

# Développer et déployer des conteneurs à l'aide des solutions Red Hat et AWS



## Accélérez la conteneurisation avec Red Hat et AWS

Les avancées en matière de technologies cloud, les microservices et les perturbations du marché pressent actuellement le passage au numérique des entreprises. Dans leur quête de rapidité, d'agilité et de flexibilité, elles s'appuient sur les applications pour mener à bien leurs projets numériques. Chaque seconde compte. Le délai de déploiement moyen est passé à quelques semaines seulement.

Les conteneurs d'applications accélèrent la distribution. Le secret ? L'image de conteneur, c'est-à-dire un fichier statique et inaltérable qui contient du code exécutable et qui peut s'exécuter sur l'infrastructure comme un processus indépendant. Cette image est composée de bibliothèques système, d'outils système et d'autres paramètres de plateforme nécessaires à l'exécution d'un logiciel sur une plateforme de conteneurisation. Elle partage le noyau opérationnel de la machine hôte.

Les conteneurs assurent la cohérence du processus, car les environnements de développement et de déploiement sont identiques. Résultat : le cycle de publication des versions s'accélère, puisque chaque étape est comparée à une représentation standard et immuable de l'application avant validation.

Les conteneurs Linux® représentent un outil idéal pour la majorité des projets de transformation numérique, car ils tirent parti de l'isolation des processus dans Linux en parallèle des espaces de noms afin de créer des processus isolés. Avec ses nombreuses options de création de conteneurs, la solution Red Hat® Entreprise Linux est couramment utilisée pour l'exécution des conteneurs Linux en entreprise. Ces options simplifient la mise en route des conteneurs, mais les équipes de développement ont également besoin de gérer les déploiements et l'orchestration.

Dans ce livre numérique, vous découvrirez comment les outils de conteneurs Red Hat et les outils de développement Amazon Web Services (AWS) facilitent la conteneurisation et la distribution des applications.

## Les avantages des conteneurs d'applications Linux

Les conteneurs d'applications Linux aident les services informatiques à répondre plus rapidement à la demande. L'équipe de développement commence par segmenter les applications en microservices modulaires, qu'elle déploie ensuite sous la forme de conteneurs Linux. Les composants d'application ainsi conteneurisés prennent peu de place et se déploient très rapidement.

Les applications conteneurisées présentent une excellente portabilité entre les architectures, car elles sont indépendantes du matériel et du système d'exploitation. Ce format simplifie le déploiement et l'exécution des applications dans différents environnements, que ce soit sur un ordinateur portable ou dans des clusters de production. Il dissocie également les mises à niveau et restaurations des applications des serveurs où elles sont exécutées. Les conteneurs réduisent la complexité de l'exploitation et améliorent sa modularité ainsi que sa flexibilité, allégeant ainsi la charge de travail de l'équipe d'administration. Les bénéfices en termes de modularité et de flexibilité augmentent encore avec l'utilisation de systèmes d'orchestration tels que Kubernetes et Red Hat OpenShift pour la gestion des conteneurs.

Les conteneurs Linux peuvent réduire les coûts liés à l'infrastructure. Petits et légers, ils sont faciles à déplacer entre les systèmes bare metal, ainsi que dans les environnements de cloud public, privé, hybride et multicloud dotés du même système d'exploitation. Ils accélèrent le développement des nouvelles applications, contribuent à l'optimisation des anciennes et les connectent toutes, à condition qu'elles soient compatibles avec le système d'exploitation sous-jacent.

Privilégiez les conteneurs par rapport aux machines virtuelles pour les tâches suivantes :

- ▶ Création d'applications cloud-native
- ▶ Mise en paquet des microservices
- ▶ Adoption de pratiques DevOps ou d'intégration et de distribution continues (CI/CD)
- ▶ Déplacement des projets informatiques évolutifs entre différents environnements qui partagent le même système d'exploitation

Les machines virtuelles sont, quant à elles, capables d'exécuter davantage d'opérations et s'utilisent donc traditionnellement pour les workloads monolithiques. Toutefois, les fonctionnalités étendues de ces machines virtuelles nuisent à leur portabilité, puisqu'elles dépendent de leur système d'exploitation, de leurs applications et de leurs bibliothèques.

Privilégiez les machines virtuelles par rapport aux conteneurs pour les tâches suivantes :

- ▶ Prise en charge des charges de travail monolithiques, traditionnelles et d'ancienne génération
- ▶ Isolation des cycles de développement risqués
- ▶ Provisionnement des ressources d'infrastructure (réseaux, serveurs et données)
- ▶ Exécution d'un système d'exploitation au sein d'un autre système d'exploitation (Unix sous Linux, par exemple)

La réduction du nombre de machines virtuelles et de systèmes d'exploitation dans un environnement entraîne une baisse considérable du nombre de licences requises. Les conteneurs Linux permettent de diminuer encore ces dépenses. Les outils et plateformes Red Hat et AWS facilitent la création d'applications conteneurisées cloud-native ainsi que l'adoption de pratiques DevOps et CI/CD pour augmenter l'agilité et réduire le délai de mise sur le marché.

### **Le développement de conteneurs sous Red Hat Enterprise Linux**

Les solutions Red Hat vous permettent de créer des conteneurs d'applications sous Red Hat Enterprise Linux. Red Hat Enterprise Linux 8 inclut d'ailleurs un kit d'outils de conteneurisation basé sur des normes ouvertes que vous pouvez associer aux kits de développement logiciel (SDK) et de développement cloud (CDK) d'AWS afin de créer et tester des applications d'entreprise.

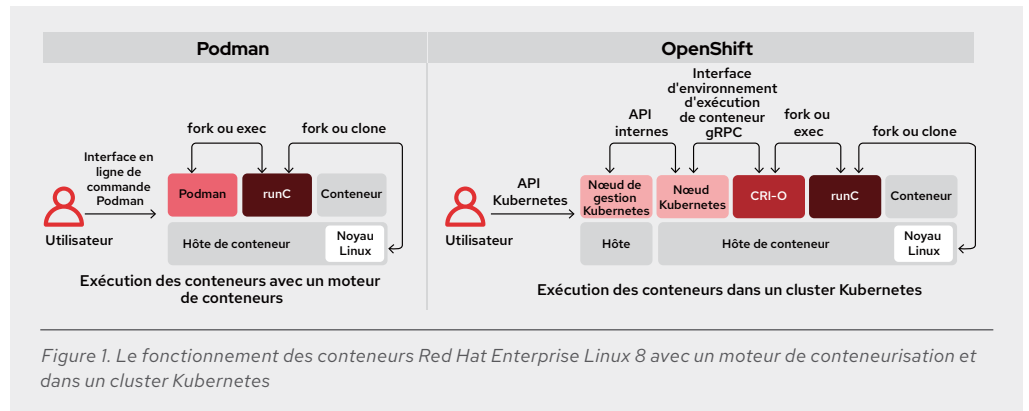
Une fois vos conteneurs développés, ils passent en phase de déploiement. Si vous utilisez Kubernetes pour l'orchestration, vous risquez de rencontrer des problèmes de complexité, d'évolutivité et de découverte de service qui freineront votre progression. La plateforme Red Hat OpenShift vous évite les éventuelles difficultés liées à Kubernetes ainsi qu'à la gestion et au déploiement des conteneurs, afin que vous puissiez vous concentrer sur la création du code. Sous Red Hat Enterprise Linux, un petit ensemble d'outils flexible suffit à développer simplement des conteneurs.

### **Le développement de conteneurs sous Red Hat Enterprise Linux 8**

La solution Red Hat Enterprise Linux simplifie le développement des conteneurs en réduisant le nombre de référentiels et en proposant des outils supplémentaires aux développeurs. En plus de ces outils, Red Hat fournit des images de base à utiliser pour créer vos propres images. Les images de base couvrent divers cas d'utilisation, de la création d'applications métier avec Node.js, PHP, Java™ et Python, aux fonctions d'infrastructure telles que la journalisation, la collecte des données et l'authentification.

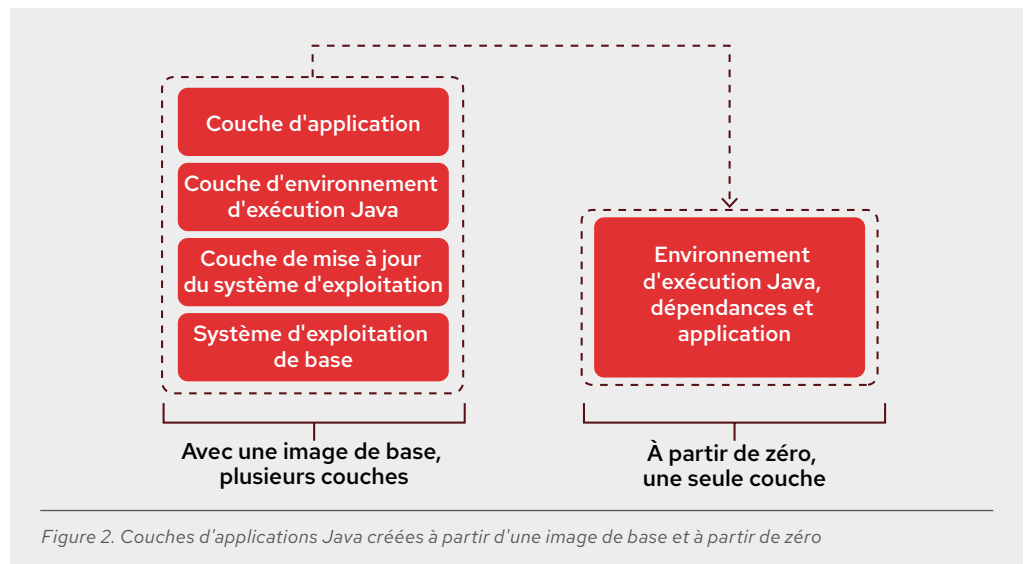
La version Red Hat Enterprise Linux 8 comprend de nouvelles fonctions pour les outils en ligne de commande, notamment Buildah, Podman, runC et Skopeo, les images UBI, le référentiel Red Hat Quay et le référentiel Supplementary, tout en réduisant la complexité du développement des conteneurs.

Le schéma ci-dessous montre l'architecture d'un conteneur Red Hat Enterprise Linux 8.



## Buildah

**Buildah** permet de créer un conteneur à partir de zéro, sans Docker ni démon. Cette option s'avère intéressante par exemple pour les applications Java aux différents stades de leur cycle de vie. En effet, si une image de conteneur nécessite potentiellement le compilateur Java, Maven, ainsi que d'autres outils pendant la phase de développement, elle ne requerra peut-être que l'environnement d'exécution Java et vos paquets une fois en production.



Puisque Buildah fonctionne sans démon, il s'exécute facilement dans un conteneur sans nécessiter la configuration d'une infrastructure spéciale sur l'hôte.

Buildah ne fournit que les éléments de base requis pour créer ou modifier des images de conteneurs conformes à l'Open Container Initiative (OCI), simplifiant ainsi son intégration aux pipelines de versions existants. Par exemple, il assemble des conteneurs qui ne contiennent pas de gestionnaire de paquets (DNF/YUM), si l'image finale n'en a pas besoin. Buildah permet de créer ces conteneurs facilement en intégrant la sécurité de manière cohérente. Il élimine également les composants et opérations superflus pour réduire la taille des images et étend les options de personnalisation pour répondre aux besoins de vos applications cloud-native.

Cet outil s'avère particulièrement intéressant pour la création d'images à partir de scripts Bash, un processus similaire à la rédaction d'un fichier Dockerfile.

### **Podman**

Compatible avec l'interface en ligne de commande Docker, l'outil [Podman](#) assure la gestion des conteneurs sans installer de démon. Il autorise des interactions directes avec le registre d'images, le stockage de conteneurs et d'images, ainsi que le noyau Linux à travers le processus de l'environnement d'exécution de conteneurs runC.

[Il est possible d'exécuter Podman sans privilèges root](#) et sans démon fournissant des droits root. Par contre, vous aurez besoin d'un emplacement séparé où Podman peut écrire ses images. L'outil Podman utilise un référentiel dans le répertoire principal de l'utilisateur afin d'éviter que le dossier `/var/lib/containers` soit accessible à tout le monde en écriture ou qu'il puisse être modifié de toute autre manière susceptible d'engendrer des problèmes de sécurité. Avec cette approche, chaque utilisateur dispose aussi de son propre ensemble de conteneurs et d'images, ce qui permet à plusieurs personnes d'utiliser Podman en même temps, sur le même hôte, sans interférence. Une fois leur travail terminé, les utilisateurs peuvent mettre leurs images modifiées à disposition des autres dans un registre commun.

### **runC**

Podman et Buildah utilisent [runC](#), [l'environnement d'exécution OCI](#), pour lancer des conteneurs. On peut soit créer et exécuter une image, soit exécuter des images au format docker avec runC. Cet outil basé sur le langage Go lit les caractéristiques d'un environnement d'exécution, configure le noyau Linux et crée, puis démarre, les processus de conteneurs. Bien qu'il soit possible de partager le code de base de runC avec Docker, runC ne dépend pas des composants de cette plateforme. Cet environnement d'exécution prend en charge les espaces de noms Linux ainsi que la migration dynamique et dispose de profils de performances portables. Il prend également en charge les fonctions de sécurité Linux, notamment SELinux, cgroups et seccomp.

### **Skopeo**

[Skopeo](#) inspecte les images et les transfère dans les emplacements capables de stocker des images OCI. Avant sa sortie, il fallait extraire l'image tout entière pour l'examiner, même pour examiner seulement certaines métadonnées. [Cet outil affiche toutes les propriétés d'une image](#), y compris ses couches, ce qui évite d'avoir à l'extraire de son hôte. Si besoin à l'issue de l'inspection, il permet aussi de copier une image de conteneur d'un emplacement vers un autre.

Skopeo permet également de supprimer une image d'un référentiel et de synchroniser un référentiel d'images externe avec un registre interne pour réaliser des déploiements hors réseau ou air gap. Il peut s'authentifier à l'aide d'informations d'identification et de certificats si le référentiel l'exige.

### Image de base universelle

Avant, il fallait créer des conteneurs Linux pour chaque plateforme. Grâce aux [images de base universelles \(UBI\)](#), ce n'est plus le cas. Une UBI est une image de conteneur de base pour entreprise sur laquelle vous pouvez créer et distribuer des applications. Sous Red Hat Enterprise Linux 8, toutes les images de base Red Hat existent au format UBI. Ce type d'image permet de créer et redistribuer des images de conteneurs basées sur Red Hat Enterprise Linux sans souscription Red Hat. Les utilisateurs non plus n'en auront pas besoin.

Basées sur les images de base Red Hat Enterprise Linux éprouvées, les [UBI](#) servent de fondation au développement en conteneur des applications cloud-native et web. De plus, elles vous épargnent la création d'images de conteneurs CentOS pour les projets communautaires ou simplement si vous préférez gérer l'assistance vous-même.

Vous pouvez créer une [application conteneurisée à partir d'une image UBI](#), la publier sur le serveur de registre de votre choix, la passer en production, puis la partager avec qui vous voulez. Et puisqu'elle est redistribuable librement, vous pouvez même la déployer sur des plateformes d'autres fournisseurs. Cette option s'avère utile si vous exécutez Ubuntu ou une autre distribution Linux et vous aide à vous familiariser avec Red Hat Enterprise Linux, même si vous utilisez une autre plateforme.

### Référentiel dans Red Hat Quay

Il est également possible de créer un référentiel d'images dans [Red Hat Quay](#), un registre de conteneurs privé qui stocke, crée et déploie des images de conteneurs. Il analyse vos images à la recherche de vulnérabilités et identifie les problèmes potentiels afin d'écartier les risques pour la sécurité. Pour créer un référentiel dans Red Hat Quay, vous avez deux options : soit vous utilisez la commande push (à partir de Docker ou Podman), soit vous passez par l'interface utilisateur de Red Hat Quay. Ce registre vous donne également accès aux référentiels publics.

### Référentiel Supplementary

Le [référentiel Supplementary](#) inclut des paquets sous licence propriétaire non compris dans les référentiels Red Hat Enterprise Linux Open Source. Bien qu'ils ne soient pas pris en charge par Red Hat Enterprise Linux 8, ces paquets permettent d'utiliser d'autres logiciels pour la conteneurisation d'applications.

### Red Hat Enterprise Linux sur Amazon Elastic Compute Cloud (EC2)

Red Hat Enterprise Linux sur Amazon Elastic Compute Cloud (Amazon EC2) est une combinaison entre la plateforme de calcul Amazon EC2 et le système d'exploitation Red Hat Enterprise Linux, où Red Hat se charge de la maintenance des images de base Red Hat Enterprise Linux pour Amazon EC2. Vous recevez les mises à jour dès leur publication par Red Hat. La fiabilité de votre environnement informatique et de votre sécurité est assurée, tandis que les applications certifiées Red Hat Enterprise Linux garantissent la compatibilité. Ensemble Red Hat Enterprise Linux et Amazon EC2 forment un environnement de développement virtuel et une plateforme fiable pour la distribution d'applications très variées, y compris en conteneur. L'architecture distribuée vous permet de déployer vos applications de manière plus flexible et agile.

Red Hat Enterprise Linux sur EC2 vous donne accès aux SDK AWS pour les langages courants, notamment Go, l'un des plus utilisés pour les conteneurs avec C++, Java et Rust. Tous sont également compatibles avec les applications conteneurisées. Il est aussi possible de combiner le SDK AWS avec les composants de base de conteneur inclus dans Red Hat Enterprise Linux. Enfin, vous pouvez tirer parti des flux d'applications et d'UBI Red Hat pour créer, partager et modifier à tout moment vos applications conteneurisées.

Comme vous pouvez le constater, les outils de développement de conteneurs sous Red Hat Enterprise Linux 8 constituent une option viable pour le développement, la création et la gestion de conteneurs individuels et d'images de conteneurs. Toutefois, [Red Hat OpenShift®](#) améliore encore l'évolutivité et l'efficacité du développement et du déploiement.

## **Red Hat OpenShift on AWS, pour associer le développement et le déploiement des conteneurs**

Vous créez des applications cloud-native, mettez des microservices en paquets, utilisez des pratiques DevOps ou CI/CD ou déplacez des projets informatiques évolutifs entre différents environnements qui partagent le même système d'exploitation ? Si oui, adoptez les conteneurs. À la fois portables et légers, ils ne sont pas liés au système d'exploitation, ni aux applications et aux bibliothèques. Toutefois, ils posent aussi un certain nombre de défis. En effet, si vous décidez de les adopter, il vous faudra trouver le moyen de les déployer, remplacer et mettre à l'échelle, de gérer un stockage persistant et d'assurer une découverte efficace des services.

Et si vous n'utilisez qu'un environnement d'exécution de conteneurs, la complexité risque de freiner votre activité. Toute expansion de l'exploitation implique l'ajout de nouveaux conteneurs et la création d'une chaîne de processus séparés. Au final, le nombre de conteneurs sur un même serveur peut grimper jusqu'à en affecter les performances. Difficile également de suivre l'état du stockage lorsqu'il est dissocié des conteneurs (volumes). Aussi, prévoyez absolument une couche d'orchestration pour le déploiement des conteneurs. Kubernetes est un outil idéal pour ce type de tâche, mais le processus de déploiement de votre propre système Kubernetes natif risque de s'avérer complexe et problématique. Il est possible d'utiliser Red Hat Quay avec Kubernetes pour gérer vos images et la configuration nécessaire à la création d'une application complète, parfaitement fonctionnelle. Mais si vous optez pour Red Hat OpenShift, sachez que cette solution élimine presque entièrement la complexité liée à Kubernetes et vous permet d'accélérer considérablement vos déploiements.

### **Outils de développement disponibles dans Red Hat OpenShift**

- ▶ Les espaces de travail [CodeReady Workspaces](#) permettent aux équipes de développement à distance de provisionner et de partager leurs environnements en un clic seulement. Résultats : accélération de l'activation et interactions à faible latence.
- ▶ Le [framework Quarkus](#) permet la création d'applications Java natives pour Kubernetes.
- ▶ La prise en charge d'aperçus pour [Buildpacks](#) et [Kaniko](#) est disponible en parallèle des versions S2I et Dockerfile via [Buildah](#).
- ▶ [Helm](#) simplifie l'utilisation des charts et des versions.

### **Déployer des conteneurs sur Red Hat OpenShift**

La solution [Red Hat OpenShift](#) a été conçue pour simplifier le déploiement et la gestion des plateformes de conteneurs. Elle automatise les opérations de maintenance et intègre les mises à niveau ainsi que la surveillance de plateforme, ce qui la rend particulièrement utile aux administrateurs.

Red Hat OpenShift s'avère idéale pour les stratégies de type « rédiger une fois, exécuter partout » qui vous permettent d'exécuter vos charges de travail là où vous en avez le plus besoin, que ce soit sur site ou dans le cloud public, privé ou même hybride. Sur AWS, cette solution est intégrée dans un autre service, listé dans la console AWS, avec lequel vous pouvez créer et gérer de manière cohérente vos clusters OpenShift plus simplement et rapidement. Vous profitez d'un système de facturation à la demande, avec facture unique et de la possibilité de contacter AWS directement en cas de besoin. Vous avez le choix entre les paramètres par défaut et les paramètres AWS personnalisés pour le [déploiement de votre cluster Red Hat OpenShift](#). Il est également possible de déployer un cluster sur une infrastructure AWS que vous provisionnez vous-même. Enfin, les modèles AWS CloudFormation fournis sont entièrement modifiables en fonction de vos besoins.

### **Développer des conteneurs sur Red Hat OpenShift**

Red Hat OpenShift inclut le framework Source-to-Image (S2I) qui simplifie considérablement le développement des conteneurs grâce au passage direct du code de l'application au conteneur. Il vous suffit de rédiger des images avec en entrée le code source de l'application et en sortie une image qui exécute l'application assemblée. En plus du framework S2I, il existe une deuxième option de création dans OpenShift : les fichiers Dockerfile.

Les fichiers Dockerfile fournis par l'utilisateur invoquent Buildah (inclus dans l'outil de création OpenShift) pour créer l'image. Buildah ajoute également des balises et transmet l'image finale. Vous pouvez aussi utiliser S2I pour générer un fichier Dockerfile en suivant le même flux que pour la version Dockerfile basée sur Buildah. L'image de l'outil de création qui en résulte contient les informations nécessaires pour produire une image exécutable (ou artéfact de version)

Comme en témoigne le framework S2I, la plateforme Red Hat OpenShift s'avère aussi efficace pour le développement que pour le déploiement de conteneurs. Elle permet aux équipes de développement de créer sur une plateforme Kubernetes au lieu de se conformer aux spécificités des infrastructures AWS en place. De plus, elle offre diverses options de développement d'applications conteneurisées, afin que vous puissiez utiliser la plus adéquate à chaque situation. Cette plateforme s'adresse aussi bien aux développeurs qui connaissent mal Kubernetes ou qui ne s'intéressent qu'au code, qu'à ceux qui maîtrisent Kubernetes et cherchent un maximum de flexibilité.

La solution Red Hat OpenShift prend en charge les langages, bases de données et outils que les équipes de développement utilisent déjà. Elle facilite également l'accès aux services de développement d'applications, tels que ceux proposés par AWS.

## Outils de développement des conteneurs sur Red Hat OpenShift

Les outils AWS simplifient encore le développement de conteneurs Red Hat OpenShift et leur usage s'étend également à d'autres plateformes, telles que Red Hat Enterprise Linux.

**AWS CodeArtifact** est un service de référentiel d'artéfacts entièrement géré qui aide les entreprises de toutes tailles à sécuriser le stockage, la publication et le partage des paquets logiciels utilisés dans leur processus de développement.

**AWS Cloud9** est un environnement de développement intégré qui permet d'écrire, d'exécuter et de déboguer du code depuis une interface de shell basée sur navigateur pour vous permettre d'installer des logiciels supplémentaires, d'exécuter un git push ou d'entrer des commandes.

**AWS Cloud Development Kit** exploite la puissance expressive de langages de programmation courants pour modéliser des applications et accélérer l'intégration à AWS, car il n'introduit que peu de nouveaux concepts à apprendre.

**Amazon CodeGuru** fournit des informations et des recommandations pour corriger les problèmes de performance et estime le coût des codes non efficaces, aidant ainsi les équipes de développement à définir leurs priorités. Cet outil s'intègre à votre workflow de développement logiciel existant sur Red Hat OpenShift et formule des suggestions pour améliorer la qualité du code et identifier les lignes de code les plus coûteuses dans une application.

## Lancez-vous dans la création, le déploiement et la gestion des conteneurs

La création de conteneurs d'applications ne doit pas nécessairement être complexe. Les outils de conteneurs de Red Hat et de développement d'AWS peuvent vous aider dans cette démarche. Le déploiement et l'orchestration des conteneurs sont les dernières étapes de leur mise en œuvre et peuvent se révéler très complexes. Red Hat OpenShift élimine presque entièrement la complexité liée à Kubernetes pour vous permettre de déployer et gérer vos conteneurs à grande échelle, où qu'ils soient construits.

Red Hat OpenShift n'est pas une simple solution de déploiement. Elle offre également aux équipes de développement de nombreuses options pour la création des conteneurs sur la plateforme avant leur déploiement. La solution Red Hat OpenShift se décline en deux versions sur AWS : autogérée ou entièrement gérée. Vous pouvez déployer et gérer vos propres mises en œuvre d'OpenShift sur AWS. Red Hat OpenShift Service on AWS et Red Hat OpenShift Dedicated s'adaptent à l'ensemble de l'offre Red Hat Cloud Services. Ainsi, vous pouvez vous concentrer sur la création et la mise sur le marché de vos applications.

### Autres outils de développement dans Red Hat OpenShift

- ▶ **Odo** permet l'itération du code dans son interface en ligne de commande. Il prend en charge Kubernetes et Red Hat OpenShift, un modèle ouvert pour les outils avec une définition standard, ainsi que le développement Java itératif rapide via Quarkus.
- ▶ La prise en charge serverless de **Knative** Serving et Eventing permet aux équipes de développement de créer des applications serverless et orientées événements qui incluent **Strimzi** (pour exécuter Apache Kafka sur OpenShift) et **Service Mesh**.
- ▶ La prise en charge de **Tekton** dans les pipelines Red Hat OpenShift, des plug-ins OpenShift pour les actions GitHub, de Jenkins et de GitLab Runner offre les capacités d'intégration continue.



Quelle que soit la manière dont vous développez ou déployez vos conteneurs, Red Hat et AWS offrent des solutions pour accélérer le processus et vous permettre de créer et de déployer plus rapidement les applications dont vous avez besoin.

Découvrez comment le [partenariat entre Red Hat et AWS](#) vous aide à [moderniser l'infrastructure avec Red Hat Enterprise Linux](#) ou apprenez-en plus sur :

- ▶ [Red Hat Enterprise Linux \(documentation\)](#)
- ▶ [Podman](#)
- ▶ [Outils de développement sur AWS](#)



## À propos de Red Hat

Premier éditeur mondial de solutions Open Source, Red Hat s'appuie sur une approche communautaire pour fournir des technologies Linux, de cloud hybride, de conteneurs et Kubernetes fiables et performantes. Red Hat aide ses clients à développer des applications cloud-native, à intégrer des applications nouvelles et existantes ainsi qu'à gérer et à automatiser des environnements complexes. [Conseiller de confiance auprès des entreprises du Fortune 500](#), Red Hat propose des services d'assistance, de formation et de consulting reconnus qui apportent à tout secteur les avantages de l'innovation ouverte. Situé au cœur d'un réseau mondial d'entreprises, de partenaires et de communautés, Red Hat participe à la croissance et à la transformation des entreprises et les aide à se préparer à un avenir toujours plus numérique.

**f** [facebook.com/redhatinc](https://facebook.com/redhatinc)  
**t** [@RedHatFrance](https://twitter.com/RedHatFrance)  
**in** [linkedin.com/company/red-hat](https://linkedin.com/company/red-hat)

[fr.redhat.com](https://fr.redhat.com)  
#F30136\_1021

**Europe, Moyen-Orient  
et Afrique (EMEA)**  
00800 7334 2835  
[europe@redhat.com](mailto:europe@redhat.com)

**France**  
00 33 1 41 91 23 23  
[fr.redhat.com](https://fr.redhat.com)

© 2021 Red Hat, Inc. Red Hat, le logo Red Hat et OpenShift sont des marques ou marques déposées de Red Hat, Inc. ou de ses filiales aux États-Unis et dans d'autres pays. Linux® est la marque déposée de Linus Torvalds aux États-Unis et dans d'autres pays. Toutes les autres marques appartiennent à leurs propriétaires respectifs. Java, ainsi que toutes les marques et tous les logos basés sur Java, sont des marques ou marques déposées d'Oracle America, Inc. aux États-Unis et dans d'autres pays. Si l'espace manque, remplacer par « Java est une marque d'Oracle America, Inc. »