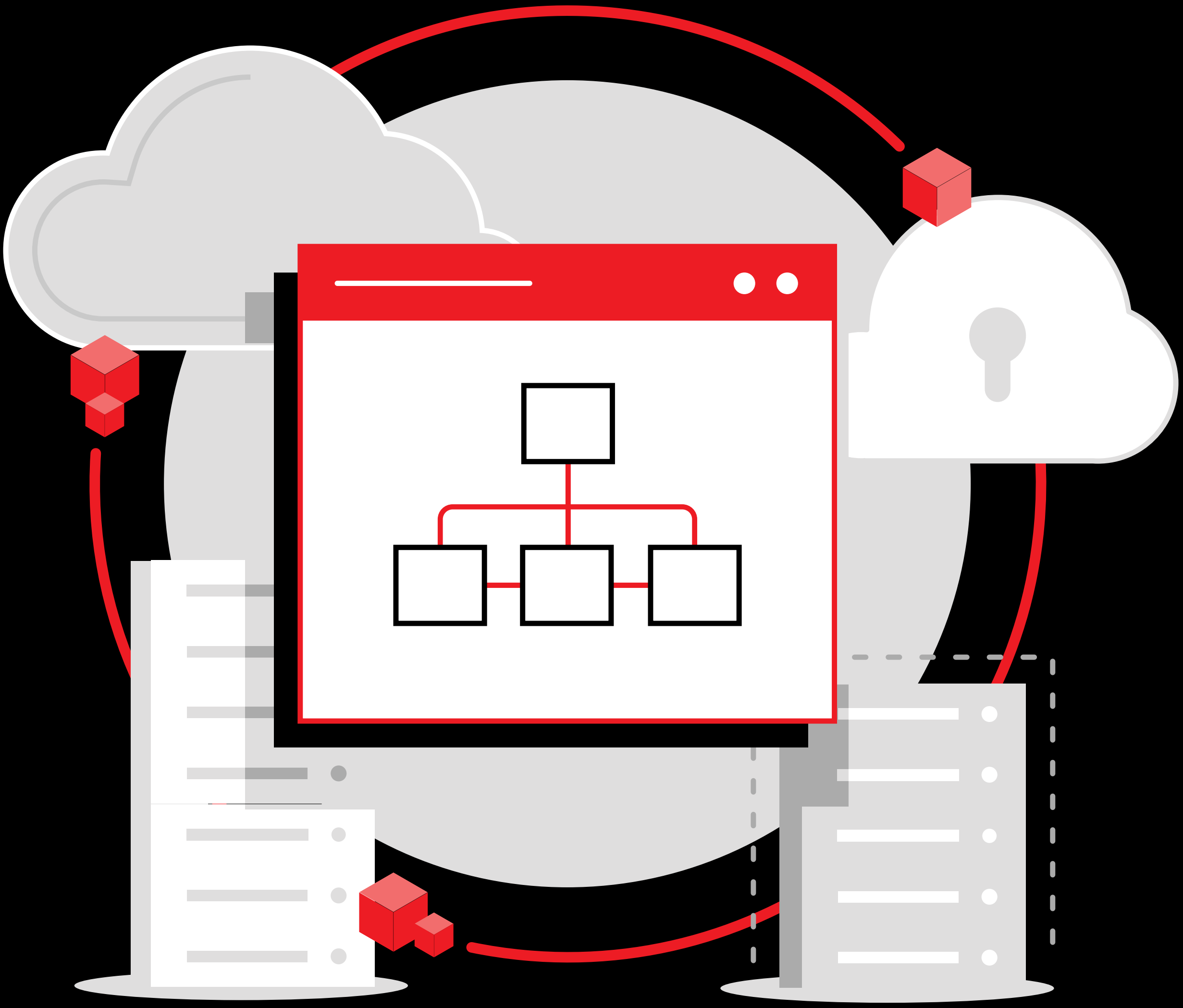
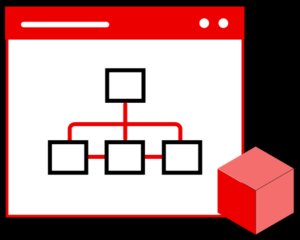
Red Hat
OpenShift
Platform Plus



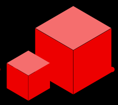
Build a software factory to support DevSecOps

An opinionated guide for starting your DevSecOps journey

Contents



Protect your business with DevSecOps



An increasing number of organizations are adopting **cloud-native**, **container**, and **microservices** technologies to innovate and **digitally transform**. As part of this transformation, many organizations use Kubernetes for container orchestration in support of cloud-native operations. Because **Kubernetes clusters** can span hosts across on-site and cloud environments, Kubernetes is an ideal platform for hosting cloud-native applications that require rapid scaling and resilient operations.

Even so, all of this introduces new challenges, particularly around security and manageability at scale. In fact, 50% of senior IT leaders at enterprises cite cybersecurity as a top-three priority for technology initiatives.[1]

Adopting DevSecOps approaches and practices can help you build security into your applications, processes, and platform to better protect your business.

This e-book discusses considerations and provides guidance for building a successful DevSecOps practice within your organization with the support of Red Hat® OpenShift® and other Red Hat technologies.

What are cloud-native applications?

A cloud-native application is a collection of small, independent, and loosely coupled services.

What are DevOps and DevSecOps?

DevOps is an approach to culture, automation, and platform design that focuses on increasing business value and responsiveness through rapid, automated, high-quality service delivery. DevSecOps extends the collaborative culture of DevOps to incorporate security throughout your application life cycles. It encompasses people, processes, and technology to make security more pervasive in distributed environments.

Through DevSecOps, security becomes a shared and enforced responsibility across teams, rather than a set of tasks owned by one team and applied at the end of the development and deployment process. Security, development, and operations teams work together, sharing information, feedback, lessons learned, and insights. This approach allows security to be integrated from the start of application development and infrastructure deployment, increasing protection and reducing risks.

88%

of surveyed organizations use Kubernetes as their container orchestrator, with 74% using it in production.[2]

74%

of surveyed organizations have a DevSecOps iniative.[2]

[1] *Flexera. "2021 Flexera State of Tech Spend Report," January 2021.*

[2] *Red Hat, "State of Kubernetes security report," 2021.*

Goals of DevSecOps

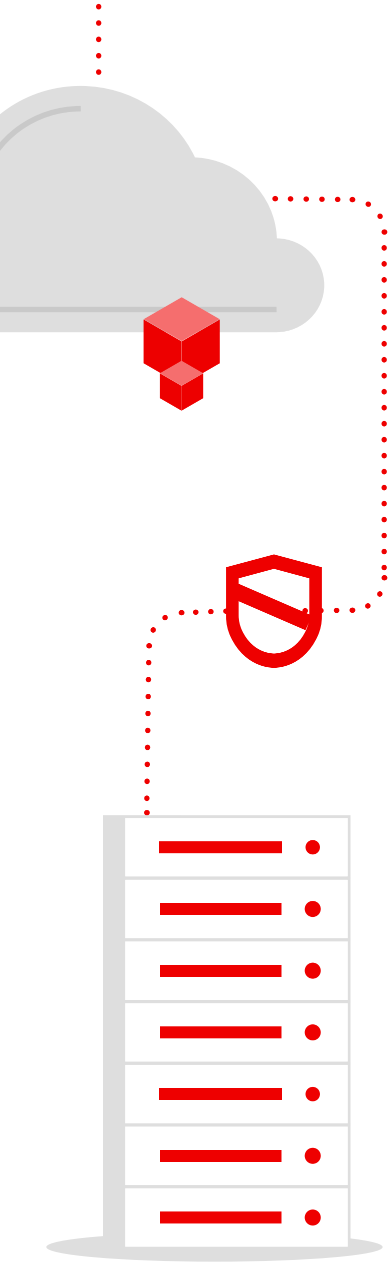
The goal of DevSecOps is to rapidly deliver and deploy high-quality, security-focused applications, services, and features at scale.

Scale

Speed

Security

Stability



Challenges for DevSecOps implementation

Manual processes

Development, test, and security tasks can be time-consuming, tedious, error-prone, and difficult to enforce when frequent human intervention is required.

Limited collaboration between teams

Development, security, and operations teams often work only within their own domain, resulting in fragmented processes, manual handoffs, and limited knowledge and understanding of the challenges and needs of other teams.

Late application of security processes

Traditional application development and launch approaches apply security practices and checks only at the end of the process, just before deploying into production.

Application environment complexity

It can be challenging to understand the connections and security implications of all of the different components – like containers, microservices, and cloud services – that make up complicated, large-scale application development, test, and production environments.

External dependencies

Cloud-native application development nearly always relies on some number of external dependencies – including sections of open source code, libraries, and services – that must also be secured.

Evolving security landscape

Security threats and regulations – including business, technical, and geographical requirements – continue to change at a rapid pace, making it difficult to stay up to date and in compliance.

People, process, and technology are crucial

DevSecOps is not a team, or a single process—it's an enterprise-wide capability that requires change and alignment in three areas: people, process, and technology.

People

People are at the core of any enterprise-wide initiative, and DevSecOps is no different. In order to adopt DevSecOps across your organization, all teams—including development, security, and operations—must be on board, participate, and trust each other.

Process

Processes move projects from start to finish. Clear processes for creating, deploying, managing, and adapting applications and infrastructure—and incorporating security throughout their life cycles—are essential for broad DevSecOps adoption.

Technology

Your application platform provides the capabilities for building, deploying, and running applications and infrastructure. A unified platform that supports development, security, and operations teams can give you a foundation for building and adapting your DevSecOps practice.

Prepare your organization for DevSecOps success

No organization can build a complete DevSecOps practice overnight. DevSecOps adoption is an iterative learning journey, not an all-or-nothing proposition. You need a logical, sustainable strategy to guide your progress and help you learn over time.

Encourage cross-team collaboration.

Use incentives and design processes to promote collaboration across your organization. Coordination allows teams to create complete DevSecOps workflows that deliver more value. Working with others also helps to cultivate shared ownership and accountability for development, security, and operations.

Document your current state.

Document your existing development, change management, and governance processes in detail using dynamic frameworks like **GitOps**. Understanding where you are and which challenges you have will help you plan your path forward. As you adapt your processes, be sure to document the new process as well as why changes were made.

Assess your processes.

Identify and adapt processes that do not support your DevSecOps goals. This includes ineffective or disparate continuous integration/continuous deployment (CI/CD) setups and infrastructure, overly centralized processes, and processes that rely on frequent manual intervention.

Define and measure success.

Determine what DevSecOps success looks like for your organization and identify measurable metrics or key performance indicators (KPIs) for tracking progress. Metrics could be application build and deployment time, change release and defect rates, problem resolution time, or application availability.

Share knowledge and best practices.

Create a core team of stakeholders – often called a community of practice (CoP) or center of excellence (CoE) – that share DevSecOps best practices, experiences, and accomplishments across your organization. This team should also help other teams that are ready to adopt DevSecOps and get started.

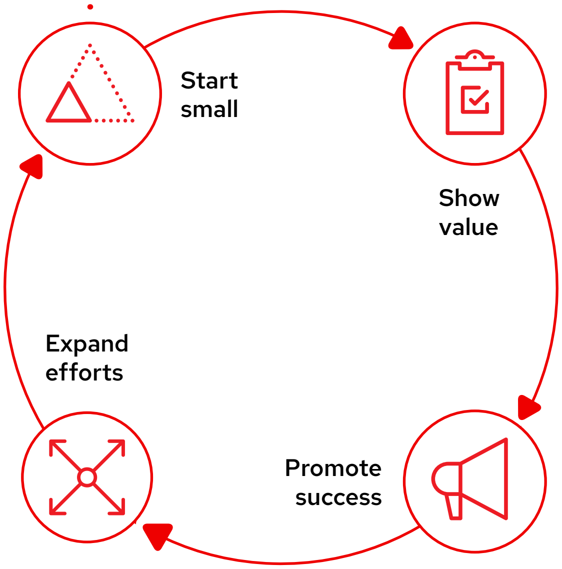
Commit across your organization.

Ensure that everyone in your organization is committed to adopting DevSecOps. Help each team understand the reasons for each change and emphasize the positive impact on their roles. Executive sponsorship and metric-based incentives will help teams progress on their journey.

Start your DevSecOps practice

Once you have your DevSecOps strategy defined, it's time to get started. Not every development team will be ready to adopt DevSecOps immediately. Start with teams that have already shown measurable success adopting new processes and platforms. Members of these teams are often good candidates for your core stakeholder team as well.

Start small, show value, expand conservatively, and repeat. Work to accomplish incremental successes over short periods of time. Monitor progress using your metrics and learn from projects or processes that are less successful. For each win, promote the value of DevSecOps and share the team's experience across your organization. This establishes a base for others to build upon each team's experiences and deliver even more value.



Take a factory approach to software delivery

Modern software delivery relies on speed, consistency, and quality. A software factory approach helps you enable, accelerate, and enforce the behavioral changes and behaviors needed to adopt a DevSecOps culture within your organization. This approach allows you to rapidly develop and deploy high-quality applications using a **trusted software supply chain** and a consistent set of agile processes like test driven development.

Benefits of a software factory

A software factory approach delivers measurable benefits:

Low lead time for changes

High deployment frequency

Low time to restore failed services

Low change failure rate

Quantified software delivery performance metrics[3]

Software delivery performance metric	With a software factory	Without a software factory
Lead time for changes	<1 hour	1-6 months
Deployment frequency	On demand (>1 per day)	Once every 1-6 months
Time to restore services	<1 hour	1 day to 1 week
Change failure rate	0%-15%	16%-30%

[3] *Google Cloud. "**Accelerate State of DevOps 2021**," September 2021.*

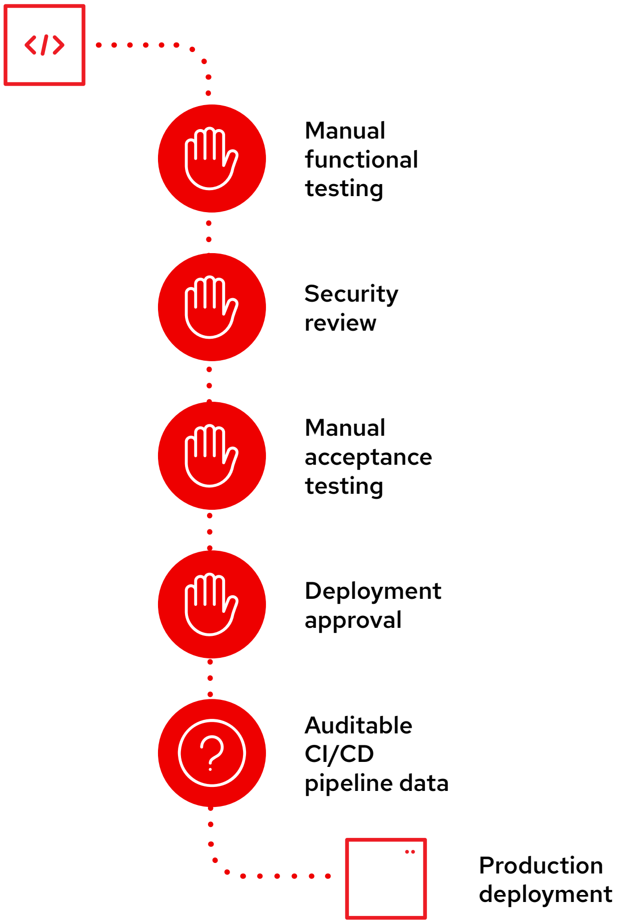
What does a software factory look like?

A software factory moves you from inconsistent, manual processes to consistent, automated operations.

Without a software factory

Manual processes and sign-offs result in slow development and deployment, unclear expectations, and inconsistent security enforcement. Because even small changes can take days or weeks to implement, teams often try to make a large number of changes in a single deployment. This increases the risk of failed changes and security issues.

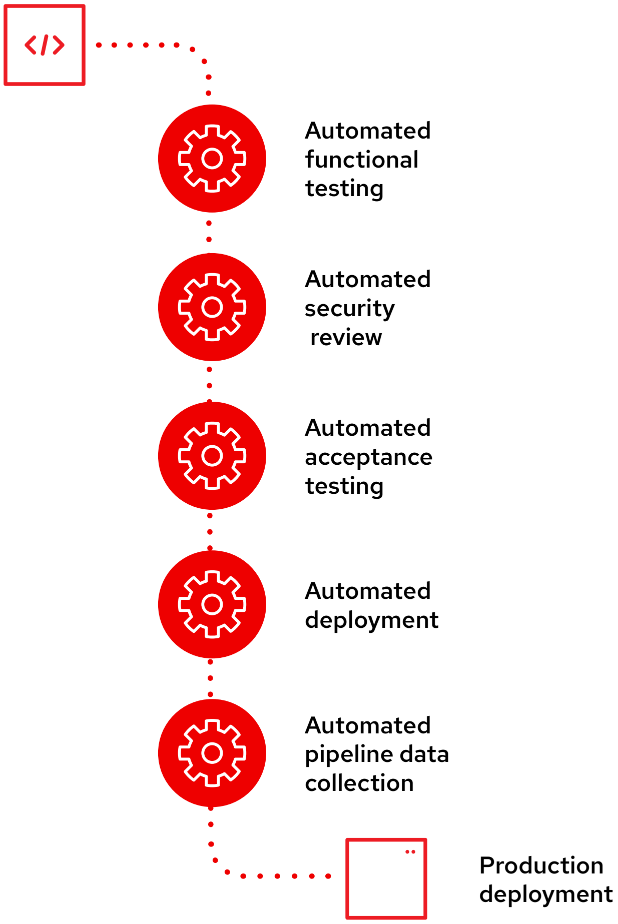
Trust between teams is often tenuous because there is a lack of transparency throughout the process. Security and compliance measures are manually applied late in the process, so issues may not be identified during development. As a result, applications may be returned to developers to fix unexpected security and compliance issues. These surprises frequently cause frustration and mistrust at an already stressful phase.


Manual functional testing
Security review
Manual acceptance testing
Deployment approval
Auditable CI/CD pipeline data
Production deployment

With a software factory

Defined, automated processes speed development and deployment, consistently enforce security, and set clear expectations for all teams involved. Because small changes can be rolled out in minutes, teams can quickly deploy many small changes on a daily basis, resulting in less risk overall.

Transparency and visibility are key features throughout software factories, making it easier to build trust between development, operations, and security teams. Security and compliance measures are automatically applied during development, so issues can be found and fixed earlier in the process. Documented processes and policies help teams understand expectations throughout the process and prevent surprises when it is time to deploy applications to production.


Automated functional testing
Automated security review
Automated acceptance testing
Automated deployment
Automated pipeline data collection
Production deployment

Build your own software factory

Automation is at the core of the software factory approach. It is critical for operating cloud-native environments and adopting DevSecOps practices. Automation helps you scale your development, delivery, deployment, and infrastructure operations in a controlled manner. You can also dynamically provision and retire resources, environments, and applications. As a result, your organization can respond faster to change.

Consider automating all aspects of your DevSecOps workflow, including your development, test, code quality control, compliance validation, vulnerability detection, and remediation processes. Use CI/CD pipelines to automate both application development and improvement as well as infrastructure deployment and management. Define and document security and risk policies and automate compliance checking and remediation against those policies throughout your software life cycles.

Declarative, intent-driven automation will help you scale and adapt more quickly and easily.

Declarative automation allows you to define a desired application or infrastructure configuration, rather than a set of instructions for setting up resources. You simply describe the end goal, rather than the means of getting there. Your application platform then provisions and configures the resources needed to reach the desired state. It also self-remediates to ensure that resources stay configured correctly over time. Finally, this approach prepares you for **GitOps** – a set of practices for managing infrastructure and application configurations using the Git version control system.

Deciding what to automate and when

Like DevSecOps as a whole, deploying automation is also a journey and requires planning. Follow these steps to get started with automation:

1. Document your process in detail.
2. At each manual step in your process, record what is being decided and how that decision is made. Decision-making may involve reading certain materials, considering specific factors, consulting with various experts, or other actions.
3. Identify all of the manual steps that can be automated easily and assess the level of change that should be automated. For example, you might automate small changes, but require approval from certain teams for larger changes.
4. For manual steps that can't be easily automated, assess what would be needed to automate them and create a plan for implementing automation.

Start automating immediately – don't wait until you've identified all possible areas of automation. Iteratively automating processes is, in itself, a DevOps process. As you automate, adapt, and refine your processes, you'll gain valuable skills and experience to support your overall DevSecOps practice.

Focus on interesting work

Automation is not meant to replace people – the focus is productivity, consistency, and efficiency. This is the paradox of automation – when you automate, human involvement becomes both more important and less frequent.

Some may see automation as a tool that eliminates jobs, but the reality is that it allows more experienced IT staff to focus on bigger problems and their solutions, rather than mundane, day-to-day, repeated tasks.

Learn how to automate across your enterprise

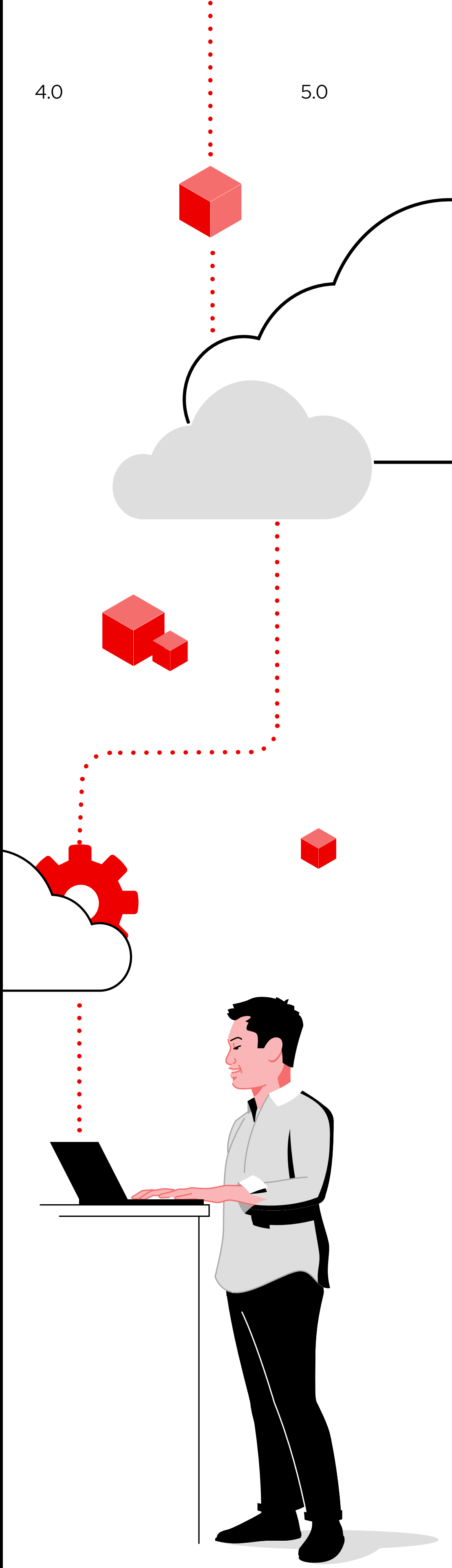
Automation can bring your people, processes, and technologies together to increase business agility, innovation, and value.

Read **The automated enterprise e-book** to learn how you can adopt automation across your organization.

Tools for your software factory

Tools are an important part of your software factory. We recommend using – and automating – these categories of tools within your software factory. Examples are given for each tool type, but you can also use others.

Tool category	Examples
Project management	▸ Confluence with Jira ▸ Trello
Source code management (SCM)	▸ Github ▸ Gitlab
Integrated development environments (IDEs)	▸ VS.code ▸ **Red Hat OpenShift Dev Spaces**
Artifact repositories	▸ Nexus ▸ Artifactory
CI/CD	▸ **Red Hat OpenShift Pipelines** ▸ **Jenkins**
Runtimes	▸ **Red Hat Runtimes** ▸ Golang
Build	▸ Maven ▸ Dotnet build
Unit testing	▸ JUnit ▸ NUnit
Source code analysis	▸ Sonarqube ▸ Fortify
Static application security testing (SAST)	▸ CheckMarx ▸ **Red Hat Advanced Cluster Security for Kubernetes**
User acceptance testing	▸ Cucumber ▸ Cyprus
Dynamic application security testing (DAST)	▸ Veracode ▸ Synopsys
Telemetry, metrics, and logging	▸ **Prometheus** ▸ **Grafana** ▸ **Elasticsearch, Fluentd, and Kibana (EFK)** ▸ Splunk
Service mesh	▸ Linkerd ▸ **Red Hat OpenShift Service Mesh**



Build, deploy, run

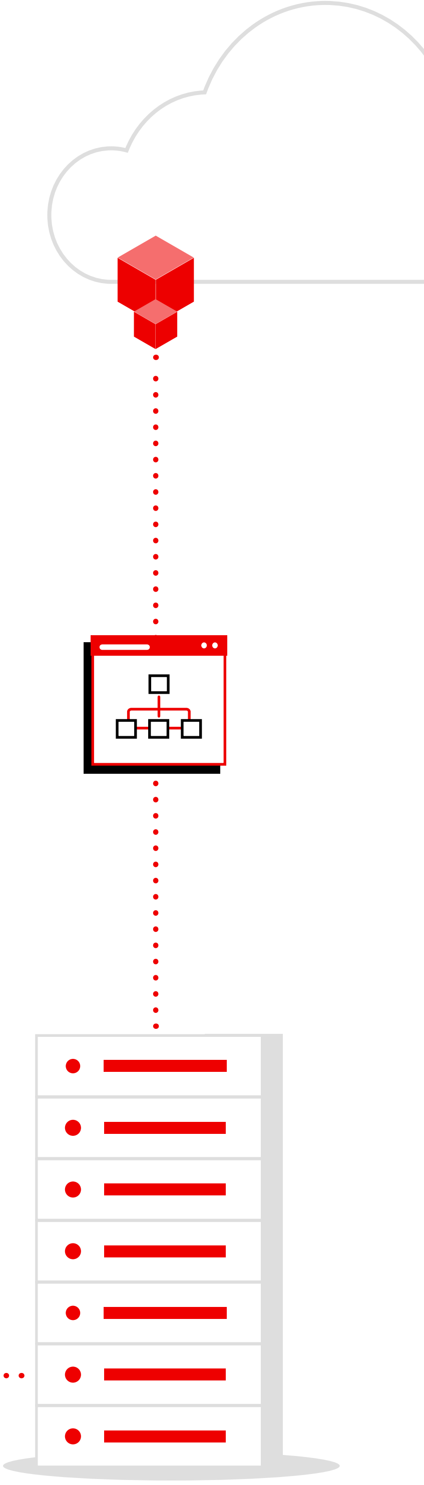
Platform architects or DevOps engineers often configure software factories on behalf of developers. When constructing your software factory, consider security best practices in these three areas: build, deploy, and run.

Build

Control application security and compliance.

Building security into your applications is critical for cloud-native deployments.

- Use trusted sources for external container and application content, including runtimes.
- Adopt a trusted, private container registry to manage images.
- Automate your development and deployment pipelines.
- Implement non-functional requirements in code using agile practices like TDD.
- Integrate security into your application pipelines with code quality, image vulnerability, and Kubernetes deployment analysis.
- Automate application deployment and placement.

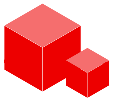


Deploy

Protect your platform.

Effective security requires safeguarding your Kubernetes platform and automating deployment policies.

- Reduce your attack surface by using an operating system optimized for containers.
- Automate configuration management and policy enforcement across clusters.
- Implement least-privilege access with fine-grained role based access controls (RBAC).
- Encrypt platform and application data in transit and at rest.
- Use automated compliance, risk assessment, and remediation solutions.
- Reduce deployment risk with Kubernetes pod admission control policies.



Run

Secure your container runtimes.

Maintain application security at runtime.

- Isolate running applications with Security-Enhanced Linux® (SELinux), Security Context Constraints (SCC), Kubernetes namespaces, RBAC, and network policies.
- Use quotas to prevent resource conflicts and related performance issues.
- Manage application access and protect application data with single sign-on user management, ingress and egress security management, encrypted pod-to-pod traffic, and application programming interface (API) management.
- Audit and monitor platform and application activity.
- Automate threat detection and response against pods with anomalous behavior, privilege escalation events, and risky processes like cryptomining.
- Use admission controllers to prevent deployment of containers that do not comply with security policies.
- Build zero-trust networks using service meshes and network policies.

Security tip

Read **A layered approach to container and Kubernetes security** to learn more about protecting containerized applications that are managed with Kubernetes.

Build	Deploy	Run
Application life cycle	**Fleet configuration management**	**Fleet observability and alerts**
Vulnerability analysis	Policy admission controller	Runtime behavioral analysis
Application configuration analysis	Compliance assessment	Network policy recommendations
APIs for CI/CD integration	Risk profiling	Threat detection and response
Trusted content	Kubernetes platform life cycle	Container isolation
Container registry	Identity and access management	Network isolation
Build management	Platform data	Application access and data
CI/CD pipelines	Deployment policies	Observability

DevSecOps

Implement DevSecOps with the experts

Red Hat brings together a certified partner ecosystem, extensive expertise, and innovative platforms for building, securing, and deploying applications across hybrid cloud environments. We have years of experience in supporting enterprise organizations and helping them overcome their technological and business challenges using industry best practices and open source technologies.

With a trusted content supply chain, support from a dedicated security team, and key security feature backports, Red Hat platforms provide an ideal foundation for DevSecOps solutions. We also offer **training and certification courses**, **interactive labs**, **consulting engagements**, and **managed offerings** to help you build a successful DevSecOps practice faster.

Red Hat meets you wherever you are in your DevSecOps journey.

With our proven open source platforms and expert services, you can deploy what you need today, adapt to future change, and learn the methods and approaches needed for efficient, effective DevSecOps adoption.

Learn more about choosing Red Hat for DevSecOps.

Get the most from your DevSecOps investment

Red Hat Services can provide you with the resources you need to begin, accelerate, and expand your DevSecOps practice.

- **Red Hat Open Innovation Labs**
  A residency-style consulting engagement where customers and Red Hatters work together as a team in order to learn new ways of working – like DevSecOps – while delivering business outcomes
- **Red Hat Services Solution: DevSecOps**
  A service engagement that helps you implement a software factory using a modular approach
- **Red Hat Services Journey: Container Adoption**
  A consulting service that addresses container adoption in key workstreams.
- **Red Hat Services Journey: Automation Adoption**
  A consulting service that provides a framework for managing your organization-wide automation adoption journey

Deploy a platform for DevSecOps success

Red Hat OpenShift Platform Plus provides a technological foundation and opinionated framework for DevSecOps. It is an innovative application platform that operates and scales consistently across on-site and cloud infrastructure. Red Hat OpenShift Platform Plus combines a leading enterprise Kubernetes platform with consistent ways to build, deploy, run, protect, and manage applications across your environment. Multicluster management tools provide complete visibility into and control of your Kubernetes clusters. Kubernetes-native security and DevSecOps capabilities protect your software supply chain, infrastructure, and workloads. A scalable, globally-distributed registry and cluster data management safeguard your environment and information.

Open integration interfaces and Red Hat's certified partner ecosystem let you use both existing and new development, test, operations, and security tools with Red Hat OpenShift Platform Plus. Many vendors offer certified Red Hat OpenShift operators or certified software containers to simplify installation and management of their software on Red Hat platforms. You can also purchase and deploy many software products directly from Red Hat Marketplace. Finally, Red Hat works with key cloud provider partners to deliver fully managed Red Hat OpenShift cloud services that streamline deployment and operations while saving costs over in-house construction.

Red Hat OpenShift Platform Plus components

Red Hat
OpenShift

Red Hat OpenShift is an enterprise-ready Kubernetes application platform with full-stack automated operations to manage hybrid cloud and edge deployments. It includes developer-focused capabilities to boost productivity and speed.

Red Hat
Advanced Cluster
Management
for Kubernetes

Red Hat Advanced Cluster Management for Kubernetes is a console that delivers visibility into your entire Kubernetes domain with built-in governance and application life-cycle management capabilities.

Red Hat
Advanced Cluster
Security
for Kubernetes

Red Hat Advanced Cluster Security for Kubernetes is a solution that provides Kubernetes-native security features to enhance infrastructure and workload protection and visibility throughout your entire application life cycle.

Red Hat
Quay

Red Hat Quay is an open source container image registry that provides storage and allows you to build, distribute, and deploy containers across datacenter and cloud environments.

Red Hat
OpenShift
Data Foundation

Red Hat OpenShift Data Foundation is a scalable data and storage services layer that provides data efficiency, resilience, and security for Red Hat OpenShift environments.

Red Hat OpenShift Platform Plus supports you at all points in your DevSecOps journey. It meets you where you are today and gives you a foundation to move forward at your own pace.

Built-in security capabilities

Monitor running workloads for security issues and threats with system-level data collection and analysis and more than 60 built-in security policies that can be applied and enforced throughout your entire application life cycle.

Consistent operations

Apply consistent operational policies for security, configuration, compliance, and governance to Red Hat OpenShift clusters across on-site datacenter and cloud infrastructures.

Developer tools

Create, run, and deploy applications faster with an included library of supported build tools, languages, pipelines, and frameworks. The operator framework delivers integrations for the latest developer tools tested and verified to run with Red Hat OpenShift.

End-to-end management

Manage your Red Hat OpenShift environment consistently with a uniform interface for both administrators and developers that works across on-site, cloud, and edge environments, including those based on different Kubernetes distributions.

Support for DevSecOps

Integrate declarative security into developer tooling and workflows. Use Kubernetes-native controls to mitigate threats, enforce security policies, and minimize operational risk.

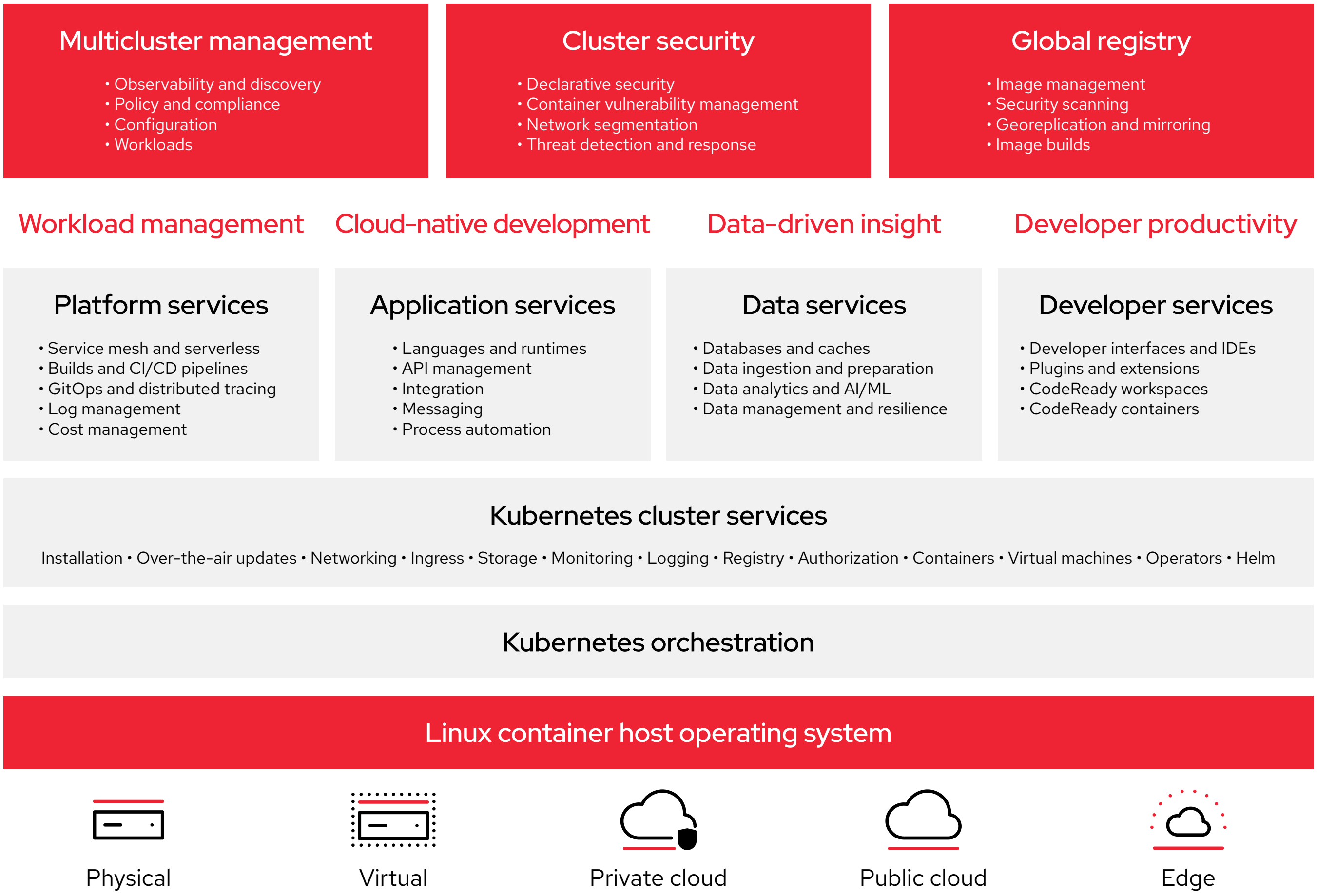
Scalable data services

Streamline data management across your clusters. With support for file, block, and object data protocols, Red Hat OpenShift Data Foundation delivers resilient persistent storage for stateful applications and cluster services.

Zero-trust networking capabilities

Implement **zero-trust networks** to provide resilient, secure, observable communications between applications and services. **Red Hat OpenShift Service Mesh** is included and integrated with Red Hat OpenShift to help you safeguard your communications more easily.

Red Hat OpenShift Platform Plus delivers the technologies and capabilities needed for effective DevSecOps adoption. Read the **Red Hat OpenShift security guide** to learn how security is addressed throughout the technology stack.



Multicluster management

- Observability and discovery
- Policy and compliance
- Configuration
- Workloads

Cluster security

- Declarative security
- Container vulnerability management
- Network segmentation
- Threat detection and response

Global registry

- Image management
- Security scanning
- Georeplication and mirroring
- Image builds

Workload management

Cloud-native development

Data-driven insight

Developer productivity

Platform services

- Service mesh and serverless
- Builds and CI/CD pipelines
- GitOps and distributed tracing
- Log management
- Cost management

Application services

- Languages and runtimes
- API management
- Integration
- Messaging
- Process automation

Data services

- Databases and caches
- Data ingestion and preparation
- Data analytics and AI/ML
- Data management and resilience

Developer services

- Developer interfaces and IDEs
- Plugins and extensions
- CodeReady workspaces
- CodeReady containers

Kubernetes cluster services

Installation • Over-the-air updates • Networking • Ingress • Storage • Monitoring • Logging • Registry • Authorization • Containers • Virtual machines • Operators • Helm

Kubernetes orchestration

Linux container host operating system

Physical

Virtual

Private cloud

Public cloud

Edge

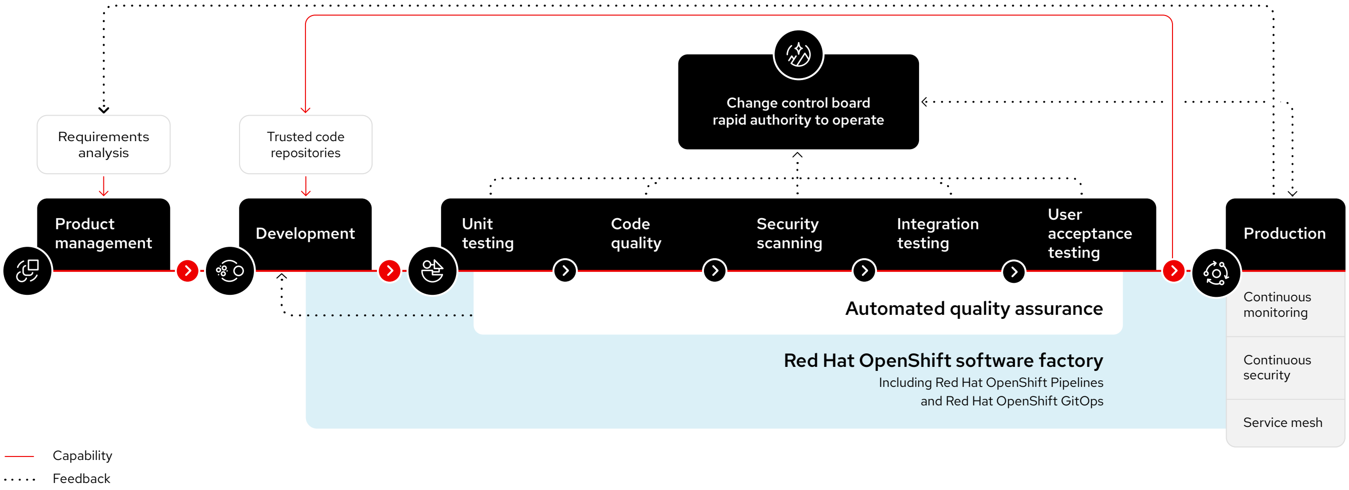
Get started faster with Red Hat OpenShift cloud services

Red Hat OpenShift cloud services are available on **AWS**, **Google Cloud**, **IBM Cloud**, and **Microsoft Azure**, so you can choose the option that best fits your organization's needs. Each service provides complete, full-stack environments with all necessary services, simple self-service options, and expert 24x7 support via stringent service-level agreements (SLAs).

Read the **Achieve more with Red Hat OpenShift cloud services brief** to learn more.

Build a foundation for your software factory with Red Hat OpenShift Platform Plus

Red Hat OpenShift Platform Plus provides a reliable, adaptable, composable foundation for your software factory. It lets you embed security checks into your CI/CD pipelines to give developers automated guardrails within existing workflows, protect workloads and Kubernetes infrastructure against misconfiguration and noncompliance, and implement runtime threat detection and response.



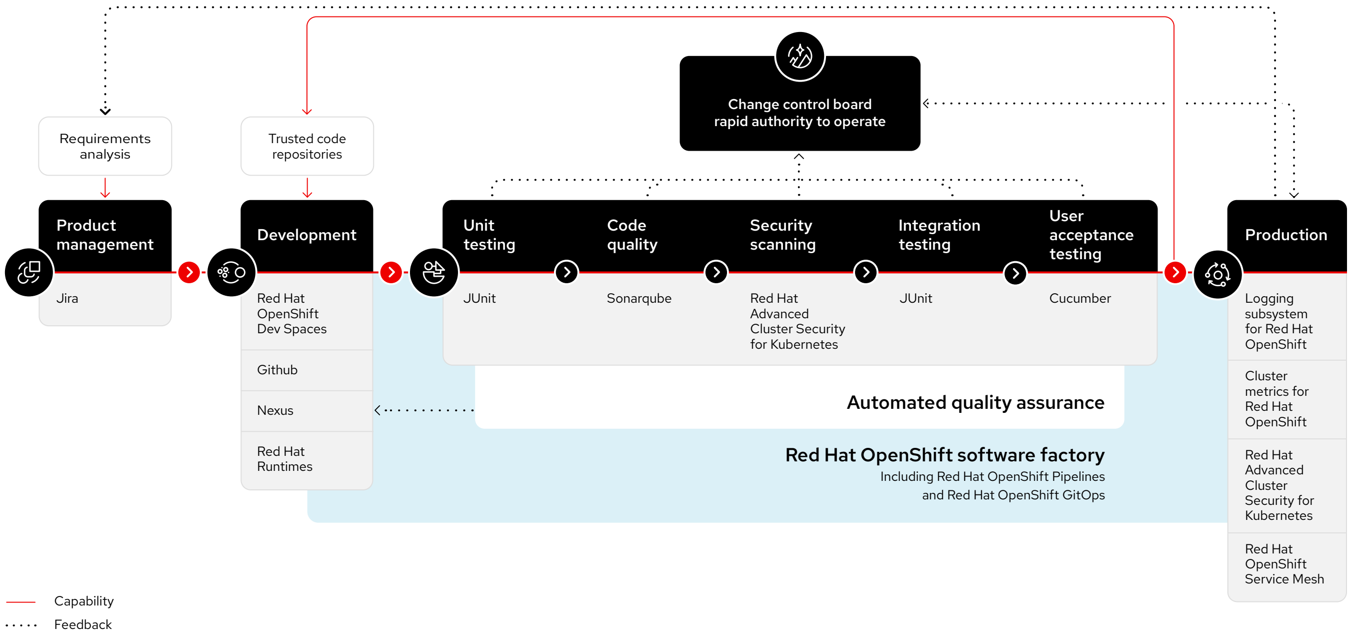
Compose complete software factories with an ecosystem of third-party tools

Each use case requires different tools within your software factory. Based on a foundation of Red Hat OpenShift Platform Plus, you can compose each stage of your software factory using your preferred third-party products and technologies, including:

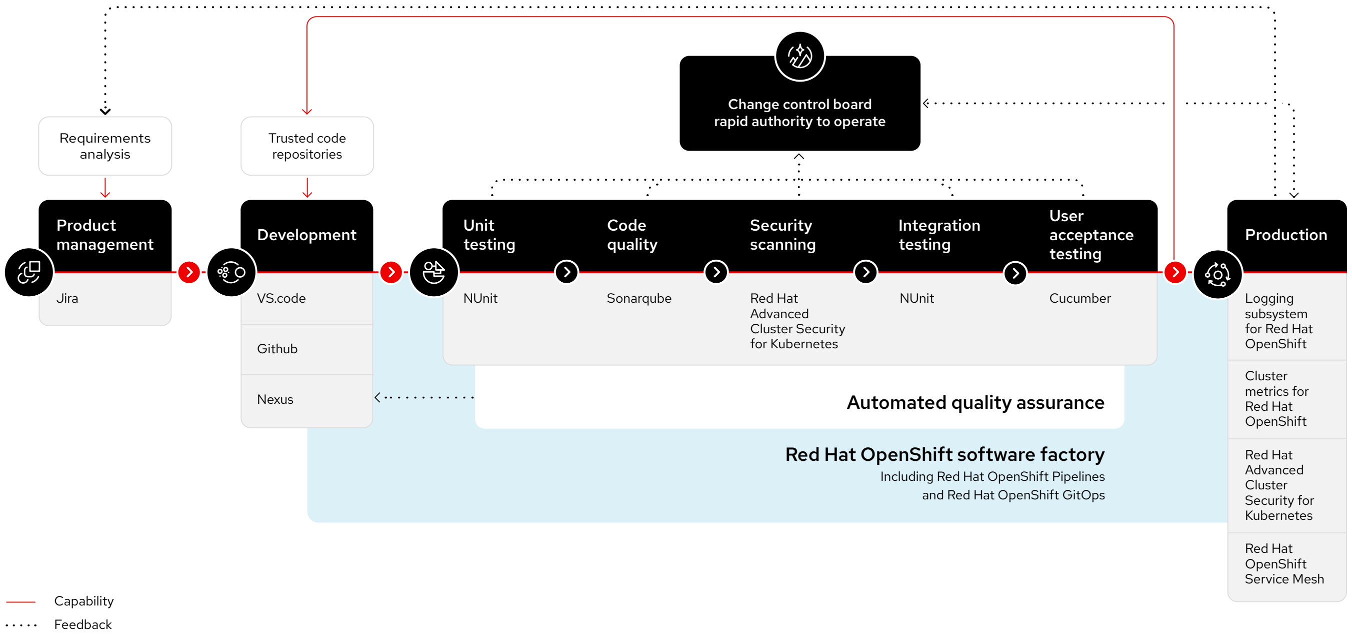
- Privileged access management tools.
- External certificate authorities.
- External vaults and key management solutions.
- Container content scanners and vulnerability management tools.
- Container runtime analysis tools.
- Security information and event management (SIEM) systems.
- Source control management tools.
- Artifact repositories.
- Software testing tools.

For example, a software factory for cloud-native development of Spring Boot applications would use different runtime, build, and testing tools than a software factory for .Net Core applications. Possible compositions for both of these software factories are shown below to illustrate the flexibility of a Red Hat software factory foundation.

Software factory for cloud-native development of microservices-based Spring Boot applications


Requirements analysis
Trusted code repositories
Change control board rapid authority to operate
Product management
Jira
Development
Red Hat OpenShift Dev Spaces
Github
Nexus
Red Hat Runtimes
Unit testing
JUnit
Code quality
Sonarqube
Security scanning
Red Hat Advanced Cluster Security for Kubernetes
Integration testing
JUnit
User acceptance testing
Cucumber
Automated quality assurance
Red Hat OpenShift software factory
Including Red Hat OpenShift Pipelines and Red Hat OpenShift GitOps
Production
Logging subsystem for Red Hat OpenShift
Cluster metrics for Red Hat OpenShift
Red Hat Advanced Cluster Security for Kubernetes
Red Hat OpenShift Service Mesh
Capability
Feedback

Software factory for cloud-native development of microservices-based .Net Core applications


Requirements analysis
Trusted code repositories
Change control board rapid authority to operate
Product management
Jira
Development
VS.code
Github
Nexus
Unit testing
NUnit
Code quality
Sonarqube
Security scanning
Red Hat Advanced Cluster Security for Kubernetes
Integration testing
NUnit
User acceptance testing
Cucumber
Automated quality assurance
Red Hat OpenShift software factory
Including Red Hat OpenShift Pipelines and Red Hat OpenShift GitOps
Production
Logging subsystem for Red Hat OpenShift
Cluster metrics for Red Hat OpenShift
Red Hat Advanced Cluster Security for Kubernetes
Red Hat OpenShift Service Mesh
Capability
Feedback

See success in action

Snam, one of the world's largest natural gas networks, adopted Red Hat technologies and services—including Red Hat OpenShift, Red Hat Quay, and Microsoft Azure Red Hat OpenShift—to help drive the organization's digital transformation. The company can now deploy applications in an automated manner in as little as 30 minutes, improving by more than 10x the time to delivery of its new software products. Snam can also scale workloads and applications across any public or private cloud in order to meet future business requirements, reducing potential risks around cloud lock-in.

> Red Hat OpenShift is a cornerstone of our transformation project; it has enabled us to create an efficient, high-performing, reliable IT platform, simplifying the management of complex systems and applications.

Roberto Calandrini

Head of architecture, Digital and AI Services, Snam

Learn more

VodafoneZiggo, one of the leading communication and entertainment service providers for consumers and businesses in the Netherlands, deployed a hybrid cloud platform based on Red Hat OpenShift to unify the organization's application infrastructure. The company also engaged Red Hat Consulting to provide guidance for embracing DevSecOps as well as moving to a more open, collaborative culture. VodafoneZiggo will now be able to scale out more rapidly and efficiently across multiple clouds and to the edge as business needs and market demands evolve.

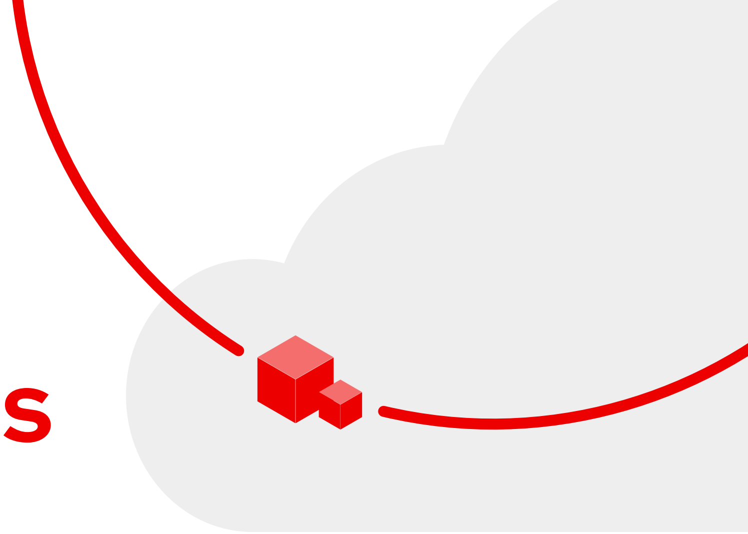
> We view Red Hat OpenShift as a consistent layer for cloud-native applications and services that will enable us to boost productivity and deliver continuous innovation.

André Beijen

Director, Mobile Network, VodafoneZiggo

Learn more

Get started with DevSecOps



Speed, scale, and security are critical in a cloud-native world.

A software factory based on Red Hat OpenShift Platform Plus can help you build a successful DevSecOps practice that accelerates development, streamlines operations, and protects your business.

Try Red Hat OpenShift for free:

cloud.redhat.com/try

Learn about Red Hat OpenShift Platform Plus:

red.ht/openshift-platform-plus