전 관리 편의성		1	1	- 이미지 버전 관리 가능하며 사용이 편리함
	이미지 관리를 위한 추가 기능 확인	별도의 관리 Portal 제공여부, 웹취약점, Docker signed 확인 기능		2	2	- Google 콘솔 내 관리 Page가 존재함 - 취약점 스캔 기능 제공, 위험한 이미지는 배포되지 않도록 컨테이너 이미지 자동 잠금 트리거링 기능 제공함
컨테이너 관리	컨테이너 자원할당 지정	CPU, Memory, storage 지정	25	3	3	- 기능확인함 클러스터 생성 시 CPU, Memory를 지정해서 배포 가능

구분	항목	설명	배점	가중치	Google Kubernetes Engine	
					평가결과	평가의견
	컨테이너 이미지 관리(버전 관리)	컨테이너 이미지, 버전 관리 기능		3	3	- Container Registry를 사용하여 namespace, tag등을 docker hub와 동일하게 사용이 가능함 - namespace, tag등을 docker hub와 비슷하게 쓸 수도 있음. (간단한 업무명칭으로 정해 쓰는게 일반적임) - Build Trigger를 설정하여 자동으로 배포관리 될 수 있도록 설정이 가능함 - Deployment를 Edit 하여 저장하면 이미지 변경분이 자동으로 반영됨 (kubectl edit deployment hello)

구분	항목	설명	배점	가중치	Google Kubernetes Engine	
					평가결과	평가의견
	컨테이너 확장 시 자동식별 및 부하분산	컨테이너 수동/ 자동으로 확장 시 부하분산 기능		5	5	- 수동확장은 웹콘솔 내 Kubernetes Engine > 클러스터 > 수정에서 node 개수 변경 가능하고 CLI로도 가능함(kubectl scale deployment <appName> --replicas=4) - 자동확장은 설정한 CPU 사용량을 기준으로 가능함 '- 서비스 생성 시 유형을 LoadBalancer로 생성 가능함 (kubectl expose deployment [Deployment명] --port 80 --type LoadBalancer) - 타겟 CPU를 정해서 (Default 80%) 해당 값 이상으로 변경되면 자동으로 node가 늘어남 - node의 최소, 최대 개수를 정할 수 있음 - 타겟 CPU를 5%로 설정하여 테스트 후 CPU 변화에 따라 node가 추가되는 것을 확인함 (https://google.qwiklabs.com/focuses/651?parent=catalog 참조)
	컨테이너 확장 시 부하조정	컨테이너 확장 시 부하 비중 비율 조정 가능 여부		4	4	- Istio를 사용하여 Service와 Label을 이용하여 조정이 가능함
	Container to Container 통신	컨테이너 간 통신 설정		3	0	- 서비스 단위로 통신 가능
	route 관리	사용자 정의 route 등록/변경/삭제		2	1	- VPC Network > Route 메뉴를 통해 Route를 관리할 수 있음

구분	항목	설명	배점	가중치	Google Kubernetes Engine	
					평가결과	평가의견
	사용자 인증서 추가	사용자가 Route에 별도의 인증서 등록, 관리 기능		3	2	- Helm을 이용해 ngnix ingress를 설치 후 Kubernetes secrete 생성 -> 어플리케이션 배포 후 YAML에 tls 정의하여 사용가능
	어플리케이션 health check	http/port/process 기반의 Health Check 기능		2	2	- 기능확인함 - Service를 통해 Health Check를 진행하려면 fireware rules를 설정해야함 (130.211.0.0/22 및 35.191.0.0/16 에 대한 접근을 허용해주어야 함)
어플리케이션 개발	로컬에서 Eclipse Plug-in을 이용한 개발	이클립스 기반의 개발자 도구가 제공 여부	15	2	1	- Eclipse에 Google Cloud Platform을 사용할 수 있는 SDK를 지원하나 개별 구성해야 하고, SDK를 찾기가 쉽지 않고 가이드가 없음 - App engine, kubectl을 component로 지원함
	로컬에서 원격 debugging	로컬 환경에서 원격 디버깅 지원 여부		3	0	- 제공되지 않음
	컨테이너 실시간 로그 분석	컨테이너(Pod) 내 실시간 로그 확인		3	3	- 기능확인함 (CLI를 통해 확인이 가능함)
	컨테이너 SSH 접속	컨테이너(Pod)에 SSH 접속 가능 여부 확인		3	3	- Cloud shell을 활성화 하여 원하는 Pod에 접속이 가능함 kubectl exec -ti POD_NAME bash
	파이프라인 생성/수정	파이프라인 생성/수정 확인 (Jenkins, Spinnaker, Concourse 등 활용 가능 여부 확인)		4	2	- 서비스로 기본제공 하지 않으며, 설치형으로 Jenkins/ Spinnaker를 활용하여 사용할 수 있음

구분	항목	설명	배점	가중치	Google Kubernetes Engine	
					평가결과	평가의견
어플리케이션 성능/장애 분석	컨테이너 과거 로그 분석	프로젝트/Pod/서비스 단위의 로그 분석 기능	15	3	3	- StackDriver를 통해 지원함 - StackDriver의 Debug 메뉴에서 실행중인 Application의 소스에 Log를 추가하여 확인이 가능함
	Java 분석 기능 (Jconsole, Dump 수집)	Jconsole을 이용한 모니터링 가능 여부 Java의 Heap, Thread Dump 생성 및 추출 가능 여부		2	2	- Java에서 제공해주는 기본 분석 기능 사용이 가능함
	컨테이너 자원 모니터링	Pod별 CPU, 메모리, 디스크 사용량에 대한 모니터링		5	5	- Kubernetes Engine > 클러스터 > 클러스터명선택 > 노드 > 특정노드선택으로 해당 pod의 CPU, 메모리, 디스크 사용량 등에 대한 모니터링이 가능함 - Kubernetes graphical dashboard를 제공하여 Cluster별 관련 Resource (Deployment, Replica Set, Pod, Service 등)에 대한 모니터링이 가능함
	어플리케이션 호출 분석	MSA 서비스간 call trace 분석 가능 여부		2	2	- Stackdriver를 통해 지원함
	사용자를 위한 알람 기능 제공	플랫폼 사용자 관점의 알람 기능 제공 여부		3	3	- Stackdriver를 통해 지원함
프로젝트 관리	개발자 사용을 위한 GUI 제공	개발자용 웹콘솔 확인	15	1	1	- 웹콘솔 및 cli(Cloud Shell), 코드편집기 등이 제공됨
	CLI 설치 편의성	CLI 환경 구성 시 설치 편의 및 자동화 여부		1	1	- 별도의 설치 없이 Cloud Shell을 통해 CLI 사용이 가능함

구분	항목	설명	배점	가중치	Google Kubernetes Engine	
					평가결과	평가의견
	사용자 관리	사용자 관리 기능 및 편의성 LDAP, SSO 등과의 연동을 통한 외부 사용자 정보를 이용해 일괄 사용자 등록이 가능한지에 대한 확인		2	2	- LDAP과 연계하여 프로젝트에 사용자 추가가 가능함 - Service Account는 Google API를 사용할 수 있는 계정으로 프로젝트 생성 시 1개의 Compute Engine Service 계정과 1개의 App Engine Service 계정을 자동으로 생성함. 프로젝트 관리자는 98개의 service Account 추가로 생성할 수 있음
	권한 관리	리소스별/ 사용자별 권한 관리		3	3	- 생성된 Service Account에 Resource별 권한을 부여할 수 있음 - Kubernetes와 연관된 Role은 Kubernetes Engine Admin, Cluster Admin, Developer, Host Service Agent User, Viewer가 있음 - 여러 권한의 조합도 가능
	프로젝트 생성/ 삭제	프로젝트 생성 및 프로젝트 관리자 할당		2	3	- 기능 지원 - Console 및 cloud shell을 사용하여 프로젝트 생성 및 삭제 기능 제공
	프로젝트 내에서 하위그룹 관리	개발, 테스트, 운영 등 개발단계별 관리		1	0	- 제공되지 않음
	Quota 관리	CPU, MEMORY, Container수, Route 수 제한		2	2	- IAM을 통해 각 자원별 Quota를 확인할 수 있으며, 계정이 업그레이드 되면 사용자가 사용할 수 있는 Quota가 늘어나는 방식으로 프로젝트 단위로 관리가 됨

구분	항목	설명	배점	가중치	Google Kubernetes Engine	
					평가결과	평가의견
	Instance App Template 생성	사용자 정의 Application Template 생성이 가능한가 이미지와 템플릿 사이의 workflow 차이 확인		3	2	- Helm을 이용하여 가능함
보안/네트워킹	프로젝트 단위 보안 설정	프로젝트간 통신 설정이 가능한지 확인 (Multi tenency Enabled)	10	2	1	- Namespace를 이용하여 프로젝트 개념으로 사용이 가능함
	컨테이너 그룹 단위 보안 설정	컨테이너 그룹을 정의하고 그룹간 통신 설정이 가능한지 확인		2	0	- 지원안됨
	지정 컨테이너 단위 보안 설정	컨테이너별 통신 설정이 가능한지 확인		2	1	- Istio를 사용하여 Service 단위로 보안 설정을 제어할 수 있음
	특정 Role로 통신 제한(inbound)	특정 APP 이 외부의 정해진 IPs, PORT 에 대해서만 inbound 설정 가능한지 확인		2	0	- 지원안됨 - 방화벽을 통해서 차단하거나, istio를 사용하여 Cookie 정보를 이용하여 접근 권한을 제어할 수 있음
	특정 Role로 통신 제한(outbound)	특정 APP 이 외부 연계를 위해 특정 IPs, PORT 로 outbound 가능하도록 설정 가능한지 확인		2	2	- Network Policy를 지원함

- 테스트 결과 요약(see page 16)
- 테스트 결과 상세(see page 16)

4 GKE 4. 도입구축 시 고려사항

4.1 서비스 리전

Google은 현재 시점('18년 10월)에 한국에 데이터선터를 가지고 있지 않으며, 한국과 가장 가까운 데이터센터는 일본 도쿄이다. (아시아에는 대만, 싱가폴, 일본에 데이터 센터가 있다.)
'19년 한국에 데이터센터를 설립할 예정이며 GKE 서비스가 될 예정이다.

4.2 비용 산정

4.2.1 Container Registry

Container Registry 비용은 저장한 이미지 개수가 아니라 이미지 저장에 사용되는 **Cloud Storage 저장소 비용**과 데이터 전송에 사용한 **Network 송신량 비용**으로 구성됩니다. Container Registry 저장소에 처음으로 이미지를 Push 하면 시스템에서 모든 이미지를 저장하기 위한 Cloud Bucket을 만듭니다. 이때 사용되는 Cloud Bucket은 Multi-Regional 스토리지 입니다.

- Cloud Storage 및 네트워크 송신 가격표[1]

4.2.2 GKE

GKE(Google Kubernetes Engine)은 GCE(Google Compute Engine)을 클러스터 노드로 사용합니다. 따라서 GCP 노드 대수에 비례하여 비용이 산정되며 클러스터 노드 생성 시점 부터 삭제될 때까지 청구되며 최소 사용 기준은 1분이며 그 이후로는 초단위로 사용시간 만큼 과금 됩니다. 아래의 링크를 참조하여 구성하고자 하는 옵션을 선택하며 월비용을 예상해 볼 수 있습니다.

- GCP Pricing Calculator[2]

4.2.3 Stackdriver

Stackdriver를 사용하면 자원 사용량과 지출관리를 할 수 있으며 stackdriver 사용가격은 사용한 만큼 지불하는 방식 입니다. 용도별 월별 무료 사용 할당량이 있으며 무료 사용량 이상 사용한 부분에 대해서 사용료를 지불하게 됩니다. Cloud Bill Reports 페이지와 Logging, Monitoring, Trace 콘솔에서 현재 사용량을 확인 할 수 있으며 이를 기준으로 향후 사용량을 예상하여 Stackdriver 예상 청구금액을 산정 할 수 있습니다.

1 https://cloud.google.com/storage/pricing#storage-pricing
2 https://cloud.google.com/products/calculator/

- stackdriver 예상 청구액 확인[3]
- Stackdriver Pricing Calculator[4]

4.2.3.1 Stackdriver 가격

기능	가격[5]	월별 무료 할당량	시행일
Logging[6]	$0.50/GiB	프로젝트당 50GiB까지	2018년 7월 1일
Monitoring[7] 데이터	$0.2580/MiB: 150~100,000MiB$0.1510/MiB: 100,000~250,000MiB$0.0610/MiB: 250,000MiB 초과	모든 GCP 측정항목[8] 결제 계정당 150MiB까지	2018년 7월 1일
Monitoring[9] API 호출	API 호출 1,000개당 $0.01	API 호출 100만 개까지	2018년 7월 1일
Trace[10] 내부 데이터화	스팬 100만 개당 $0.20	스팬 250만 개까지	2018년 11월 1일
검색한 Trace[11] 스팬	스팬 100만 개당 $0.02	스팬 2,500만 개까지	공지 예정

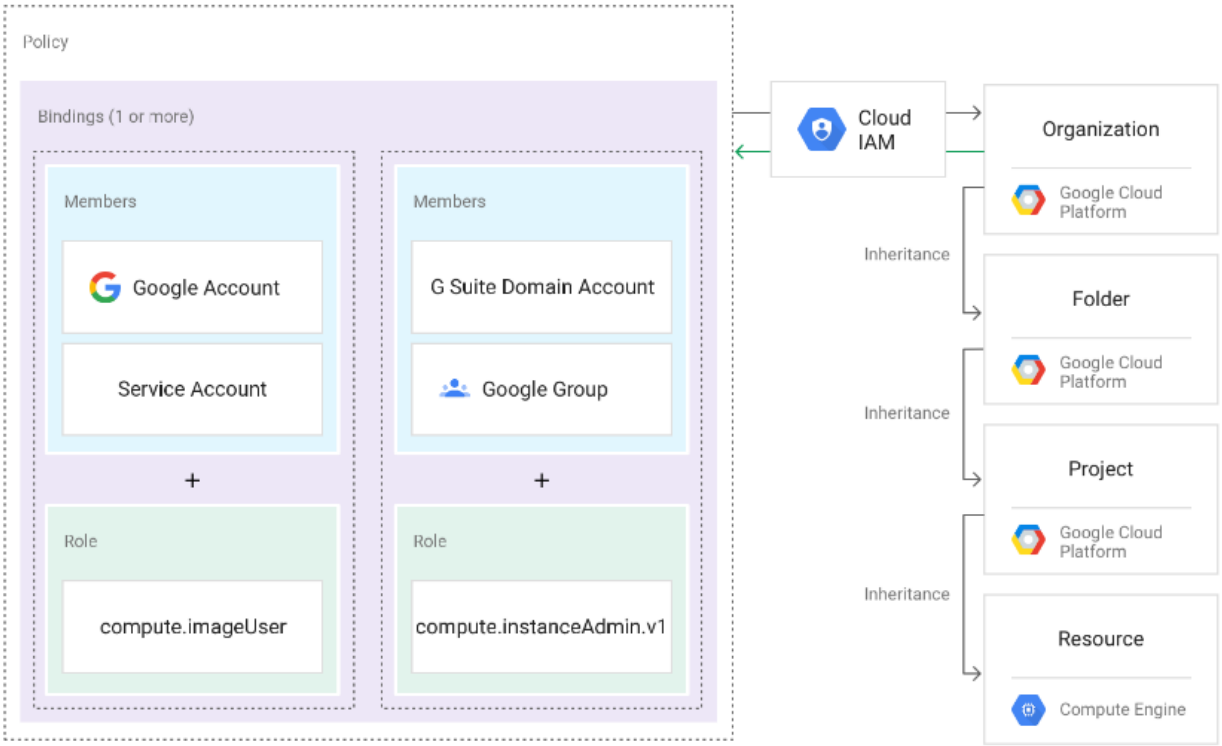
4.2.3.2 Stackdriver 로그 보관 기간

로그 유형	보관 기간
관리 활동 감사 로그	400일(프리미엄 등급)400일(기본 등급)
데이터 액세스 감사 로그	30일(프리미엄 등급)7일(기본 등급)
감사 로그 외의 로그	30일(프리미엄 등급)7일(기본 등급)

4.3 Google Cloud IAM 계정 관리 방안

3 https://cloud.google.com/stackdriver/estimating-bills
4 https://cloud.google.com/products/calculator/#tab=google-stackdriver
5 https://cloud.google.com/stackdriver/pricing#binary-units
6 https://cloud.google.com/logging/docs
7 https://cloud.google.com/monitoring/docs
8 https://cloud.google.com/monitoring/api/metrics_gcp
9 https://cloud.google.com/monitoring/docs
10 https://cloud.google.com/trace/docs
11 https://cloud.google.com/trace/docs

Google Cloud 내에서는 IAM을 이용하여 조직이 필요로 하는 보안 정책을 수립하고 정책에 따라 자동으로 사용자의 계정과 권한을 관리할 수 있다. IAM을 이용해 효율적으로 클라우드 운영 조직을 관리 하기 위해서 부서나 서비스 단위로 독립적인 프로젝트 및 빌링 관리가 필요하며 또한 실제 회사의 조직을 클라우드 운영 조직에 맵핑할 수 있어야 한다.



조직은 프로젝트의 상위개념으로 IAM 에서 설정해 줄 수 있다. 프로젝트는 Resource의 관리 단위로 빌링의 단위이다. GKE를 구성할 때 Quota를 설정해 준다면 이 값은 프로젝트 단위로 관리가 된다. 실제 프로젝트를 진행할 경우에 사용자의 계정은 LDAP 를 연결하여 생성할 수 있다. 참고로 GKE의 Service Account는 Google API를 사용할 수 있는 계정을 의미하는 것으로 최대 100개까지 생성이 가능한다.

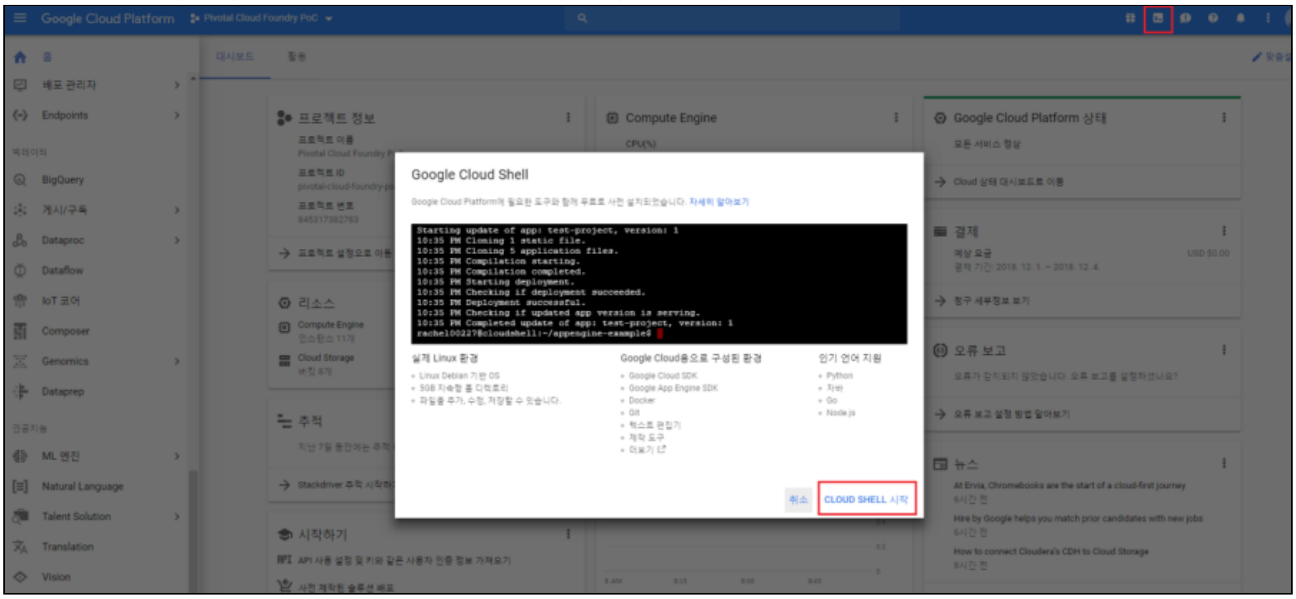
4.4 Cloud shell 소개

GKE에서는 콘솔화면을 통해 사용자가 바로 이용이 가능한 가상 머신 인스턴스를 제공한다.
Google Cloud 콘솔 화면에 접속하면 Cloud shell을 활성화하면 가상머신을 이용이 가능하다. Cloud shell에는 다음과 같은 도구들이 설치되어 있다.

유형	도구
Linux 셸 인터프리터	bash, sh
Linux 유틸리티	표준 Debian 시스템 유틸리티

유형	도구
Google SDK 및 도구	Google App Engine SDK, gcloud 명령줄 도구를 포함한 Google Cloud SDK, Cloud Storage용 gsutil
텍스트 편집기	Emacs, Vim, Nano
빌드 및 패키지 도구	Gradle, Make, Maven, Bazel, npm, nvm, pip
소스 제어 도구	Git, Mercurial
추가 도구	Kubectl, Docker, iPython, MySQL 클라이언트, gRPC 컴파일러, TensorFlow

화면 우측 상단의 아이콘을 클릭하면 Cloud shell에 접속이 가능하다.



Cloud shell에 접속 시 현재 선택된 프로젝트 정보를 이용하여 인증이 이루어진다. 인증 여부를 확인 하기 위해서는 아래 명령어를 수행하면 확인이 가능하다.

```
# 인증 정보 확인
$ gcloud auth list


# 프로젝트 변경
$ gcloud config set project [PROJECT_ID]
```

- 서비스 리전(see page 24)
- 비용 산정(see page 24)