

애플리케이션을 VMware Pivotal Container Service로 이전하여 배포 가속화

최신 플랫폼을 통한 경쟁력 강화

애플리케이션을 컨테이너에서 실행할 수 있도록 패키징을 재구성한 다음 최신 플랫폼으로 이동하는 과정, 즉 애플리케이션의 플랫폼을 재구축하는 과정은 컨테이너 기술의 이점을 활용하는 데 매우 중요한 단계입니다. 플랫폼 재구축은 소프트웨어 개발을 가속화하고, 인프라 관리를 간소화하고, 배포를 자동화합니다. 플랫폼이 재구축된 애플리케이션은 배포 후 Kubernetes를 통해 온디맨드로 조정 및 확장할 수 있습니다.

Kubernetes를 통한 조정 및 최적화

Kubernetes는 애플리케이션 컨테이너를 관리 또는 조정하는 플랫폼입니다. 컨테이너는 애플리케이션 및 해당 종속성을 거의 어디에서나 실행할 수 있는 배포 가능한 이미지로 패키징하여 소프트웨어의 개발 및 배포를 간소화합니다. Kubernetes는 컨테이너형 애플리케이션을 조정하여 리소스 활용, 장애 처리, 가용성, 구성, 확장성 및 원하는 상태를 관리하고 자동화합니다.

애플리케이션과 해당 서비스가 가상 머신으로 구성된 분산형 클러스터의 컨테이너 안에서 실행되는 동안 Kubernetes는 컴퓨팅 리소스 사용을 최적화하고 애플리케이션을 원하는 상태로 유지하기 위해 모든 가변 요소를 동기화된 상태로 운영하도록 조정합니다. Kubernetes의 컨테이너형 워크로드 조정 기능은 애플리케이션 플랫폼 재구축의 이점을 활용하는 데 핵심적인 기능입니다.

플랫폼 재구축의 이점

애플리케이션의 플랫폼을 재구축하면 애플리케이션을 재설계하거나 재편하는 복잡한 과정을 수행하지 않고도 다음과 같은 애플리케이션 개발 및 배포의 가속화와 관련된 여러 목적을 달성할 수 있습니다.

- 워크로드 통합(특히 클라우드형 애플리케이션 개발 방향으로 전환 중인 경우)
- 무중단 통합 및 배포(CI/CD) 파이프라인을 통한 통합 간소화 및 개선
- 자동화, 보안, 수명주기 관리를 통한 애플리케이션 관리 운영 효율성 향상

플랫폼 재구축은 애플리케이션을 컨테이너에 패키징 재구성한 후에 진행되므로 다음과 같은 패키징 재구축의 이점도 얻을 수 있습니다.

- 개발, 테스트, 운영, 클라우드 환경에 걸친 이동성 확보
- 예측성 및 재현성을 통해 사용자마다 다른 결과가 발생하는 현상 제거
- 업그레이드, 패치 적용, 유지 보수 간소화
- 개발 반복, 테스트, 배포의 대응력 향상에 필요한 속도 확보
- 개발자가 때와 장소에 관계없이 익숙한 툴로 개발할 수 있는 유연성 확보
- 변경 불가능한 컨테이너 이미지에 대한 추적 기능을 통해 투명성, 규정 준수, 재사용 개선

Kubernetes의 이점

- 다른 워크로드에 영향을 주지 않으면서 간편하게 애플리케이션을 수정, 업데이트, 확장 또는 재배포
- 컨테이너화된 워크로드의 확장성 자동화
- 애플리케이션을 머신에서 분리하여 이동성과 유연성 확보
- 애플리케이션 배포, 로깅, 모니터링을 간편하고 신속하게 수행
- 효율적인 리소스 활용을 통해 서버를 통합하고 비용을 절감
- 자가 복구 및 고가용성을 통해 머신 장애를 효과적으로 처리

VMware Pivotal Container Service의 이점

- Kubernetes 클러스터의 온디맨드 프로 비저닝, 확장, 패치 적용, 업그레이드
- Google Kubernetes Engine(GKE)과 의 지속적인 호환성을 통해 안정적인 최 신 Kubernetes 릴리스에 대한 액세스 보장
- 연속적인 업데이트, 상태 점검, 자동 복구를 통해 Kubernetes 구성 요소 (마스터, 작업자, etcd 노드)에 고가용 성 제공
- 마이크로 세분화, 로드 밸런싱, 보안 정책 을 통해 고급 컨테이너 네트워킹 제공
- 컨테이너 이미지의 취약점 검색, 서명, 검 증을 통해 배포 보호

애플리케이션 플랫폼을 재구축하면 특히 일관되고 원활한 설치 프로세스를 위해 컨테이너 종속성을 포함하여 점점 더 많이 사전 패키징되고 있는 ISV(독립 소프트웨어 벤더) 제공 애플리케이션의 변경 사항을 활용할 수 있습니다.

VMware Pivotal Container Service

VMware® Pivotal Container Service(PKS)는 고급 네트워킹, 프라이빗 컨테이너 레지스 트리 및 전체 수명주기 관리 기능이 탑재된 운영 환경 수준의 Kubernetes 기반 컨테이너 서비스를 제공합니다. 이 솔루션은 Kubernetes 클러스터의 구축 및 운영을 대폭 간소화하 여 VMware vSphere 또는 퍼블릭 클라우드에서 컨테이너를 규모에 맞게 실행 및 관리할 수 있게 해 줍니다.

아키텍처

PKS는 Kubernetes, BOSH, VMware NSX®-T, Project Harbor를 결합하여 가용성 이 높은 컨테이너 서비스를 실현합니다. 인텔리전스와 통합을 내장한 PKS는 이러한 모든 오픈 소스 및 상업용 모듈을 하나로 통합하여 간편하게 사용할 수 있는 제품을 제공하여 Kubernetes를 효율적으로 구축하고 관리할 수 있도록 합니다.

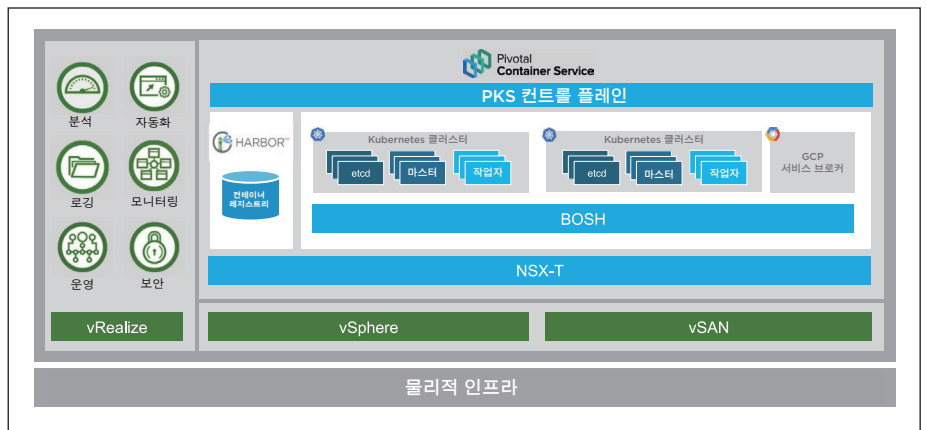


그림 1. VMware Pivotal Container Service의 아키텍처

BOSH를 통한 운영 및 자동화

BOSH는 대규모 분산 시스템의 배포 및 수명주기 관리를 간소화하는 릴리스 엔지니어링 을 위한 오픈 소스 툴이며, 개발자가 소프트웨어를 지속적이고 재현 가능한 방식으로 쉽게 버전 지정, 패키징화 및 배포하도록 지원합니다. BOSH는 VMware vSphere®, Google Compute Platform, Amazon Elastic Compute Cloud(EC2) 등 다양한 IaaS 공급업체 에 걸친 배포를 지원합니다.

VMware NSX를 통한 컨테이너 네트워킹

VMware NSX-T는 Kubernetes 클러스터에 마이크로 세분화, 로드 밸런싱, 수신 제어, 보 안 정책 등의 고급 컨테이너 네트워킹 및 보안 기능을 제공합니다. NSX는 Kubernetes에 포트 수준 네트워킹에 필요한 완벽한 레이어 2-7 네트워킹 서비스를 제공합니다. 컨테이너 와 포트의 마이크로 세분화와 온디맨드 네트워크 가상화를 통해 네트워크를 빠르게 구축할 수 있습니다.

VMware 클라우드형 솔루션 및 PKS에 대해 자세히 알아보기

VMware가 고객의 클라우드형 애플리케이션 빌드, 실행 및 관리를 지원하는 방법에 대해 자세히 알아보려면 <https://cloud.vmware.com/cloud-native-apps>를 방문하십시오.

VMware Pivotal Container Service에 대해 자세히 알아보려면 <https://cloud.vmware.com/pivotal-container-service>를 참조하십시오.

Project Harbor를 통한 이미지 레지스트리 보안

Harbor는 Docker 이미지를 방화벽으로 보호되는 프라이빗 레지스트리에 저장하고 배포하는 VMware의 오픈 소스 엔터프라이즈급 레지스트리 서버입니다. Harbor에는 역할 기반 액세스 제어, 컨테이너 이미지 취약점 검색, 정책 기반 이미지 복제, LDAP 또는 Microsoft Azure Directory과의 통합, 공증인 및 감사 서비스가 포함됩니다.

PKS에서 플랫폼 재구축을 위한 워크로드 지정

유연하고 강력한 기능을 갖춘 VMware PKS는 다음과 같은 워크로드의 플랫폼 재구축에 적합합니다.

- 데이터 지속성이 필요한 애플리케이션(예: MongoDB, CouchDB, Elasticsearch)
- 분산 클러스터로 관리되는 애플리케이션(특히 클러스터 내 노드가 서로 통신해야 하는 경우)
- 인프라 기본 작업이 필요한 애플리케이션(예: 영구 스토리지)
- 여러 개의 포트가 필요한 애플리케이션

PKS는 내장된 레지스트리를 통해 개발자가 컨테이너 이미지를 관리하고, Kubernetes용 컨테이너 및 포트 템플릿을 구축하고, 개발자가 원하는 포트 바인딩을 구성하고, 종속성을 관리할 수 있도록 지원하는 서비스를 제공합니다. 따라서 PKS는 Elasticsearch, Spark와 같은 최신 데이터 서비스 및 사용자 지정 스택 또는 인프라 기본 작업에 대한 액세스를 필요로 하는 기타 애플리케이션의 플랫폼을 재구축하는 데 이상적입니다.

모놀리식 구성을 단계별로 분할

VMware PKS를 통해 애플리케이션의 플랫폼을 재구축한 다음 단계별로 이를 세 가지 구성 요소로 분할할 수 있습니다. 첫 번째 단계에서는 모놀리식 구조에서 데이터베이스를 분리하여 독립적인 확장성을 부여할 수 있습니다. 두 번째 단계에서는 사용자 인터페이스 및 명령줄 인터페이스를 포함한 애플리케이션의 프론트엔드를 분리하여 별도로 관리 및 업데이트할 수 있게 합니다. 세 번째 단계에서는 보안에 중점을 두어 구성 요소 간 통신을 보호합니다.

요약

VMware Pivotal Container Service는 컨테이너 네트워킹, 보안 및 수명주기 관리 기능이 탑재된 운영 환경 수준의 고가용성 Kubernetes 기반 컨테이너 서비스를 제공합니다. vSphere를 통한 사내 및 Google Cloud Platform과 같은 퍼블릭 클라우드에 배포 가능한 VMware PKS는 컨테이너화 및 조정의 이점을 활용할 수 있는 애플리케이션 플랫폼 재구축에 적합합니다.

