6 meilleures pratiques DevSecOps pour le développement

1 Réduisez les risques de dépendance des applications

Il est possible que les éléments logiciels composant vos applications présentent des vulnérabilités. Les outils SCA (Software Composition Analysis) peuvent vous aider à réduire les risques dans la chaîne d'approvisionnement des logiciels, notamment si vous utilisez des composants Open Source. Les outils SCA doivent permettre de réaliser les opérations suivantes :

- Analyser les dépendances des applications afin de garantir l'absence de vulnérabilités connues
- Automatiser la mise en conformité des licences logicielles en identifiant les composants et leurs licences, et en signalant les problèmes de compatibilité
- Confirmer que les dépendances des applications sont toujours valables et proviennent d'une communauté active qui publie encore des mises à jour
- Automatiser l'utilisation des outils SCA dans le processus de création d'applications et l'environnement de développement, de manière à résoudre les éventuels problèmes avant leur intégration et limiter ainsi les cas d'échec lors de la création des applications.

2 Unifiez le code et la gestion des configurations

Les pratiques du modèle GitOps, très courant dans les environnements Kubernetes et conteneurisés, peuvent améliorer considérablement votre posture de sécurité dès le développement :

- Respectez les meilleures pratiques de développement pour la gestion du code source (SCM) lors de la configuration. En utilisant les mêmes commandes pour la vérification, la fusion et l'approbation du code, il est possible de tracer l'auteur et la date d'un changement de configuration de l'infrastructure.
- Pensez à la configuration dès le début du processus et planifiez l'environnement de production de l'application, au lieu de vous en remettre à l'équipe d'exploitation. Le recours à un environnement et à des contrôles de sécurité similaires pour le développement, les tests et la production facilite la gestion des configurations pendant tout le cycle de vie.
- Utilisez un pipeline automatisé afin de créer des images de conteneurs et des artéfacts de fichiers binaires pour l'intégration et la distribution continues (CI/CD). Le déploiement de ces images en production ne devrait pas nécessiter de changements ad hoc.
- Ne stockez pas vos données sensibles dans un système SCM. Utilisez plutôt des outils d'analyse pour vous assurer que les configurations et images de conteneurs ne contiennent pas de secrets intégrés.

3 Protégez les secrets des applications

Il est essentiel de gérer les identités et les secrets (mots de passe, jetons, clés, etc.) tout au long du cycle de vie des applications, ainsi que de contrôler l'accès aux systèmes SCM, registres de conteneurs et référentiels de fichiers binaires. Les informations d'identification pour l'accès des applications aux bases de données et services, ainsi que les builds automatisés et les tests, doivent également être protégées. Et attention aux secrets stockés dans des systèmes SCM ou des fichiers de configuration, car ils peuvent être accidentellement divulgués. Pour protéger les secrets des applications :

- Dès le début du cycle de vie, gérez les identités et contrôlez les accès.
- Pensez à utiliser un coffre-fort ou une boîte noire transactionnelle (HSM) pour gérer et protéger les secrets, au repos comme en transit. Les HSM fonctionnent avec du matériel spécialisé et offrent une meilleure protection que les coffres-forts à secrets, souvent disponibles sous la forme d'un logiciel. Quel que soit l'outil retenu, il devra être intégré à l'infrastructure de gestion des identités.

4 Utilisez des images de base fiables

Les images de base des conteneurs sont des distributions Linux® ultraréduites. Les centaines de paquets qui peuvent être préinstallés présentent des risques de vulnérabilités. Pour limiter les risques liés aux images de conteneurs :

- Choisissez des images fiables bénéficiant de mises à jour également fiables, régulières et dûment testées. Analysez les sources des images ainsi que les options d'assistance proposées.
- Utilisez des outils pour détecter toute vulnérabilité connue. Vous devez également analyser les images pour vérifier la sécurité des configurations et l'absence de secrets intégrés.
- Réduisez le nombre de vecteurs d'attaque en supprimant les fichiers binaires inutiles et vulnérables, comme les outils du système d'exploitation.

5 Gérez les problèmes de conformité et d'audit dès le départ

Pour raccourcir les délais de mise en production, il faut bien comprendre les règles de conformité et les contrôles techniques qui s'appliquent au début du développement. Des contrôles automatisés peuvent être ajoutés au pipeline de création afin d'assurer le respect des exigences de conformité et de sécurité.

Commencez à documenter les procédures et les règles bien en amont, car cette tâche peut représenter jusqu'à 50 % d'un audit. La documentation des politiques inclut le contrôle des accès et des modifications, les sauvegardes et la conservation des données. Pour la documentation des procédures, vous devez inclure les contrôles de sécurité tels que les tests de la sécurité des applications et l'analyse SCA.

6 Commencez par renforcer votre plateforme et votre écosystème

Les menaces de sécurité continuent de croître. C'est pourquoi il est crucial d'utiliser une plateforme avec des capacités de sécurité complètes qui offre des solutions intégrées et prises en charge. Red Hat® OpenShift® est une plateforme Kubernetes d'entreprise proposant de nombreuses fonctions pour le développement et l'exploitation. Extrêmement puissants, les pipelines de création et de déploiement de Red Hat OpenShift sont parfaitement adaptés à la mise en œuvre de contrôles et vérifications de sécurité automatisés à n'importe quelle étape du processus, de l'intégration du code source dans les images au déploiement en production.

Nous avons formé un écosystème de partenaires pour la sécurité afin d'étendre et d'améliorer les fonctionnalités de sécurité de Red Hat OpenShift. Ensemble, nous proposons des solutions prises en charge totalement compatibles avec notre plateforme. Vous avez donc le choix parmi un large éventail de solutions en fonction de vos exigences de sécurité et des besoins de l'entreprise.

L'environnement de développement Red Hat CodeReady Workspaces, natif pour Kubernetes et exécuté sur Red Hat OpenShift, permet d'accélérer le développement des applications basées sur les conteneurs. Enfin, les images Red Hat Universal Base Image et la gamme Red Hat Runtimes fournissent une base solide issue d'une source fiable pour vos applications.

Structure Red Hat DevSecOps

Obtenez une vision globale du cycle de vie de la sécurité, et découvrez comment les fonctions de sécurité pour le développement et la structure Red Hat DevSecOps interagissent. Rendez-vous sur red.ht/DevSecOps.

Découvrez des solutions de sécurité pour le développement

Regardez les webinars de Red Hat et de ses partenaires pour la sécurité pour apprendre à intégrer la sécurité pendant tout le cycle de vie de vos applications.

À propos de Red Hat

Red Hat aide ses clients à standardiser leurs environnements, à développer des applications cloud-native et à intégrer, automatiser, sécuriser et gérer des environnements complexes en offrant des services d'assistance, de formation et de consulting primés.

facebook.com/redhatinc
@RedHatFrance
linkedin.com/company/red-hat

Europe, Moyen-Orient et Afrique (EMEA)
00800 7334 2835
europe@redhat.com

France
00 33 1 41 91 23 23
fr.redhat.com

fr.redhat.com
#F29682_0921