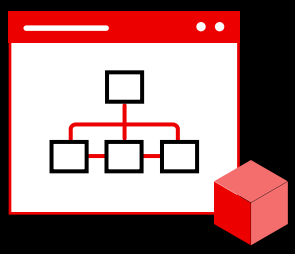
Red Hat
OpenShift
Platform Plus

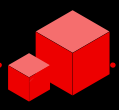
DevSecOps をサポートするソフトウェアファクトリーの構築

DevSecOps の導入を始めるための独自ガイド

目次



DevSecOps でビジネスを保護



クラウドネイティブ、**コンテナ**、**マイクロサービス**といったテクノロジーを導入してイノベーションと**デジタル・トランスフォーメーション**を実現する組織が増えています。この変革の一環として、多くの組織が Kubernetes を使用してコンテナをオーケストレーションし、クラウドネイティブな運用をサポートしています。**Kubernetes クラスタ**は、オンサイト環境とクラウド環境にまたがるホストに対応できるため、Kubernetes は、迅速なスケーリングと回復力のある運用が必要なクラウドネイティブ・アプリケーションをホストするには理想的なプラットフォームです。

とはいえ、これらはすべて、特に広範囲にわたるセキュリティと管理性に関して、新たな課題をもたらします。実際、組織のシニア IT リーダーの 50% が、テクノロジーに関する取り組みにおける上位 3 つの優先事項の中にサイバーセキュリティを挙げています。[1]

DevSecOps のアプローチを導入し、それを実践することで、アプリケーション、プロセス、プラットフォームにセキュリティを組み込み、ビジネスの保護を強化できます。

この e ブックでは、Red Hat® OpenShift® と他の Red Hat テクノロジーのサポートにより、組織内の DevSecOps の実践で成功するための考慮事項について説明し、ガイダンスを提供します。

クラウドネイティブ・アプリケーションとは

クラウドネイティブ・アプリケーションは、疎結合された小型で独立したサービスの集合です。

DevOps および DevSecOps とは

DevOps は、高品質かつ迅速で自動化されたサービス提供によりビジネス価値や対応スピードを向上することに重点を置いた、企業文化、自動化、およびプラットフォームの設計に対するアプローチです。**DevSecOps** では、DevOps の協調的な文化が拡張され、アプリケーション・ライフサイクル全体にセキュリティが組み込まれます。これには、分散型の環境でセキュリティの普及を促進する人材、プロセス、テクノロジーが含まれます。

DevSecOps により、セキュリティの適用は 1 つのチームが担当し、開発およびデプロイのプロセスの最終時点で行われる一連の作業ではなく、すべてのチームが共有し、実施する任務になります。セキュリティ、開発、運用の各チームが連携して、情報、フィードバック、学んだ教訓、知見を共有します。このアプローチにより、アプリケーション開発とインフラストラクチャのデプロイメントの開始時からセキュリティを統合できるため、保護が強化され、リスクが軽減されます。

88%

調査対象の組織のうち、Kubernetes をコンテナ・オーケストレーターとして使用している組織。プロダクションで使用している組織は 74%[2]

74%

調査対象の組織のうち、DevSecOps に取り組んでいる組織[2]

1 Flexera、「2021 Flexera State of Tech Spend Report」、2021 年 1 月。

2 Red Hat、「Kubernetes のセキュリティの現状に関するレポート」、2021 年。

DevSecOps の目標

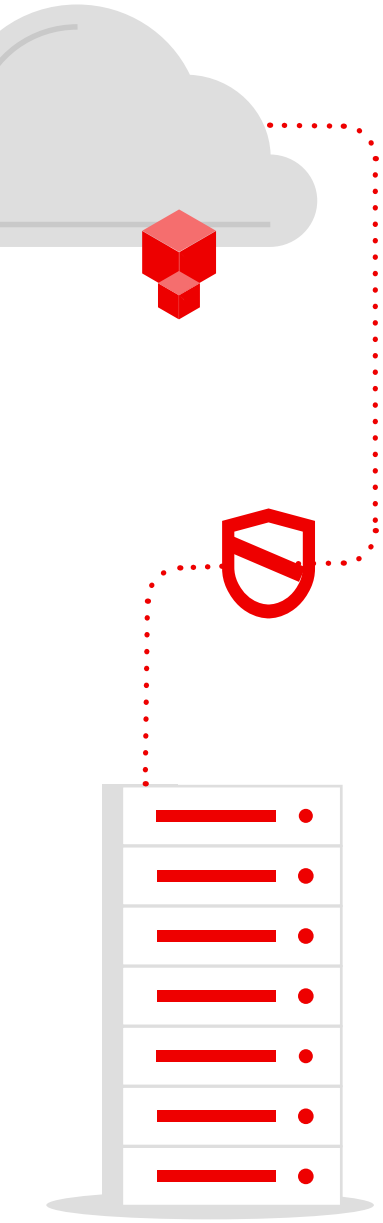
DevSecOps の目標は、高品質でセキュリティに重点を置いたアプリケーション、サービス、機能を大規模かつ迅速に提供し展開することです。

スケーリング

スピード

セキュリティ

安定性



DevSecOps 実装の課題

手作業のプロセス

開発、テスト、セキュリティのタスクは、人による作業が頻繁に必要になる場合、時間がかかり、面倒で、エラーが発生しやすく、実施が困難になることがあります。

チーム間のコラボレーションが限定的

開発、セキュリティ、運用の各チームは、自分の領域内でのみ作業することが多く、プロセスが断片化され、手作業による引き渡しが行われ、他のチームの課題とニーズに関する知識と理解が制限されます。

セキュリティプロセスの適用が遅い

従来のアプリケーション開発とリリースのアプローチでは、セキュリティプラクティスとセキュリティチェックはプロセスの最後、つまりプロダクションにデプロイする直前に適用されます。

アプリケーション環境の複雑さ

複雑で大規模なアプリケーション開発環境、テスト環境、プロダクション環境を構成するさまざまなコンポーネント (コンテナ、マイクロサービス、クラウドサービスなど) のすべての接続とセキュリティへの影響を理解することは困難です。

外部依存関係

クラウドネイティブ・アプリケーション開発は、ほとんどの場合、いくつかの外部依存関係 (オープンソースコード、ライブラリ、サービスの部分など) を利用しており、これらも保護する必要があります。

進化するセキュリティ事情

セキュリティの脅威と規制 (ビジネス上の要件、技術的要件、地理的要件など) は急速に変化し続けており、最新情報を把握してコンプライアンスを維持するのは困難になっています。

人材、プロセス、テクノロジーが重要

DevSecOps は 1 つのチームや単一のプロセスではなく、人材、プロセス、テクノロジーの 3 つの領域での変化と連携を必要とする全社的な機能です。

人材

人材は、どのような全社的な取り組みでも中心的な存在で、DevSecOps も例外ではありません。組織に DevSecOps を導入するためには、開発、セキュリティ、運用を含むすべてのチームが協力し、取り組みに参加し、お互いを信頼する必要があります。

プロセス

プロセスはプロジェクトを開始から終了まで進めるためのものです。アプリケーションとインフラストラクチャの作成、デプロイ、管理、および適応の明確なプロセスと、そのライフサイクル全体でのセキュリティの組み込みは、DevSecOps の幅広い導入に不可欠です。

テクノロジー

アプリケーション・プラットフォームは、アプリケーションとインフラストラクチャを構築、デプロイ、実行するための機能を提供します。開発、セキュリティ、運用の各チームをサポートする統合プラットフォームは、DevSecOps の実践を構築し適応させるための基盤になります。

DevSecOps の成功に向けて組織を準備

一夜にして完全な DevSecOps の実践を構築できる組織はありません。DevSecOps の導入は反復的な学習を伴うものであり、0 か 100 かの問題ではありません。組織を進歩させ、長期的な学びを支援する、論理的で持続可能な戦略が必要です。

チーム間のコラボレーションを促進する

インセンティブと設計プロセスを使用して組織でのコラボレーションを促進します。チーム間で調整することにより、全体を網羅した DevSecOps ワークフローを作成でき、より高い価値を提供できます。他チームと連携すると、開発、セキュリティ、運用に対する当事者意識や責任感を共有できるようにもなります。

現在の状態を文書化する

GitOps などの動的なフレームワークを使用して、既存の開発、変更管理、およびガバナンスのプロセスを詳細に文書化します。現在の状況と抱えている課題を理解することは、今後の道筋を計画する上で役立ちます。プロセスを調整する場合は、新しいプロセスだけでなく、変更が行われた理由を記録しておくようにしましょう。

プロセスを評価する

組織の DevSecOps の目標に対応していないプロセスを特定し、適応させます。これには、効果のない、または統合されていない継続的インテグレーション/継続的デプロイメント (CI/CD) のセットアップとインフラストラクチャ、過度に一元化されたプロセス、頻繁に手作業を必要とするプロセスなどがあります。

成功を定義して数値化する

組織がどのような結果を DevSecOps の成功と見なすかを決定し、進捗状況を追跡するための測定可能な指標または主要業績評価指標 (KPI) を明確にします。指標には、アプリケーションのビルドとデプロイにかかる時間、変更のリリースと欠陥の割合、問題解決にかかる時間、アプリケーションの可用性などがあります。

知識とベストプラクティスを共有する

中心的な関係者からなるチームを結成します。これは一般にコミュニティ・オブ・プラクティス (CoP) またはセンター・オブ・エクセレンス (CoE) などと呼ばれ、DevSecOps のベストプラクティス、経験、達成を組織内で共有します。また、このチームは、DevSecOps の導入と開始の準備ができている他のチームを支援する必要があります。

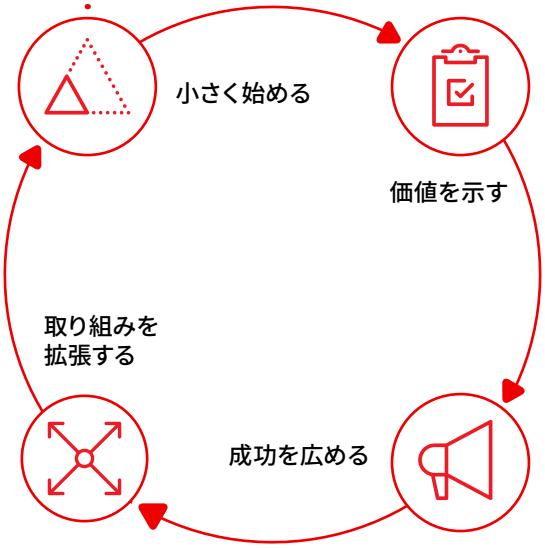
組織全体で取り組む

DevSecOps の導入には組織内の全員で取り組むようにしましょう。各チームにそれぞれ変更が必要な理由を理解してもらい、彼らの職務にプラスの影響があることを強調します。経営陣による後押しと指標に基づくインセンティブは、チームが取り組みを進めるための助けとなります。

DevSecOps の実践を開始する

DevSecOps 戦略を定義したら、DevSecOps に着手する準備が整います。すべての開発チームが即座に DevSecOps を導入できるわけではありません。すでに新しいプロセスとプラットフォームを導入し、具体的な成功を収めているチームから始めましょう。このようなチームのメンバーは、多くの場合、中心的な関係者からなるチームのメンバーとしても適しています。

小規模から始めて価値を示し、少しずつ拡張するのを繰り返します。短期間で段階的に成功を積み重ねていきます。指標を使用して進捗を監視し、あまりうまく行っていないプロジェクトやプロセスから学びます。成功を収めるごとに、組織に DevSecOps の価値を広めて、チームの経験を共有します。これにより、各チームの成功経験を踏まえて他の人がさらなる価値を実現していくための基礎が確立されます。



ソフトウェア提供にファクトリーアプローチを採用

先進的なソフトウェア提供では、スピード、一貫性、品質が重要です。ソフトウェア・ファクトリー・アプローチは、組織にDevSecOps 文化を導入するために必要な行動および行動の変更を可能にし、加速し、実施するために役立ちます。このアプローチにより、**信頼できるソフトウェア・サプライチェーン**と、テスト駆動型開発のような一貫性のある一連のアジャイルプロセスを使用して、高品質のアプリケーションを迅速に開発しデプロイできます。

ソフトウェアファクトリーのメリット

ソフトウェア・ファクトリー・アプローチは、次のような測定可能なメリットをもたらします。

変更のリードタイムが短い

デプロイメントの頻度が高い

障害が発生したサービスの復旧時間が短い

変更の失敗率が低い

ソフトウェア提供のパフォーマンスを数値化[3]

ソフトウェア提供のパフォーマンス指標	ソフトウェアファクトリーあり	ソフトウェアファクトリーなし
変更のリードタイム	1時間以下	1 - 6 カ月
デプロイメントの頻度	オンデマンド (1日につき1回以上)	1 - 6 カ月に1回
サービスの復旧時間	1時間以下	1日から1週間
変更の失敗率	0% - 15%	16% - 30%

3 Google Cloud、「**Accelerate State of DevOps 2021**」、2021年9月。

ソフトウェアファクトリーとはどのようなものか

ソフトウェアファクトリーによって、一貫性のない手作業による処理から、一貫性のある自動化された運用へと移行できます。

ソフトウェアファクトリーなし

処理やサインオフを手作業で行う場合、開発とデプロイに遅れが生じ、予測が不明確になり、セキュリティの適用に一貫性がなくなります。小さな変更でも実装に数日または数週間かかることがあるため、チームは多くの場合、一度のデプロイで多数の変更を加えようとします。これにより、変更が失敗したり、セキュリティの問題が発生したりするリスクが高まります。

プロセス全体で透明性が欠如しているため、チーム間の信頼が希薄であることはよくあります。セキュリティとコンプライアンスへの対応策がプロセスの後半に手作業で適用されるため、開発中に問題が特定されないことがあります。その結果、予期していなかったセキュリティやコンプライアンスの問題を修正するために、アプリケーションが開発者に戻される場合があります。これは開発者にとって予期せぬ驚きであり、多くの場合、すでに多くのストレスを抱えている状況ではフラストレーションと不信感を引き起こす要因になります。

ソフトウェアファクトリーあり

定義され、自動化されたプロセスにより、開発とデプロイが加速され、セキュリティが一貫して適用され、関係するすべてのチームが明確な予測を立てることができます。小さな変更は数分で展開できるため、チームは毎日多くの小さな変更を迅速にデプロイでき、全体的にリスクが軽減されます。

透明性と可視性は、ソフトウェアファクトリー全体の重要な機能であり、開発、運用、セキュリティの各チーム間の信頼関係の構築を容易にします。セキュリティとコンプライアンスへの対応策は開発時に自動的に適用されるため、プロセスの早い段階で問題を発見して修正できます。プロセスとポリシーを文書化しておくと、チームはプロセスを通じて予測されることを理解でき、アプリケーションをプロダクションにデプロイする際に予期せぬ事態が生じるのを防ぐことができます。

独自のソフトウェアファクトリーを構築

自動化は、ソフトウェア・ファクトリー・アプローチの中核をなすものです。これは、クラウドネイティブ環境を運用し、DevSecOps の実践を導入するために重要です。自動化は、制御された方法で開発、提供、デプロイ、およびインフラストラクチャの運用をスケーリングするのに役立ちます。また、リソース、環境、アプリケーションの動的なプロビジョニングと廃棄も可能になります。その結果、組織は変更への対応を迅速化することができます。

開発、テスト、コードの品質管理、コンプライアンスの検証、脆弱性の検出、修復プロセスなど、DevSecOps ワークフローのあらゆる側面の自動化を検討してください。CI/CD パイプラインを使用して、アプリケーションの開発と改善だけでなく、インフラストラクチャのデプロイと管理も自動化しましょう。セキュリティとリスクのポリシーを定義して文書化し、ソフトウェア・ライフサイクル全体でそれらのポリシーに対するコンプライアンスチェックと修復を自動化します。

宣言型でインテント駆動型の自動化によって、より迅速かつ容易に拡張し適応することができます。

宣言型の自動化により、リソースをセットアップするための一連の指示ではなく、アプリケーションまたはインフラストラクチャの望ましい設定を定義できます。最終目標に到達するための手段ではなく、ただ最終目標を記述するだけです。すると、アプリケーション・プラットフォームが、望ましい状態に到達するために必要なリソースのプロビジョニングと設定を行います。また、時間が経過してもリソースの正しい設定が維持されるように、自己修復を行います。そして、このアプローチによって **GitOps**の準備が整います。GitOps は、Git バージョン管理システムを使用してインフラストラクチャとアプリケーションの設定を管理するための一連のプラクティスです。

何をいつ自動化するかを決定する

全体としての DevSecOps と同様、自動化のデプロイも一連のプロセスであり、計画を立てる必要があります。以下の手順に従って、自動化を開始してください。

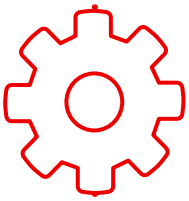
1. プロセスを詳細に文書化します。
2. プロセスにおける手作業の手順それぞれについて、何が決定され、その決定がどのように行われたかを記録します。意思決定には、特定の資料を読む、特定の要因を検討する、さまざまなエキスパートに相談するといった行動が必要な場合があります。
3. 手作業の手順のうち、簡単に自動化できるものをすべて特定し、自動化すべき変更のレベルを評価します。たとえば「小さな変更は自動化するが、大きな変更には一部のチームの承認が必要になる」などです。
4. 手作業の手順のうち、簡単に自動化できないものについては、それらを自動化するために何が必要かを評価し、自動化の実装計画を作成します。

今すぐ自動化を始めましょう。自動化できる領域をすべて特定するまで待つ必要はありません。プロセスを繰り返し自動化すること自体が、DevOps プロセスです。プロセスを自動化し、適応させ、改良することで、DevSecOps の実践全体をサポートする貴重なスキルと経験を得ることができます。

やりがいのある仕事に専念

自動化は人を置き換えることを意図したものではなく、生産性、一貫性、効率性に焦点を当てています。これが自動化のパラドックスです。自動化すると、人間が関与する頻度は少なくなりますが、関与することの重要性が増します。

自動化を、仕事を減らすツールと見なす人もいるかもしれません。しかし実際には、自動化により、経験豊富な IT スタッフが、ありふれた日常的な反復タスクではなく、より大きな問題とそのソリューションに集中できるようになります。



組織全体で自動化する方法を学ぶ

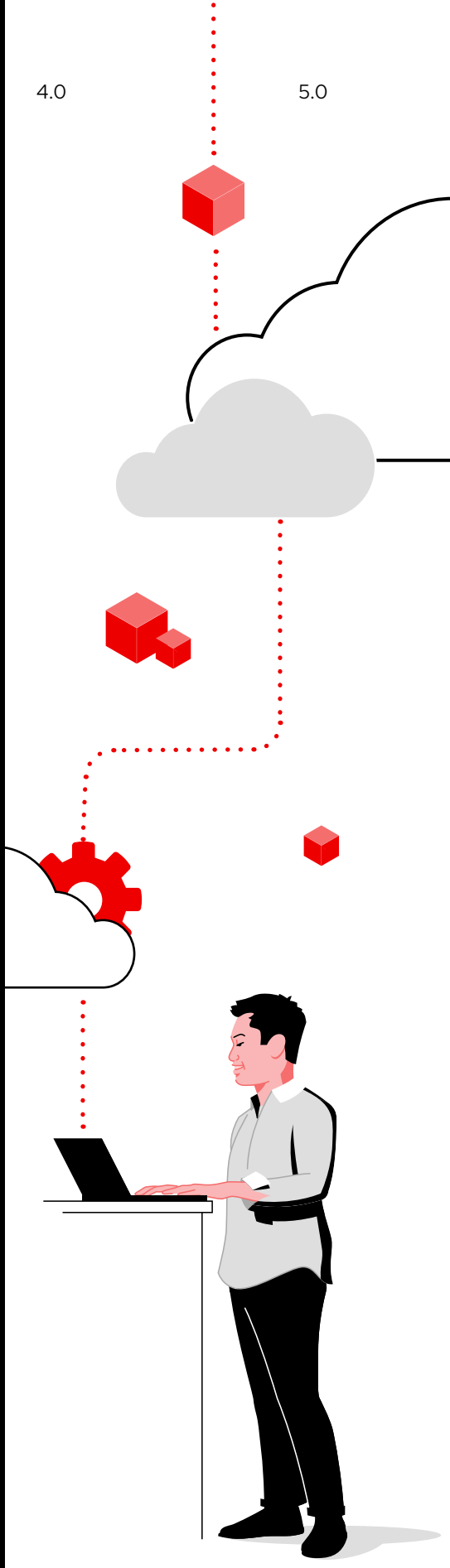
自動化によって人、プロセス、テクノロジーをつなげ、ビジネスアジリティ、イノベーション、価値を向上できます。

組織全体に自動化を導入する方法は、**e ブック「組織を自動化する」**をご覧ください。

ソフトウェアファクトリーのツール

ツールは、ソフトウェアファクトリーの重要な要素です。以下のカテゴリのツールをソフトウェアファクトリー内で使用し、自動化することをお勧めします。ツールの種類ごとに例を示しますが、他のツールを使用することもできます。

ツールのカテゴリ	例
プロジェクト管理	▸ Confluence と Jira を併用 ▸ Trello
ソースコード管理 (SCM)	▸ Github ▸ Gitlab
統合開発環境 (IDE)	▸ VS.code ▸ **Red Hat OpenShift Dev Spaces**
アーティファクト・リポジトリ	▸ Nexus ▸ Artifactory
CI/CD	▸ **Red Hat OpenShift Pipelines** ▸ **Jenkins**
ランタイム	▸ **Red Hat Runtimes** ▸ Golang
ビルド	▸ Maven ▸ Dotnet build
ユニットテスト	▸ JUnit ▸ NUnit
ソースコード分析	▸ Sonarqube ▸ Fortify
静的アプリケーション・セキュリティ・テスト (SAST)	▸ CheckMarx ▸ **Red Hat Advanced Cluster Security for Kubernetes**
ユーザー受け入れテスト	▸ Cucumber ▸ Cyprus
動的アプリケーション・セキュリティ・テスト (DAST)	▸ Veracode ▸ Synopsys
テレメトリ、メトリクス、ロギング	▸ **Prometheus** ▸ **Grafana** ▸ **Elasticsearch、Fluentd、および Kibana (EFK)** ▸ Splunk
サービスメッシュ	▸ Linkerd ▸ **Red Hat OpenShift Service Mesh**



ビルド、デプロイ、実行

プラットフォーム・アーキテクトや DevOps エンジニアは、開発者に代わってソフトウェアファクトリーを設定することがよくあります。ソフトウェアファクトリーを構築する場合は、ビルド、デプロイ、実行の 3 つの領域でセキュリティのベストプラクティスを検討してください。

ビルド

アプリケーションのセキュリティとコンプライアンスを制御します。

アプリケーションへのセキュリティの組み込みは、クラウドネイティブ・デプロイにとってきわめて重要です。

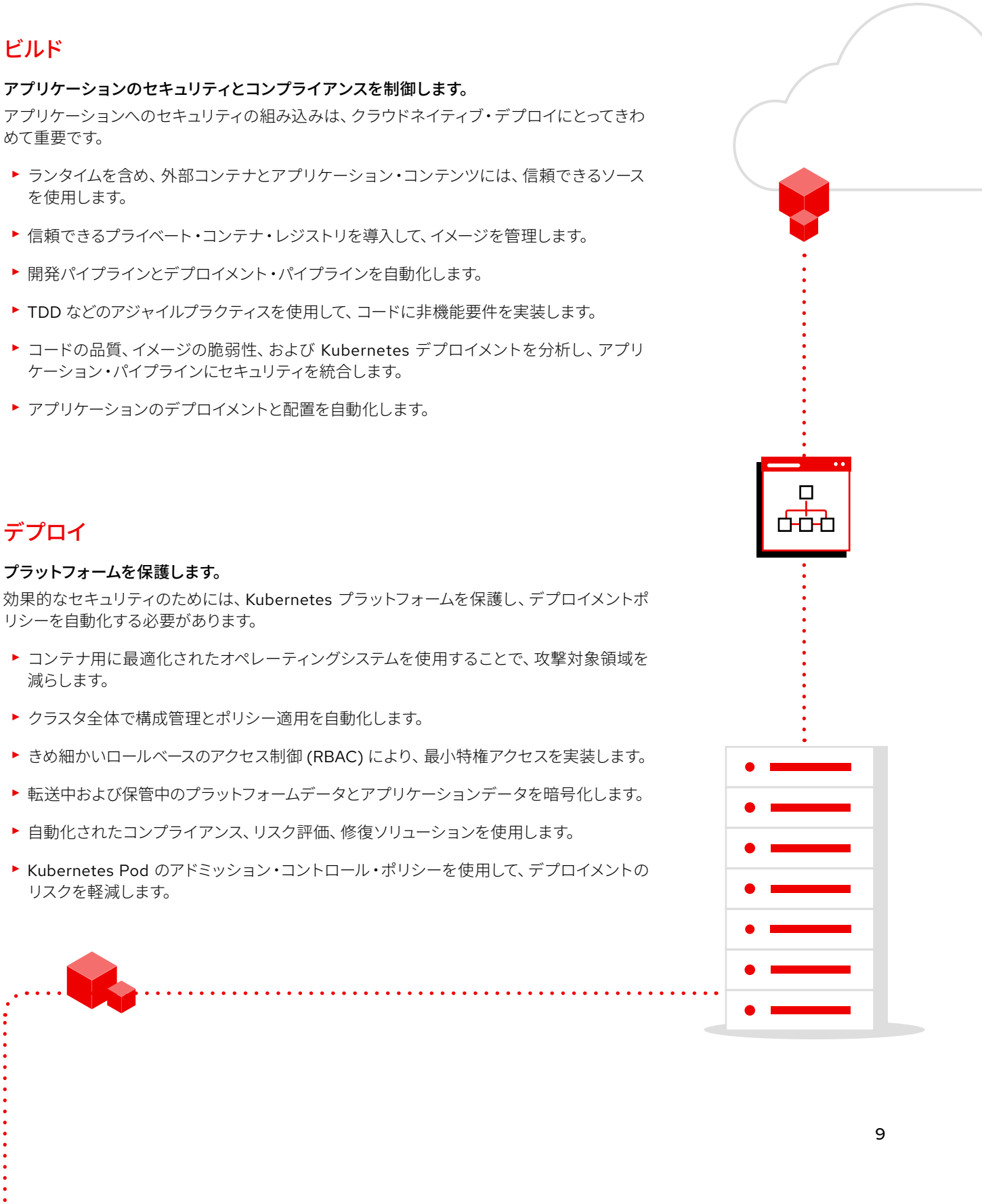
- ランタイムを含め、外部コンテナとアプリケーション・コンテンツには、信頼できるソースを使用します。
- 信頼できるプライベート・コンテナ・レジストリを導入して、イメージを管理します。
- 開発パイプラインとデプロイメント・パイプラインを自動化します。
- TDD などのアジャイルプラクティスを使用して、コードに非機能要件を実装します。
- コードの品質、イメージの脆弱性、および Kubernetes デプロイメントを分析し、アプリケーション・パイプラインにセキュリティを統合します。
- アプリケーションのデプロイメントと配置を自動化します。

デプロイ

プラットフォームを保護します。

効果的なセキュリティのためには、Kubernetes プラットフォームを保護し、デプロイメントポリシーを自動化する必要があります。

- コンテナ用に最適化されたオペレーティングシステムを使用することで、攻撃対象領域を減らします。
- クラスタ全体で構成管理とポリシー適用を自動化します。
- きめ細かいロールベースのアクセス制御 (RBAC) により、最小特権アクセスを実装します。
- 転送中および保管中のプラットフォームデータとアプリケーションデータを暗号化します。
- 自動化されたコンプライアンス、リスク評価、修復ソリューションを使用します。
- Kubernetes Pod のアドミッション・コントロール・ポリシーを使用して、デプロイメントのリスクを軽減します。



実行

コンテナのランタイムを保護します。

ランタイムのアプリケーションのセキュリティを維持します。

- Security-Enhanced Linux® (SELinux)、Security Context Constraints (SCC)、Kubernetes 名前空間、RBAC、およびネットワークポリシーにより、実行中のアプリケーションを分離します。
- クォータを使用して、リソースの競合や関連するパフォーマンスの問題を防ぎます。
- シングルサインオン・ユーザー管理、Ingress および Egress セキュリティ管理、暗号化された Pod 間トラフィック、およびアプリケーション・プログラミング・インタフェース (API) 管理により、アプリケーションアクセスを管理し、アプリケーションデータを保護します。
- プラットフォームとアプリケーションのアクティビティを監査および監視します。
- 異常な動作、特権昇格イベント、クリプトマイニングなどのリスクの高いプロセスを伴う Pod に対する脅威の検出と対応を自動化します。
- アドミッション・コントローラーを使用して、セキュリティポリシーに準拠していないコンテナのデプロイを防止します。
- サービスメッシュとネットワークポリシーを使用して、ゼロトラストネットワークを構築します。

セキュリティのヒント

Kubernetes で管理されるコンテナ化アプリケーションの保護について詳しくは、「**コンテナおよび Kubernetes セキュリティへの階層型アプローチ**」をお読みください。

ビルド	デプロイ	実行
アプリケーション・ライフサイクル	**フリート設定管理**	**フリートの可観測性とアラート**
脆弱性分析	ポリシー・アドミッション・コントローラー	ランタイム動作分析
アプリケーション設定分析	コンプライアンス評価	ネットワークポリシーの推奨事項
CI/CD 統合のための API	リスクプロファイリング	脅威の検出と対応
信頼できるコンテンツ	Kubernetes プラットフォームのライフサイクル	コンテナ分離
コンテナレジストリ	ID 管理とアクセス管理	ネットワーク分離
ビルド管理	プラットフォームデータ	アプリケーションのアクセスとデータ
CI/CD パイプライン	デプロイポリシー	可観測性

DevSecOps

エキスパートと共に DevSecOps を実装

Red Hat は、ハイブリッドクラウド環境でのアプリケーションのビルド、保護、デプロイに対応する認定パートナーエコシステム、広範な専門知識、革新的なプラットフォームをすべて備えています。当社は長年にわたって企業組織をサポートしており、業界のベストプラクティスとオープンソース・テクノロジーを使用して、技術上の課題やビジネス上の課題を克服できるよう支援しています。

Red Hat プラットフォームは、信頼できるコンテンツ・サプライチェーン、専任のセキュリティチームによるサポート、主要なセキュリティ機能のバックポートを備えており、DevSecOps ソリューションに最適な基盤を提供します。また、**トレーニングと認定のコース**、**インタラクティブなラボ**、**コンサルティング業務**、**マネージドサービス**を提供しており、DevSecOps の実践でより迅速に成功できるよう支援します。

Red Hat は、お客様の DevSecOps 実践の進捗状況にかかわらず、そのニーズに応えます。

当社の実績あるオープンソース・プラットフォームと専門家サービスにより、お客様は、現在必要なものをデプロイし、将来の変化に適応し、効率的かつ効果的な DevSecOps の導入に必要な手法とアプローチを習得できます。

Red Hat の DevSecOps を選ぶ理由について**詳細をご覧ください。**

DevSecOps への投資を最大限に活用

Red Hat サービスは、お客様が DevSecOps の実践を開始し、加速し、拡大するために必要なリソースを提供できます。

- **Red Hat Open Innovation Labs**
  お客様と Red Hat の社員がチームとして協力し、ビジネス成果を実現しながら DevSecOps などの新しい作業方法を学ぶ、研修スタイルのコンサルティング業務
- **Red Hat サービス・ソリューション：DevSecOps**
  モジュール式のアプローチを使用してソフトウェアファクトリーの実装を支援するサービス契約
- **Red Hat サービス・ジャーニー：Container Adoption**
  主要なワークストリームでのコンテナ導入に対処するコンサルティングサービス
- **Red Hat サービス・ジャーニー：Automation Adoption**
  組織全体に自動化を導入するプロセスを管理するためのフレームワークを提供するコンサルティングサービス

DevSecOps を成功させるプラットフォームをデプロイ

Red Hat OpenShift Platform Plus は、DevSecOps のための技術的基盤と独自のフレームワークを提供します。オンサイトとクラウド・インフラストラクチャで一貫して動作して拡張する、革新的なアプリケーション・プラットフォームです。Red Hat OpenShift Platform Plus は、主要なエンタープライズ向け Kubernetes プラットフォームと、組織の環境全体で一貫したアプリケーションのビルド、デプロイ、実行、保護、管理を行うための手段を兼ね備えています。マルチクラスタ管理ツールにより、Kubernetes クラスタを完全に可視化し、制御できます。Kubernetes ネイティブのセキュリティと DevSecOps 機能が、組織のソフトウェア・サプライチェーン、インフラストラクチャ、ワークロードを保護します。スケーラブルで世界中に分散されたレジストリとクラスタデータ管理により、組織の環境と情報が保護されます。

オープン統合インタフェースと Red Hat の**認定パートナーエコシステム**により、 Red Hat OpenShift Platform Plus で開発、テスト、運用、およびセキュリティ用の既存ツールと新規ツールを使用できます。多くのベンダーは、Red Hat プラットフォームでのソフトウェアのインストールと管理を単純化するために、**認定された Red Hat OpenShift Operator** や**認定されたソフトウェアコンテナ**を提供しています。また、多くのソフトウェア製品を **Red Hat Marketplace** から直接購入してデプロイすることもできます。そして、Red Hat は主要なクラウドプロバイダーのパートナーと連携して、組織内で構築する場合よりもコストを節約しながらデプロイと運用を効率化する、フルマネージドの **Red Hat OpenShift クラウドサービス**を提供します。

Red Hat OpenShift Platform Plus のコンポーネント

Red Hat OpenShift はエンタープライズ対応の Kubernetes アプリケーション・プラットフォームで、ハイブリッドクラウドとエッジのデプロイメントを管理するフルスタックの自動運用機能を備えています。生産性とスピードを向上させる開発者向けの機能が含まれています。

Red Hat Advanced Cluster Management for Kubernetes は、Kubernetes ドメイン全体を可視化するコンソールで、ガバナンス機能とアプリケーション・ライフサイクル管理機能が組み込まれています。

Red Hat Advanced Cluster Security for Kubernetes は、Kubernetes ネイティブのセキュリティ機能を提供するソリューションで、インフラストラクチャとワークロードの保護を強化し、アプリケーション・ライフサイクル全体を通じて可視性を向上します。

Red Hat Quay は、オープンソースのコンテナ・イメージ・レジストリで、ストレージを提供し、データセンターとクラウド環境でのコンテナの構築、配布、デプロイを可能にします。

Red Hat OpenShift Data Foundation は、スケーラブルなデータおよびストレージのサービスレイヤーで、Red Hat OpenShift 環境にデータ効率、回復力、セキュリティを提供します。

Red Hat OpenShift Platform Plus は、DevSecOps 導入プロセスのあらゆる時点でお客様をサポートします。お客様の現在の状況に対応し、お客様自身のペースで前進するための基盤を提供します。

組み込み型のセキュリティ機能

システムレベルのデータ収集および分析に加え、アプリケーションのライフサイクル全体を通して適用、実施可能な 60 以上の組み込みセキュリティポリシーを用いて、実行中のワークロードにセキュリティ上の問題や脅威がないかを監視します。

一貫性のあるオペレーション

オンサイト・データセンターとクラウド・インフラストラクチャにまたがる Red Hat OpenShift クラスタに、セキュリティ、構成、コンプライアンス、ガバナンスのための一貫した運用ポリシーを適用します。

開発者用ツール

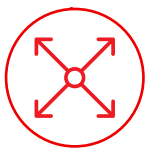
サポート対象のビルドツール、言語、パイプライン、およびフレームワークのライブラリが組み込まれており、アプリケーションをより迅速に作成、実行、デプロイできます。Operator Framework は、Red Hat OpenShift で動作するようにテストおよび検証された最新の開発者ツールの統合を実現します。

エンドツーエンドの管理

さまざまな Kubernetes ディストリビューションに基づく環境を含め、オンサイト環境、クラウド環境、エッジ環境にわたって機能する、管理者および開発者向けの統合インタフェースにより、Red Hat OpenShift 環境を一貫して管理します。

DevSecOps のサポート

宣言型セキュリティを開発者のツールやワークフローに統合します。Kubernetes ネイティブのコントロールを使用して、脅威を軽減し、セキュリティポリシーを施行し、運用リスクを最小限に抑えます。



スケーラブルなデータサービス

クラスタ全体のデータ管理を効率化します。Red Hat OpenShift Data Foundation は、ファイル、ブロック、およびオブジェクト・データ・プロトコルのサポートにより、ステートフル・アプリケーションとクラスタサービスに回復力のある永続ストレージを提供します。

ゼロトラストネットワーク機能

ゼロトラストネットワークを実装して、アプリケーションとサービス間で回復力があり、安全で観測可能な通信を提供します。**Red Hat OpenShift Service Mesh** が含まれており、Red Hat OpenShift と統合されているため、通信の保護が容易になります。

Red Hat OpenShift Platform Plus は、効果的な DevSecOps 導入に必要なテクノロジーと機能を提供します。**Red Hat OpenShift セキュリティガイド**で、テクノロジースタック全体でセキュリティに対処する方法をご覧ください。

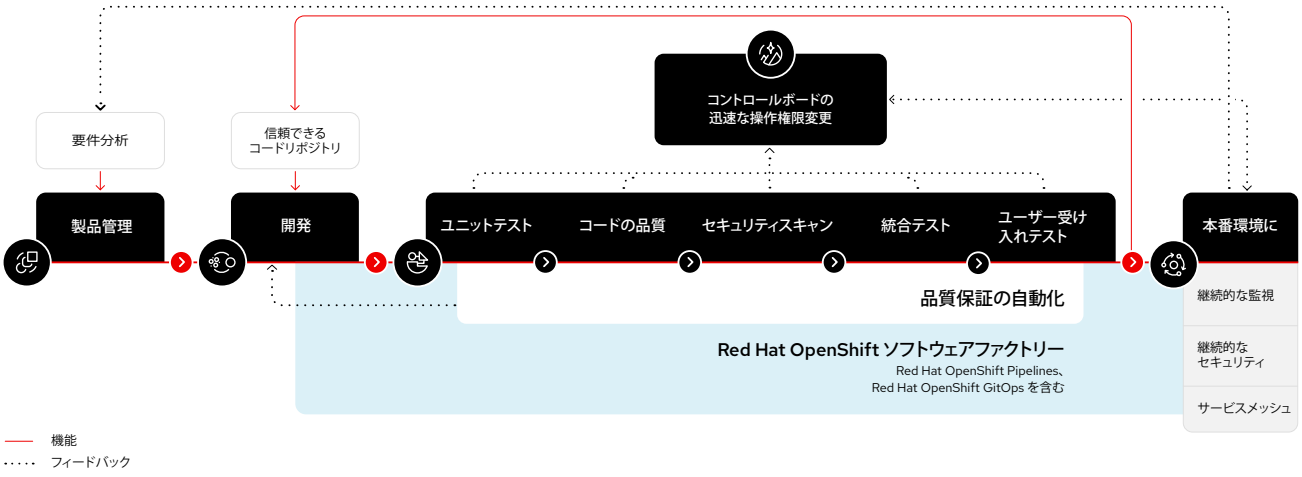
Red Hat OpenShift クラウドサービスで導入を加速

Red Hat OpenShift クラウドサービスは、**AWS**、**Google Cloud**、**IBM Cloud**、**Microsoft Azure** で利用できるため、組織のニーズに最適なオプションを選択できます。各サービスを通じて、必要なすべてのサービス、シンプルなセルフサービスオプション、厳格なサービスレベル契約 (SLA) に基づく 24 時間年中無休のエキスパートによるサポートを備えたフルスタック環境がお客様に提供されます。

詳細は、「**概要：Red Hat OpenShift クラウドサービスでさらに多くのことを実現**」をご覧ください。

Red Hat OpenShift Platform Plus でソフトウェアファクトリーの基盤を構築

Red Hat OpenShift Platform Plus は、信頼性と適応性に優れた構成可能なソフトウェアファクトリーの基盤を提供します。これにより、セキュリティチェックを CI/CD パイプラインに組み込んで、開発者に既存のワークフロー内の自動化されたガードレールを提供し、ワークロードと Kubernetes インフラストラクチャを構成ミスやコンプライアンス違反から保護し、ランタイムの脅威の検出と対応を実装できます。



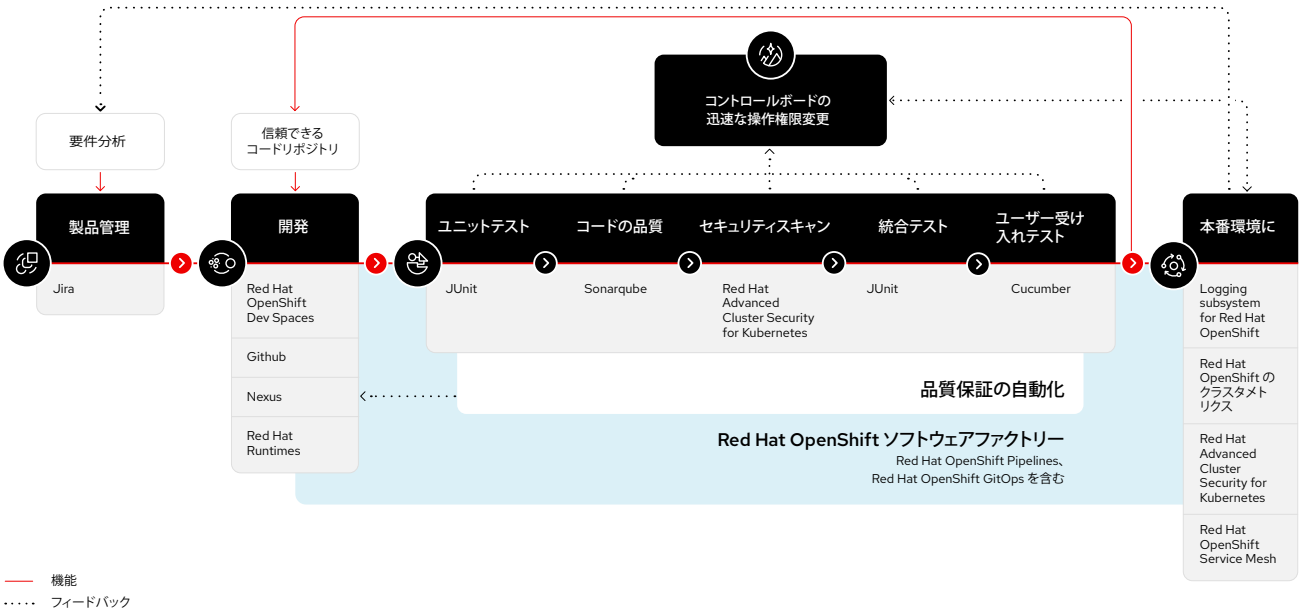
サードパーティ製ツールのエコシステムで完全なソフトウェアファクトリーを構成

ユースケースごとに、ソフトウェアファクトリー内でさまざまなツールが必要になります。Red Hat OpenShift Platform Plus を基盤として、以下のようなサードパーティ製品やテクノロジーを好みに応じて使用し、ソフトウェアファクトリーの各ステージを構成できます。

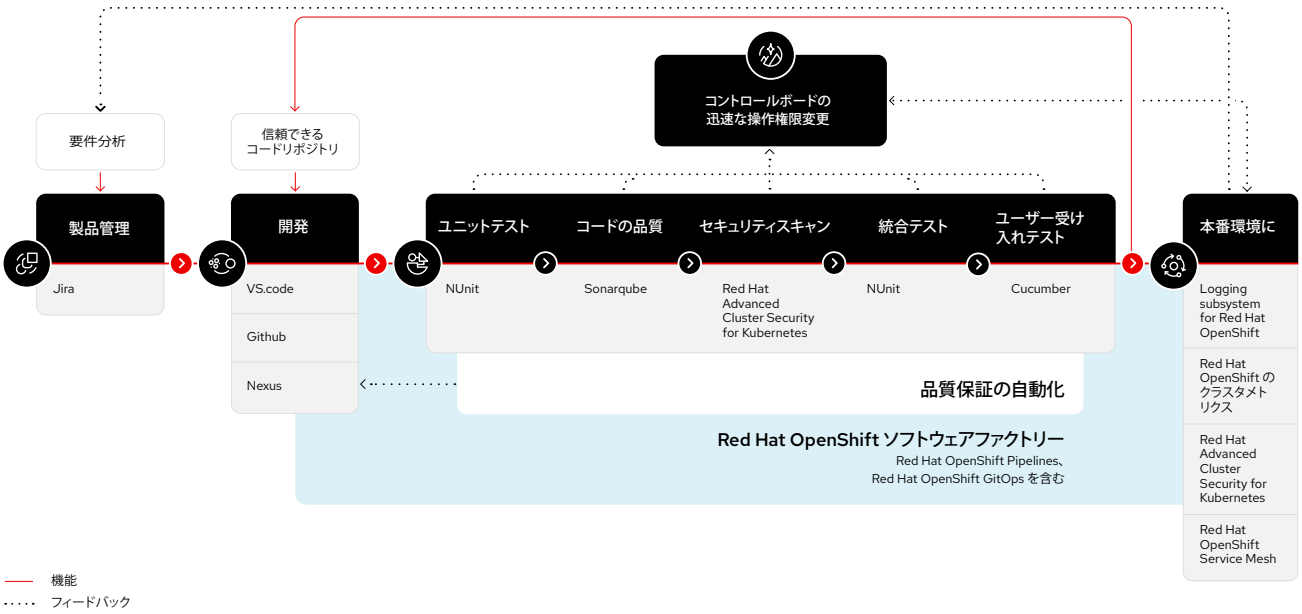
- 特権アクセス管理ツール
- 外部認証局
- 外部 Vault および鍵管理ソリューション
- コンテナ・コンテンツ・スキャナーおよび脆弱性管理ツール
- コンテナランタイム分析ツール
- セキュリティ情報およびイベント管理 (SIEM) システム
- ソースコントロール管理ツール
- アーティファクト・リポジトリ
- ソフトウェアテストツール

たとえば、Spring Boot アプリケーションのクラウドネイティブ開発用のソフトウェアファクトリーは、.Net Core アプリケーション用のソフトウェアファクトリーとは異なるランタイムツール、ビルドツール、テストツールを使用します。Red Hat ソフトウェアファクトリー基盤の柔軟性を示すために、これらのソフトウェアファクトリーについて可能な構成を以下に示します。

マイクロサービスベースの Spring Boot アプリケーションのクラウドネイティブ開発用ソフトウェアファクトリー



マイクロサービスベースの .Net Core アプリケーションのクラウドネイティブ開発用ソフトウェアファクトリー



成功事例を見る

世界最大の天然ガスネットワーク会社の 1 つである Snam は、Red Hat OpenShift、Red Hat Quay、Microsoft Azure Red Hat OpenShift などの Red Hat テクノロジーとサービスを導入して、組織のデジタル・トランスフォーメーションを推進しています。現在は、アプリケーションを自動化された方法でわずか 30 分でデプロイできるようになり、新しいソフトウェア製品の提供にかかる時間が 10 倍以上短縮されました。また、将来のビジネス要件に対応できるよう、パブリッククラウドやプライベートクラウドでワークロードとアプリケーションをスケーリングすることも可能で、クラウドロックインに関連する潜在的なリスクを軽減しています。

> Red Hat OpenShift は当社の変革プロジェクトの基盤です。おかげで、効率的で高性能で信頼性の高い IT プラットフォームを作成することができ、複雑なシステムとアプリケーションの管理がシンプルになりました。

Roberto Calandrini 氏
Snam デジタルおよび AI サービス アーキテクチャ統括責任者

詳細はこちら

オランダの消費者および企業向けの通信とエンターテイメントの大手サービスプロバイダーである VodafoneZiggo は、Red Hat OpenShift に基づくハイブリッドクラウド・プラットフォームを導入して、組織のアプリケーション・インフラストラクチャを統合しました。また、Red Hat コンサルティングと連携して、DevSecOps の採用と、よりオープンで協調的な文化への移行に関するガイダンスを提供しました。VodafoneZiggo は、ビジネスニーズや市場の要求の進化に合わせて、複数のクラウドからエッジへと、より迅速かつ効率的にスケールアウトできるようになりました。

> Red Hat OpenShift は、生産性を向上させて継続的なイノベーションを実現できる、クラウドネイティブなアプリケーションとサービスの一貫したレイヤーであると考えています。

André Beijen 氏
VodafoneZiggo モバイルネットワーク 取締役

詳細はこちら

DevSecOps の導入を始める

クラウドネイティブの世界では、スピード、スケール、セキュリティが重要です。

Red Hat OpenShift Platform Plus に基づくソフトウェアファクトリーは、開発を加速し、運用を効率化し、ビジネスを保護する DevSecOps の実践で成功するために役立ちます。

Red Hat OpenShift を無料で試す：

cloud.redhat.com/try

Red Hat OpenShift Platform Plus の詳細：

red.ht/openshift-platform-plus

F31657_0522_KVM

Red Hat