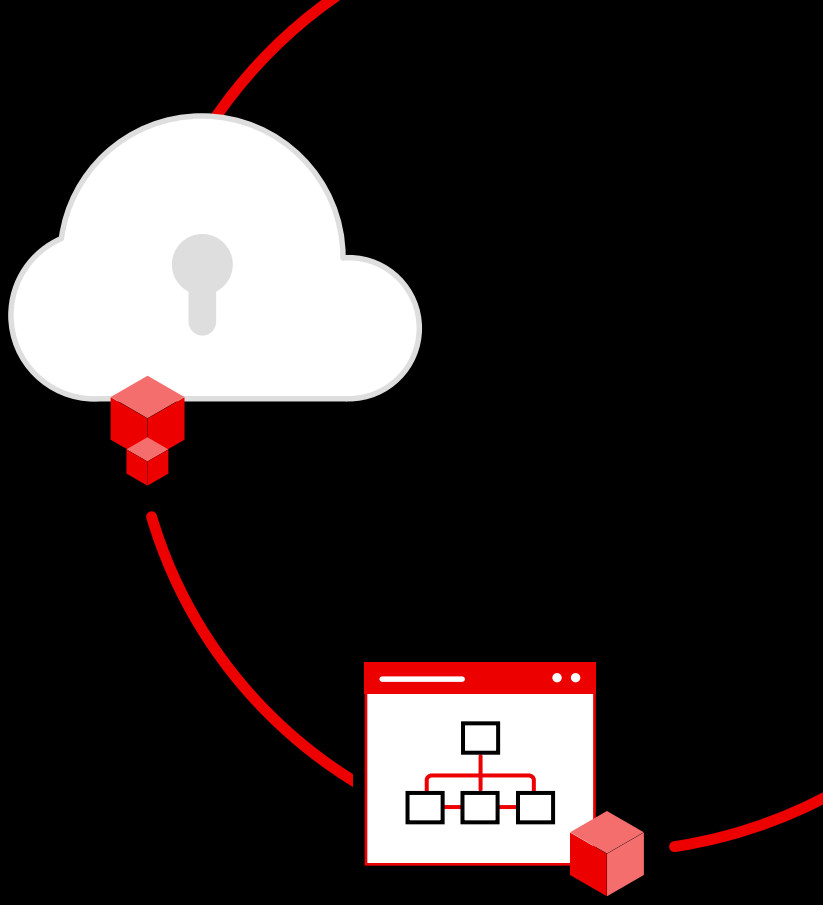
Red Hat
OpenShift
Platform Plus

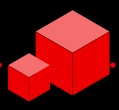
DevSecOps 지원을 위한 소프트웨어 팩토리 구축

DevSecOps 여정을 시작하기 위한 확고한 가이드

목차



DevSecOps로 비즈니스 보호



점점 더 많은 조직이 혁신과 **디지털 트랜스포메이션**을 위해 **클라우드 네이티브**, **컨테이너**, **마이크로서비스** 기술을 도입하고 있습니다. 또한 이러한 트랜스포메이션의 일환으로 컨테이너 오케스트레이션을 위한 쿠버네티스를 사용하여 클라우드 네이티브 운영을 지원하고 있습니다. **쿠버네티스 클러스터**는 온사이트와 클라우드 환경 전체로 호스트를 확장할 수 있기 때문에 신속한 확장과 탄력성을 갖춘 운영을 요하는 클라우드 네이티브 애플리케이션을 호스팅하기에 매우 적합한 플랫폼입니다.

그렇지만 이러한 방식은 특히 규모에 따른 보안과 관리 용이성 측면에서 새로운 문제점을 야기합니다. 실제로 기업의 고위 IT 리더 중 50%가 기술 이니셔티브를 위한 3대 우선순위 가운데 하나로 사이버보안을 꼽았습니다.[1]

DevSecOps 접근 방식과 사례를 도입하면 애플리케이션, 프로세스, 플랫폼에 보안을 구축하여 비즈니스를 더욱 효과적으로 보호할 수 있습니다.

이 e-book에서는 Red Hat® OpenShift®와 기타 Red Hat 기술을 통해 조직 내에 성공적으로 DevSecOps 사례를 구축하기 위한 고려 사항에 대해 논의하고 지침을 제공합니다.

클라우드 네이티브 애플리케이션이란?

클라우드 네이티브 애플리케이션은 탄력적으로 결합된 소규모의 독립적인 서비스 컬렉션입니다.

DevOps와 DevSecOps란?

DevOps는 신속하고 자동화된 고품질 서비스 제공을 통해 비즈니스 가치와 대응력을 향상시키도록 지원할 수 있는 문화, 자동화, 플랫폼 설계에 대한 접근 방식입니다. **DevSecOps**는 DevOps의 협업 문화를 확장하여 애플리케이션 라이프사이클 전체에 보안을 통합합니다. 여기에는 분산된 환경에 더욱 광범위하게 보안을 적용할 수 있도록 하는 인력, 프로세스, 기술이 모두 포함됩니다.

DevSecOps를 통해 보안은 한 팀이 소유하며 개발 및 배포 프로세스 마지막에 적용되는 일련의 태스크가 아니라 여러 팀 전반에 공유하고 시행하는 책임이 됩니다. 따라서 보안, 개발, 운영 팀의 인력이 협력하여 정보, 피드백, 지식, 인사이트를 공유합니다. 이러한 접근 방식을 통해 애플리케이션 개발과 인프라 배포 시작 시점부터 보안을 통합하여 보호 성능을 강화하고 리스크를 줄일 수 있습니다.

88%

쿠버네티스를 컨테이너 오케스트레이터로 활용한다고 답한 기업의 비율(74%는 프로덕션 단계에서 사용 중)[2]

74%

DevSecOps 이니셔티브를 추진하고 있다고 답한 기업의 비율[2]

1 Flexera. "2021 기술 지출 현황 Flexera 보고서(2021 Flexera State of Tech Spend Report)," 2021년 1월.

2 Red Hat, "쿠버네티스 보안 현황 리포트," 2021년.

DevSecOps의 목표

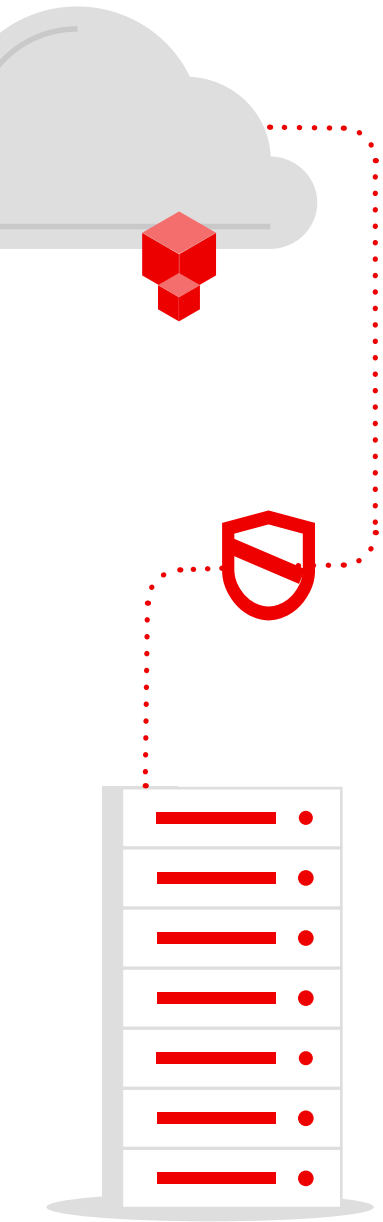
DevSecOps의 목표는 보안에 중점을 둔 고품질의 애플리케이션, 서비스, 기능을 규모에 맞춰 빠르게 제공하고 배포하는 것입니다.

확장

속도

보안

안정성



DevSecOps 구현에 따르는 과제

수동 프로세스

개발, 테스트, 보안 관련 작업은 잦은 인적 개입이 필요한 경우 시간이 많이 소요되고, 반복적이며, 오류가 발생하기 쉽고, 시행하기가 어렵습니다.

팀 간 협업 제한

개발, 보안, 운영 팀은 관련 영역 내에서만 작업을 수행하는 경우가 많기 때문에 프로세스가 단절되고, 수동 핸드오프가 필요하며, 다른 팀의 문제점이나 요구 사항에 대해 제대로 알고 이해하기 어렵습니다.

뒤늦은 보안 프로세스 적용

기존의 애플리케이션 개발과 출시 접근 방식은 보안 사례와 점검을 프로덕션 단계로 배포하기 직전인 프로세스의 마지막에 적용합니다.

복잡한 애플리케이션 환경

컨테이너, 마이크로서비스, 클라우드 서비스 등 대규모의 복잡한 애플리케이션 개발, 테스트, 프로덕션 환경을 구성하는 다양한 구성 요소의 연결과 보안이 미치는 영향을 모두 이해하기란 매우 어려울 수 있습니다.

외부 종속성

클라우드 네이티브 애플리케이션 개발은 거의 항상 오픈소스 코드, 라이브러리, 서비스 섹션을 포함하여 보안이 필요한 몇 가지 외부 종속성에 의존합니다.

보안 환경의 진화

비즈니스, 기술, 지리적 요구 사항을 비롯한 보안 위협과 규제는 매우 빠른 속도로 변화하고 있어 항상 최신 정보를 갖추고 컴플라이언스를 유지하기가 어렵습니다.

핵심 요소인 인력, 프로세스, 기술

DevSecOps는 하나의 팀이나 단일 프로세스가 아닌 인력, 프로세스, 기술이라는 세 가지 영역에서의 변화와 연계가 필요한 전사적인 역량입니다.

피플(People)

전사적 이니셔티브의 핵심은 구성원이며 DevSecOps도 이와 다르지 않습니다. 조직 전반에 DevSecOps를 도입하려면 개발, 보안, 운영을 포함한 모든 팀이 참여하고 서로 신뢰해야 합니다.

프로세스

프로세스는 처음부터 끝까지 프로젝트를 진행시킵니다. 광범위한 DevSecOps 도입을 위해서는 애플리케이션과 인프라를 구축, 배포, 관리, 조정하고 라이프사이클 전반에 보안을 통합하기 위한 명확한 프로세스가 반드시 필요합니다.

기술

애플리케이션 플랫폼은 애플리케이션과 인프라를 구축, 배포, 실행하기 위한 역량을 제공합니다. 개발, 보안, 운영 팀을 지원하는 통합 플랫폼을 통해 DevSecOps 사례를 구축하고 조정할 수 있는 기반을 갖출 수 있습니다.

조직의 DevSecOps 성공을 위한 준비

어떤 조직도 하룻밤 사이에 DevSecOps 사례를 완벽하게 구축할 수는 없습니다. DevSecOps 도입은 양자택일의 문제가 아니라 반복적인 학습 여정입니다. 여정을 안내하고 시간이 지남에 따라 학습하도록 지원하는 논리적이고 지속 가능한 전략이 필요합니다.

팀 간 협업 권장

인센티브를 사용하고 프로세스를 설계하여 조직 전반의 협업을 촉진합니다. 팀은 공동 작업을 통해 전체 DevSecOps 워크플로우를 생성하여 더 많은 가치를 제공할 수 있습니다. 또한 다른 팀과 협업하면서 개발, 보안, 운영에 대한 소유권을 공유하고 책임감을 강화할 수 있습니다.

현재 상태에 대한 도큐멘테이션

GitOps와 같은 다이나믹 프레임워크를 사용하여 기존의 개발, 변경 관리, 거버넌스 프로세스를 상세히 도큐멘테이션합니다. 현재 상황과 과제를 파악하면 앞으로 나아갈 계획을 세우는 데 도움이 됩니다. 프로세스를 조정할 때 새로운 프로세스와 변경된 사유를 도큐멘테이션해야 합니다.

프로세스 점검

DevSecOps 목표를 지원하지 않는 프로세스를 식별하고 조정합니다. 여기에는 비효율적이거나 서로 다른 지속적 통합/지속적 제공(CI/CD) 설정과 인프라, 과도하게 중앙화된 프로세스, 빈번한 수동 개입을 필요로 하는 프로세스가 포함됩니다.

성공의 정의와 측정

조직에 적합한 성공적인 DevSecOps를 결정하고 진행 사항을 추적할 수 있는 측정 가능한 메트릭 또는 핵심 성과 지표(KPI)를 식별합니다. 이러한 메트릭은 애플리케이션 빌드와 배포 기간, 변경 사항 릴리스와 결함률, 문제 해결 기간, 애플리케이션 가용성 등이 될 수 있습니다.

지식과 모범 사례 공유

흔히 CoP(Community of Practice) 또는 CoE(Center of Excellence)라 불리는 핵심 이해관계자 팀을 만들어 DevSecOps 모범 사례, 경험 그리고 성과를 조직 전체에 공유합니다. 이 팀은 DevSecOps를 도입하고 시작할 준비가 된 다른 팀도 지원해야 합니다.

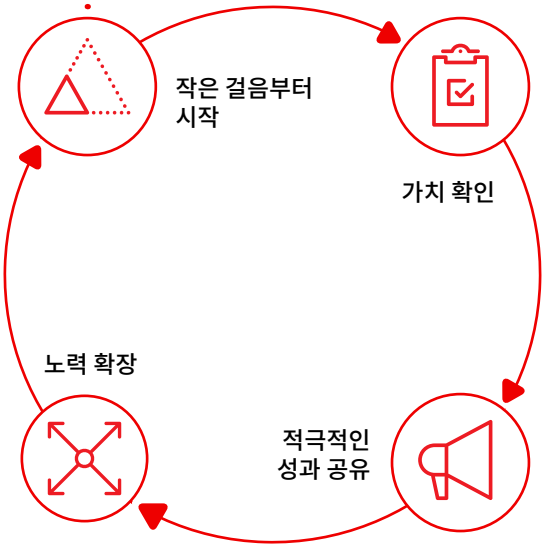
조직 전반의 노력

조직 내 모든 구성원이 DevSecOps 도입을 위해 최선을 다해야 합니다. 각 팀이 변화를 추진하는 이유를 이해하도록 돕고 팀의 역할에 미칠 긍정적인 영향을 강조합니다. 경영진의 적극적인 지지와 메트릭 기반의 인센티브는 모든 팀이 여정을 지속해 나가는 데 도움이 됩니다.

DevSecOps 사례 시작하기

DevSecOps 전략을 정의했다면 이제 DevSecOps를 시작할 수 있습니다. 모든 개발팀이 즉시 DevSecOps를 도입할 준비가 되지는 않을 것입니다. 새로운 프로세스와 플랫폼을 도입하여 측정 가능한 수준의 성공을 달성하고 있는 팀부터 시작하면 됩니다. 이러한 팀의 구성원은 핵심 이해관계자 팀에도 적합한 후보자가 될 수 있습니다.

소규모로 시작해 자동화의 가치를 확인하고 신중하게 확장한 후 이 과정을 반복합니다. 단기간에 성과가 누적될 수 있도록 합니다. 메트릭을 사용해 진행 사항을 모니터링하고 효과적이지 않은 프로젝트나 프로세스를 통해 학습합니다. 성과를 얻을 때마다 DevSecOps의 가치를 적극적으로 알리고 조직 전반에 경험을 공유합니다. 이를 통해 모든 팀이 이같은 경험을 바탕으로 더 많은 가치를 창출하기 위한 발판을 마련할 수 있습니다.



소프트웨어 제공을 위한 팩토리 접근 방식

현대적인 소프트웨어 제공은 속도, 일관성, 품질에 의존합니다. 소프트웨어 팩토리 접근 방식은 조직 내에 DevSecOps 문화를 도입하기 위해 필요한 행동의 변화와 다양한 조치를 지원, 가속화, 시행하는 데 도움을 줍니다. 이러한 접근 방식으로 **신뢰할 수 있는 소프트웨어 공급망**과 테스트 기반 개발과 같은 일관된 애자일 프로세스 세트를 활용하여 고품질의 애플리케이션을 신속하게 개발하고 배포할 수 있습니다.

소프트웨어 팩토리의 장점

소프트웨어 팩토리 접근 방식은 측정 가능한 장점을 제공합니다.

변경 리드 타임 단축

높은 배포 빈도

장애가 발생한 서비스 복원 기간 단축

낮은 변경 실패율

수치로 보는 소프트웨어 제공 성과 메트릭[3]

소프트웨어 제공 성과 메트릭	소프트웨어 팩토리 사용 시	소프트웨어 팩토리 미사용 시
변경 리드 타임	1시간 미만	1~6개월
배포 빈도	온디맨드(1일 1회 미만)	1~6개월 마다
서비스 복원 기간	1시간 미만	1일~1주일
변경 실패율	0%~15%	16%~30%

*3 Google Cloud. "*가속화: 2021년 DevOps 현황(Accelerate: State of DevOps 2021)*," 2021년 9월.*

소프트웨어 팩토리란?

소프트웨어 팩토리는 일관되지 않은 수동 프로세스를 일관되고 자동화된 운영으로 바꿉니다.

소프트웨어 팩토리 미사용 시

수동 프로세스와 승인으로 인해 개발과 배포가 늦어지고, 예상되는 결과가 명확하지 않으며, 보안이 일관되게 시행되지 못합니다. 매우 소규모의 변경 사항이라도 구현하기까지 수일에서 수주가 걸리기 때문에 관련 팀에서는 단일 배포를 통해 대규모 변경 작업을 수행하려는 경우가 많습니다. 이로 인해 변경 사항 적용에 실패하고 보안 문제가 발생할 위험이 높아집니다.

프로세스 전체의 투명성이 부족하기 때문에 팀 간 신뢰도가 굉장히 낮은 경우가 많습니다. 보안과 컴플라이언스 관련 조치는 프로세스 후반에 수동으로 적용되기 때문에 개발 중에 발생할 수 있는 문제를 식별하지 못할 수 있습니다. 따라서 예상치 못한 보안과 컴플라이언스 문제를 수정하기 위해 애플리케이션이 개발자에게 되돌아오기도 합니다. 이와 같은 예외적인 상황은 이미 스트레스가 과도한 개발자에게 불신과 불만을 초래할 수 있습니다.

소프트웨어 팩토리 사용 시

명확하고 자동화된 프로세스로 개발과 배포 속도를 높이고, 보안을 일관되게 시행하며, 관련된 모든 팀이 명확한 결과를 예측하고 설정할 수 있습니다. 소규모 변경 사항의 경우 몇 분 내로 롤아웃할 수 있기 때문에 팀에서 매일 많은 소규모 변경 사항을 빠르게 배포할 수 있으므로 전반적인 리스크가 줄어듭니다.

소프트웨어 팩토리의 핵심 기능은 투명성과 가시성으로, 개발, 운영, 보안 팀 간의 신뢰를 쌓는 데 도움이 됩니다. 보안과 컴플라이언스 조치는 개발 중에 자동으로 적용되어 프로세스 초기 단계에서 문제점을 파악하고 수정할 수 있습니다. 프로세스와 정책을 도큐멘테이션하여 팀이 프로세스 전반에 대해 예상되는 결과를 이해하고 애플리케이션을 프로덕션 단계로 배포하는 시점에 발생할 수 있는 예외적인 상황을 방지할 수 있습니다.

자체 소프트웨어 팩토리 구축

자동화는 소프트웨어 팩토리 접근 방식의 핵심입니다. 자동화는 클라우드 네이티브 환경을 운영하고 DevSecOps 사례를 도입하기 위해 반드시 필요합니다. 자동화로 개발, 제공, 배포, 인프라 운영의 규모를 통제된 방식으로 확장할 수 있습니다. 또한 리소스, 환경, 애플리케이션을 동적으로 프로비저닝하거나 사용 종료할 수 있습니다. 따라서 조직은 변화에 더욱 빠르게 대응할 수 있습니다.

개발 테스트, 코드 품질 관리, 컴플라이언스 확인, 취약점 감지, 문제 해결 프로세스 등 DevSecOps 워크플로우의 모든 측면에 대한 자동화를 고려하세요. CI/CD 파이프라인을 사용하여 애플리케이션 개발과 개선을 모두 자동화하고, 인프라 배포와 관리까지 자동화하세요. 보안과 리스크 정책을 정의하여 도큐멘테이션하고, 소프트웨어 라이프사이클 전반에서 이러한 정책에 따른 컴플라이언스 점검과 문제 해결을 자동화하세요.

선언적인 인텐트 기반 자동화는 더욱 빠르고 간편하게 확장하고 조정할 수 있도록 지원합니다.

선언적 자동화를 활용하면 리소스 설정을 위한 지침이 아닌 원하는 애플리케이션 또는 인프라 구성을 정의할 수 있습니다. 목표에 도달하기 위한 수단이 아니라 최종 목표 자체를 설명하면 됩니다. 그러면 애플리케이션 플랫폼이 원하는 상태에 도달하기 위해 필요한 리소스를 프로비저닝하고 구성하게 됩니다. 또한 시간이 지남에 따라 리소스가 올바르게 구성된 상태를 유지하도록 자체적으로 문제를 해결합니다. 마지막으로, 이러한 접근 방식은 Git 버전 제어 시스템을 사용하여 인프라와 애플리케이션 구성을 관리하기 위한 일련의 사례를 의미하는 **GitOps**를 준비할 수 있도록 합니다.

자동화 대상과 시기 결정

DevSecOps와 마찬가지로 자동화 배포 역시 하나의 여정이며 계획이 필요합니다. 다음 단계를 따라 자동화를 시작하세요.

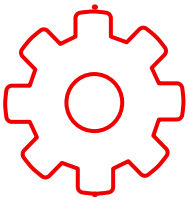
1. 프로세스를 상세히 도큐멘테이션합니다.
2. 프로세스의 각 수동 단계에서 결정되는 내용과 해당 결정을 내린 방식을 기록합니다. 의사결정에는 특정 자료를 읽고, 특정 요인을 고려하고, 다양한 전문가와 논의하거나 기타 조치를 고려하는 과정이 수반될 수 있습니다.
3. 간편하게 자동화할 수 있는 모든 수동 단계를 식별하고 자동화되어야 하는 변경 수준을 평가합니다. 예를 들어, 소규모 변경 사항을 자동화할 수도 있지만, 더 큰 규모의 변경을 위해 특정 팀의 승인이 필요할 수도 있습니다.
4. 간편하게 자동화하기 어려운 수동 단계의 경우, 이를 자동화하는 데 필요한 사항을 평가하고 자동화 구현을 위한 계획을 수립합니다.

자동화를 구현할 수 있는 모든 영역을 파악할 때까지 기다리지 말고 즉시 자동화를 시작하세요. 프로세스를 반복적으로 자동화하는 것 자체가 DevOps 프로세스입니다. 프로세스를 자동화, 조정, 개선하는 과정에서 전반적인 DevSecOps 사례를 지원할 유용한 기술과 경험을 얻을 수 있습니다.

흥미로운 작업에 집중

자동화는 인력을 대체하기 위한 것이 아니라, 생산성, 일관성, 효율성에 중점을 둡니다. 이것이 바로 자동화의 역설로서, 자동화를 사용할수록 인력의 개입은 더 중요해지고 그 빈도는 줄어듭니다.

자동화를 일자리를 줄이는 툴로 바라보는 일부 견해도 있지만, 실제로는 숙련된 IT 직원이 일상적이고 반복적인 작업 대신에 더욱 생산적인 업무에 초점을 맞추고 개선해야 하는 문제를 발견하여 해결책을 찾을 수 있도록 지원합니다.



기업 전반의 자동화에 대해 알아보기

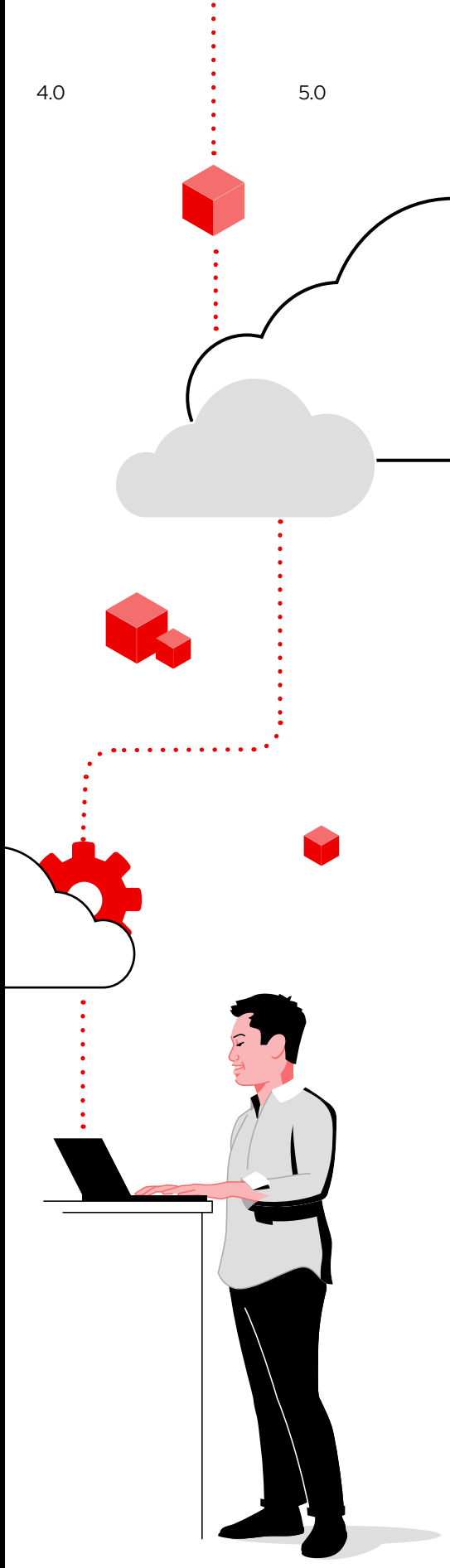
자동화는 구성원과 프로세스, 기술을 통합하여 비즈니스 민첩성과 혁신, 가치를 높일 수 있습니다.

자동화된 엔터프라이즈 e-book에서 조직 전반에 자동화를 도입하는 방법을 알아보세요.

소프트웨어 팩토리를 위한 툴

툴은 소프트웨어 팩토리에서 매우 중요한 부분입니다. Red Hat은 소프트웨어 팩토리 내에서 다음과 같은 카테고리의 툴을 사용하고 자동화할 것을 권장합니다. 각 툴 유형에 해당하는 예시가 나와있지만, 다른 툴을 사용해도 됩니다.

툴 카테고리	예
프로젝트 관리	▸ Confluence와 Jira ▸ Trello
소스 코드 관리(SCM)	▸ Github ▸ Gitlab
통합 개발 환경(IDE)	▸ VS.code ▸ **Red Hat OpenShift Dev Spaces**
아티팩트 리포지토리	▸ Nexus ▸ Artifactory
CI/CD	▸ **Red Hat OpenShift Pipelines** ▸ **Jenkins**
런타임	▸ **Red Hat Runtimes** ▸ Golang
빌드	▸ Maven ▸ Dotnet build
유닛 테스트	▸ JUnit ▸ NUnit
소스 코드 분석	▸ Sonarqube ▸ Fortify
정적 애플리케이션 보안 테스트(SAST)	▸ CheckMarx ▸ **Red Hat Advanced Cluster Security for Kubernetes**
사용자 수용 테스트	▸ Cucumber ▸ Cyprus
동적 애플리케이션 보안 테스트(DAST)	▸ Veracode ▸ Synopsys
텔레메트리, 메트릭, 로깅	▸ **Prometheus** ▸ **Grafana** ▸ **Elasticsearch, Fluentd, Kibana(EFK)** ▸ Splunk
서비스 메쉬	▸ Linkerd ▸ **Red Hat OpenShift Service Mesh**



빌드, 배포, 실행

플랫폼 아키텍트나 DevOps 엔지니어는 개발자를 대신하여 소프트웨어 팩토리를 구성하는 경우가 종종 있습니다. 소프트웨어 팩토리를 구축하는 경우 빌드, 배포, 실행의 세 가지 영역에서 보안 관련 모범 사례를 살펴보세요.

빌드

애플리케이션 보안과 컴플라이언스를 제어합니다.

클라우드 네이티브 배포에서는 애플리케이션 자체에 보안을 구축하는 과정이 매우 중요합니다.

- 런타임을 포함하여 외부 컨테이너와 애플리케이션 콘텐츠에 대해 신뢰할 수 있는 소스를 사용합니다.
- 신뢰할 수 있는 프라이빗 컨테이너 레지스트리를 도입하여 이미지를 관리합니다.
- 개발과 배포 파이프라인을 자동화합니다.
- TDD와 같은 애자일 사례를 사용하여 코드 내에 비기능적 요구 사항을 구현합니다.
- 코드 품질, 이미지 취약성, 쿠버네티스 배포 분석으로 애플리케이션 파이프라인에 보안을 통합합니다.
- 애플리케이션 배포와 배치를 자동화합니다.

배포

플랫폼을 보호합니다.

효과적인 보안을 구현하려면 쿠버네티스 플랫폼을 보호하고 배포 정책을 자동화해야 합니다.

- 컨테이너에 최적화된 운영 체제를 사용하여 공격 표면을 줄입니다.
- 클러스터 전반의 구성 관리와 정책 시행을 자동화합니다.
- 정교한 역할 기반 액세스 제어(RBAC)로 최소 권한 액세스를 구현합니다.
- 전송 중이거나 유휴 상태에 있는 플랫폼과 애플리케이션 데이터를 암호화합니다.
- 자동화된 컴플라이언스, 리스크 평가, 문제 해결 솔루션을 사용합니다.
- 쿠버네티스 포드 권한 제어 정책을 사용하여 배포 리스크를 줄입니다.

실행

컨테이너 런타임을 보호합니다.

런타임 시 애플리케이션 보안을 유지합니다.

- SELinux(Security-Enhanced Linux®), 보안 컨텍스트 제약 조건(SCC), 쿠버네티스 네임스페이스, RBAC, 네트워크 정책을 사용하여 실행 중인 애플리케이션을 격리합니다.
- 할당량을 사용하여 리소스 충돌과 관련 성능 문제를 방지합니다.
- SSO(Single Sign-On) 사용자 관리, 인그레스와 이그레스 보안 관리, 암호화된 포드 간 트래픽, API 관리를 사용하여 애플리케이션 액세스를 관리하고 애플리케이션 데이터를 보호합니다.
- 플랫폼과 애플리케이션 활동을 감사하고 모니터링합니다.
- 위협 감지를 자동화하고 비정상 동작, 권한 에스컬레이션 이벤트, 암호화폐 채굴과 같은 위험한 프로세스에 관여하는 포드에 대응합니다.
- 권한 컨트롤러를 사용하여 보안 정책을 준수하지 않는 컨테이너 배포를 방지합니다.
- 서비스 메쉬와 네트워크 정책을 사용하여 제로 트러스트 네트워크를 구축합니다.

보안 팁

컨테이너 및 쿠버네티스 보안에 대한 계층화된 접근 방식을 읽고 쿠버네티스로 관리하는 컨테이너화된 애플리케이션을 보호하는 방법에 대해 자세히 알아보세요.

빌드	배포	실행
애플리케이션 라이프사이클	**구성 관리**	**관측성 및 경고**
취약점 분석	정책 권한 컨트롤러	런타임 동작 분석
애플리케이션 구성 분석	컴플라이언스 평가	네트워크 정책 권장 사항
CI/CD 통합을 위한 API	리스크 프로파일링	위협 감지 및 대응
신뢰할 수 있는 콘텐츠	쿠버네티스 플랫폼 라이프사이클	컨테이너 격리
컨테이너 레지스트리	Identity 및 액세스 관리	네트워크 격리
빌드 관리	플랫폼 데이터	애플리케이션 액세스 및 데이터
CI/CD 파이프라인	배포 정책	관측성

DevSecOps

전문가를 통한 DevSecOps 구현

Red Hat은 인증된 파트너 에코시스템, 광범위한 전문성, 혁신적인 플랫폼을 통해 하이브리드 클라우드 환경 전반의 애플리케이션을 빌드, 보호, 배포합니다. Red Hat은 수년간 업계 모범 사례와 오픈소스 기술을 사용하여 엔터프라이즈 조직이 기술 및 비즈니스 관련 과제를 해결할 수 있도록 지원해 왔습니다.

Red Hat 플랫폼은 신뢰할 수 있는 콘텐츠 공급망, 전담 보안팀의 지원, 핵심 보안 기능 백포트를 통해 DevSecOps 솔루션을 위한 이상적인 기반을 제공합니다. 또한 **교육 및 자격증 과정**, **인터랙티브 랩**, **컨설팅 서비스**, **관리형 제품**을 제공하여 성공적인 DevSecOps 사례를 빠르게 구축할 수 있도록 지원하고 있습니다.

Red Hat은 DevSecOps 여정의 모든 단계에서 조직을 지원합니다.

Red Hat의 입증된 오픈소스 플랫폼과 전문 서비스를 통해 지금 바로 필요한 솔루션을 배포하고 향후 변화에 맞춰 조정할 수 있으며, 효율적이고 효과적인 DevSecOps 도입에 필요한 다양한 방법과 접근 방식을 익힐 수 있습니다.

DevSecOps에 Red Hat을 선택해야 하는 이유에 대해 **자세히 알아보세요.**

DevSecOps 투자 가치 극대화

Red Hat Services는 DevSecOps 사례를 시작하고, 가속화하고, 확장하는 데 필요한 모든 리소스를 제공합니다.

- **Red Hat Open Innovation Labs**
  고객과 Red Hatter가 협력하여 DevSecOps와 같은 새로운 작업 방식을 학습하고 동시에 비즈니스 성과를 제공하는 레지던스 환경의 컨설팅 서비스입니다.
- **Red Hat Services Solution: DevSecOps**
  모듈식 접근 방식을 사용하여 소프트웨어 팩토리 구현을 돕는 서비스입니다.
- **Red Hat Services 여정: 컨테이너 도입**
  주요 작업 흐름에서 컨테이너 도입을 지원하는 컨설팅 서비스입니다.
- **Red Hat Services 여정: 자동화 도입**
  전사적인 자동화 도입 여정을 관리하기 위한 프레임워크를 제공하는 컨설팅 서비스입니다.

성공적인 DevSecOps를 위한 플랫폼 배포

Red Hat OpenShift Platform Plus는 DevSecOps를 위한 기술 기반과 확고한 프레임워크를 제공합니다. 이는 온사이트와 클라우드 인프라 전반에서 일관되게 운영하고 확장할 수 있는 혁신적인 애플리케이션 플랫폼입니다. Red Hat OpenShift Platform Plus는 환경 전반에서 애플리케이션을 빌드, 배포, 실행, 보호, 관리하기 위한 일관된 방식을 선도적인 엔터프라이즈 쿠버네티스 플랫폼과 결합합니다. 멀티클러스터 관리 툴은 쿠버네티스 클러스터에 대한 완벽한 가시성과 제어 능력을 제공합니다. 쿠버네티스 네이티브 보안과 DevSecOps 기능은 소프트웨어 공급망, 인프라, 워크로드를 보호합니다. 확장 가능하며 전 세계적으로 분산된 레지스트리와 클러스터 데이터 관리는 환경과 정보를 보호합니다.

오픈 통합 인터페이스와 Red Hat의 **인증된 파트너 에코시스템**을 통해 Red Hat OpenShift Platform Plus로 기존 및 신규 개발, 테스트, 운영, 보안 툴을 모두 활용할 수 있습니다. 많은 공급업체가 Red Hat 플랫폼에서 소프트웨어를 간편하게 설치하고 관리할 수 있도록 **인증된 Red Hat OpenShift 오퍼레이터** 또는 **인증된 소프트웨어 컨테이너**를 제공합니다. **Red Hat Marketplace**에서 직접 소프트웨어 제품을 구입하여 배포할 수도 있습니다. 마지막으로, Red Hat은 주요 클라우드 공급업체 파트너와 협력하여 배포와 운영을 간소화하는 동시에 사내 구축 비용을 절감하는 전체 관리형 **Red Hat OpenShift 클라우드 서비스**를 제공합니다.

Red Hat OpenShift Platform Plus 구성 요소

Red Hat OpenShift

Red Hat OpenShift는 자동화된 풀스택 오퍼레이션으로 하이브리드 클라우드와 엣지 배포를 관리하는 엔터프라이즈급 쿠버네티스 애플리케이션 플랫폼입니다. 생산성과 속도를 향상하는 개발자 중심의 기능을 포함합니다.

Red Hat Advanced Cluster Management for Kubernetes

Red Hat Advanced Cluster Management for Kubernetes는 내장된 거버넌스와 애플리케이션 라이프사이클 관리 기능으로 쿠버네티스 도메인 전체에 대한 가시성을 제공하는 콘솔입니다.

Red Hat Advanced Cluster Security for Kubernetes

Red Hat Advanced Cluster Security for Kubernetes는 애플리케이션 라이프사이클 전체에서 인프라와 워크로드 보호, 가시성을 향상하는 쿠버네티스 네이티브 보안 기능을 제공하는 솔루션입니다.

Red Hat Quay

Red Hat Quay는 스토리지를 제공하고 데이터센터와 클라우드 환경 전반에 컨테이너를 빌드, 분산, 배포하도록 지원하는 오픈소스 컨테이너 이미지 레지스트리 플랫폼입니다.

Red Hat OpenShift Data Foundation

Red Hat OpenShift Data Foundation은 Red Hat OpenShift 환경에 대한 데이터 효율성, 복원력, 보안을 제공하는 확장 가능한 데이터 및 스토리지 서비스 레이어입니다.

Red Hat OpenShift Platform Plus는 DevSecOps 여정의 모든 지점에서 귀사를 지원합니다. 현재 단계에서 지원하며 조직의 속도에 맞춰 진행할 수 있는 기반을 제공합니다.

내장된 보안 기능

전체 애플리케이션 라이프사이클에 걸쳐 적용하고 실행할 수 있는 60개 이상의 내장된 보안 정책뿐 아니라 시스템 수준의 데이터 수집과 분석으로 실행 중인 워크로드의 보안 문제 또는 위협을 모니터링합니다.

일관된 운영

온사이트 데이터센터와 클라우드 인프라 전반의 Red Hat OpenShift 클러스터에 보안, 구성, 컴플라이언스, 거버넌스를 위한 일관된 운영 정책을 적용합니다.

개발자 툴

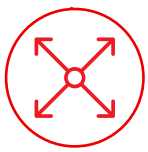
지원되는 빌드 툴, 언어, 파이프라인, 프레임워크 라이브러리가 포함되어 애플리케이션을 더욱 빠르게 구축, 실행, 배포할 수 있습니다. Red Hat OpenShift와 함께 실행하도록 테스트와 검증된 최신 개발자 툴을 오퍼레이터 프레임워크에서 통합할 수 있습니다.

엔드 투 엔드 관리

다양한 쿠버네티스 배포에 기반하는 환경을 비롯하여 온사이트, 클라우드, 엣지 환경 전반에서 작동하는 관리자와 개발자를 위한 통합 인터페이스로 Red Hat OpenShift 환경을 일관되게 관리합니다.

DevSecOps에 대한 지원

선언적 보안을 개발자 툴링과 워크플로우에 통합합니다. 쿠버네티스 네이티브 제어를 사용해 위협을 완화하고 보안 정책을 실행하여 운영상의 리스크를 최소화합니다.



확장 가능한 데이터 서비스

클러스터 전반에서 데이터 관리를 간소화합니다. 파일, 블록, 오브젝트 데이터 프로토콜을 지원하는 Red Hat OpenShift Data Foundation이 스테이트풀 애플리케이션과 클러스터 서비스를 위한 복원력을 갖춘 퍼시스턴트 스토리지를 제공합니다.

제로 트러스트 네트워크 기능

제로 트러스트 네트워크를 구현하여 애플리케이션과 서비스 간에 복원력을 갖추고, 안전하며, 관측 가능한 커뮤니케이션을 제공합니다. **Red Hat OpenShift Service Mesh**는 Red Hat OpenShift에 포함되고 통합되어 커뮤니케이션을 더욱 쉽게 보호할 수 있도록 지원합니다.

Red Hat OpenShift Platform Plus는 효과적인 DevSecOps 도입을 위해 필요한 기술과 기능을 제공합니다. **Red Hat OpenShift 보안 가이드**를 읽고 기술 스택 전반에서 보안을 시행하는 방법에 대해 살펴보세요.

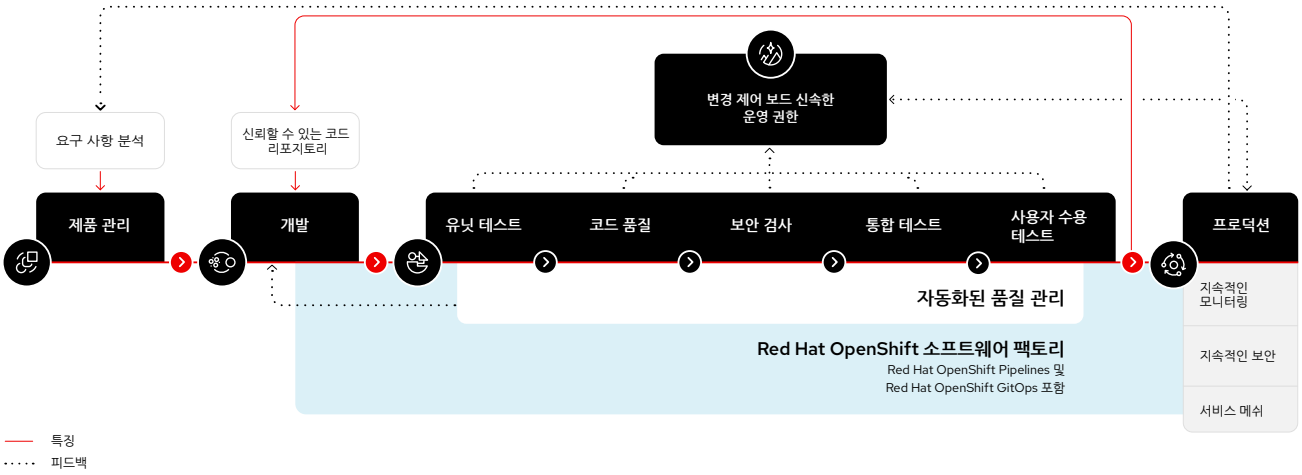
Red Hat OpenShift 클라우드 서비스를 더욱 빠르게 시작하세요

Red Hat OpenShift 클라우드 서비스는 **AWS**, **Google 클라우드**, **IBM 클라우드**, **Microsoft Azure**에서 사용할 수 있으므로 조직의 요구 사항에 가장 적합한 옵션을 선택할 수 있습니다. 각 서비스는 엄격한 서비스 수준 계약(SLA)을 통해 필요한 모든 서비스와 간편한 셀프 서비스 옵션, 전문가의 연중무휴 24시간 지원을 제공하는 완벽한 풀스택 환경을 지원합니다.

Red Hat OpenShift 클라우드 서비스로 더 많은 성과 달성 기술 개요를 읽고 자세히 알아보세요.

Red Hat OpenShift Platform Plus를 통한 소프트웨어 팩토리 구축

Red Hat OpenShift Platform Plus는 소프트웨어 팩토리를 위한 신뢰할 수 있고 구성 가능한 적응형 기반을 제공합니다. 이러한 기술을 통해 보안 검사를 CI/CD 파이프라인에 포함하여 개발자에게 기존 워크플로우 내에서 자동화된 가드레일을 제공하고, 워크로드와 쿠버네티스 인프라의 구성 오류 및 규정 미준수 위험을 방지하며, 런타임 위협 감지와 대응을 구현할 수 있습니다.



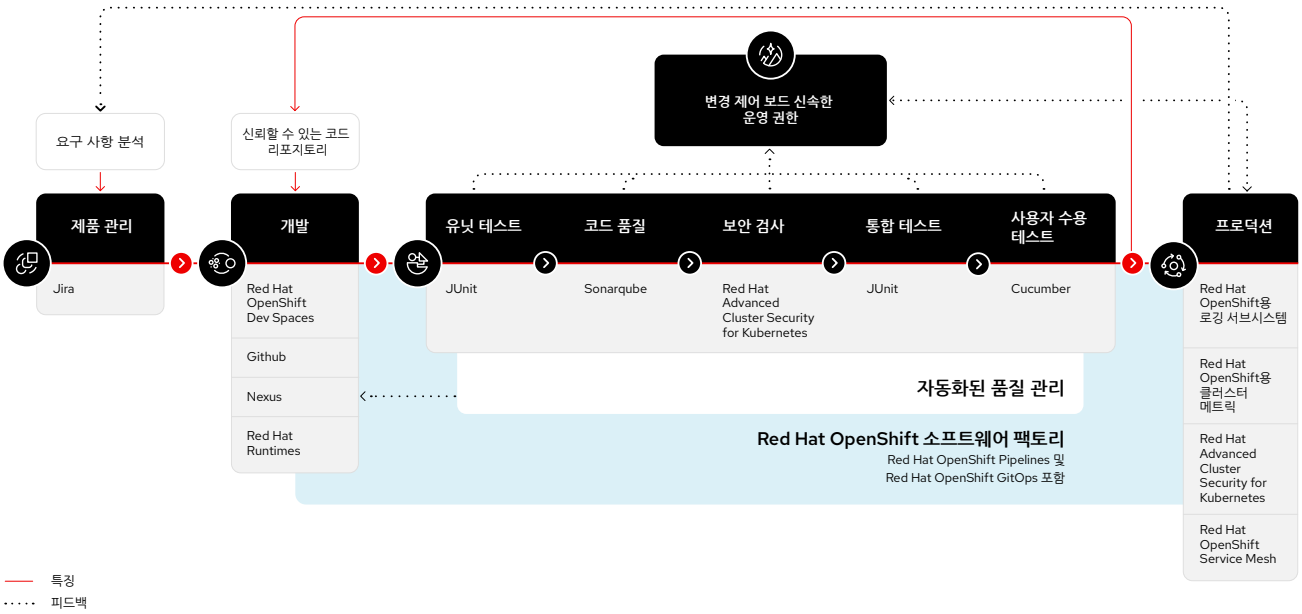
타사 툴 에코시스템으로 소프트웨어 팩토리 구성

각 활용 사례에는 소프트웨어 팩토리 내에서 서로 다른 툴을 사용해야 합니다. Red Hat OpenShift Platform Plus를 기반으로 선호하는 타사 제품과 기술을 사용하여 다음과 같이 소프트웨어 팩토리의 각 단계를 구성할 수 있습니다.

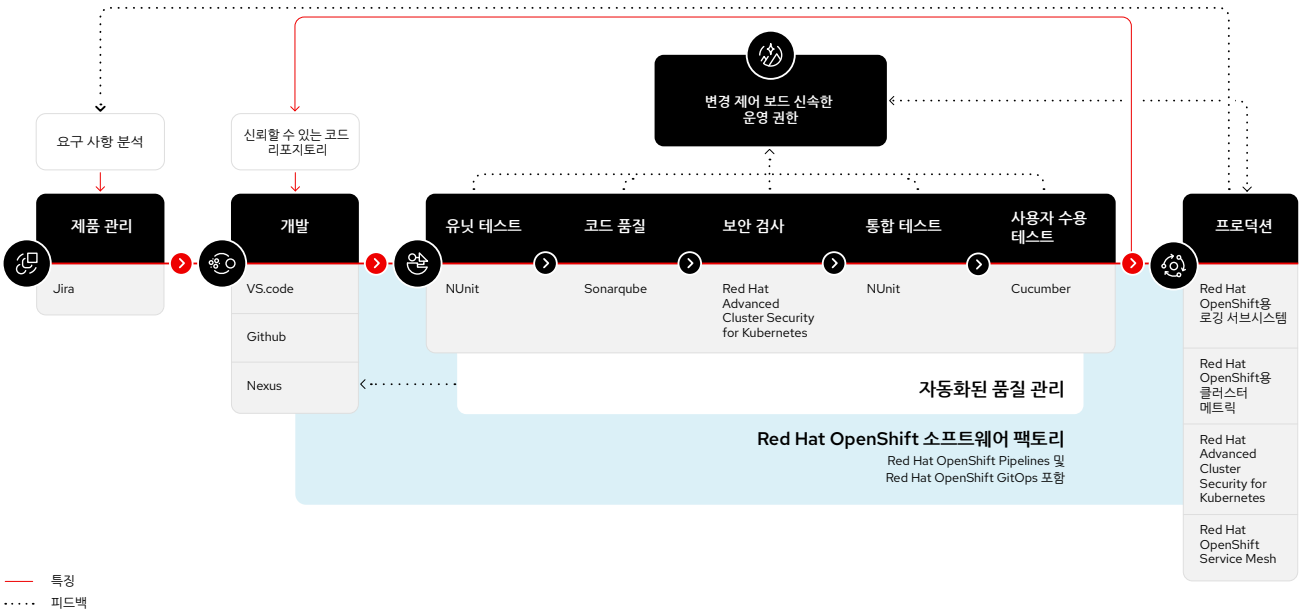
- 권한 있는 액세스 관리 툴
- 외부 인증 기관
- 외부 암호 저장소(vault) 및 키 관리 솔루션
- 컨테이너 콘텐츠 스캐너 및 취약점 관리 툴
- 컨테이너 런타임 분석 툴
- 보안 정보 및 이벤트 관리(SIEM) 시스템
- 소스 제어 관리 툴
- 아티팩트 리포지토리
- 소프트웨어 테스트 툴

예를 들어, Spring Boot 애플리케이션의 클라우드 네이티브 개발을 위한 소프트웨어 팩토리는 .Net Core 애플리케이션을 위한 소프트웨어 팩토리와는 다른 런타임, 빌드, 테스트 툴을 사용합니다. 이와 같은 두 가지 소프트웨어 팩토리에 대해 가능한 구성은 아래에 나와 있으며, 이를 통해 Red Hat 소프트웨어 팩토리 기반의 유연성을 확인할 수 있습니다.

마이크로서비스 기반 Spring Boot 애플리케이션의 클라우드 네이티브 개발을 위한 소프트웨어 팩토리



마이크로서비스 기반 .Net Core 애플리케이션의 클라우드 네이티브 개발을 위한 소프트웨어 팩토리



실제 고객 사례

Snam은 세계적인 규모의 천연 가스 네트워크로, Red Hat OpenShift, Red Hat Quay, Microsoft Azure Red Hat OpenShift를 포함하는 Red Hat 기술과 서비스를 도입하여 조직의 디지털 트랜스포메이션을 촉진하고 있습니다. 현재 Snam은 단 30분 만에 자동화된 방식으로 애플리케이션을 배포할 수 있어 신규 소프트웨어 제품을 제공하는 데 걸리는 시간이 10배 이상 단축됩니다. 또한 퍼블릭 또는 프라이빗 클라우드 전반에서 워크로드와 애플리케이션을 확장하여 향후 비즈니스 요구 사항을 충족하고 클라우드 종속성(Lock-In)과 관련한 잠재적인 리스크를 줄일 수 있습니다.

> Red Hat OpenShift는 트랜스포메이션 프로젝트의 핵심입니다. 효율적이고 높은 성능을 발휘하며 신뢰할 수 있는 IT 플랫폼을 구축하여 복잡한 시스템과 애플리케이션의 관리를 간소화해주었습니다.

Roberto Calandrini

Snam 디지털 및 AI 서비스 부문 아키텍처 책임자

자세히 알아보기

VodafoneZiggo는 네덜란드의 소비자와 기업을 위한 선도적인 커뮤니케이션 및 엔터테인먼트 서비스 제공업체로, Red Hat OpenShift 기반의 하이브리드 클라우드 플랫폼을 배포하여 조직의 애플리케이션 인프라를 통합했습니다. 또한 Red Hat Consulting과 협력하여 DevSecOps를 수용하고 더욱 개방적이고 협업적인 문화로 변화하기 위한 지침을 받았습니다. 현재 VodafoneZiggo는 비즈니스 요구 사항과 시장 요구가 계속해서 진화함에 따라 여러 클라우드 전반에서 그리고 엣지까지 더욱 빠르고 효율적으로 확장할 수 있게 되었습니다.

> Red Hat OpenShift는 생산성을 향상하고 지속적인 혁신을 가져다줄 클라우드 네이티브 애플리케이션과 서비스를 위한 일관된 레이어라고 생각합니다.

André Beijen

VodafoneZiggo 모바일 네트워크 부문 책임자

자세히 알아보기

DevSecOps로 시작하세요

클라우드 네이티브 환경에서 속도, 확장성, 보안은 매우 중요합니다.

Red Hat OpenShift Platform Plus를 기반으로 하는 소프트웨어 팩토리를 통해 개발을 가속화하고, 운영을 간소화하며, 비즈니스를 보호하는 성공적인 DevSecOps 사례를 구축할 수 있습니다.

무료로 Red Hat OpenShift 체험하기:

cloud.redhat.com/try

Red Hat OpenShift Platform Plus에 대해 자세히 알아보기:

red.ht/openshift-platform-plus

F31657_0522_KVM

Red Hat