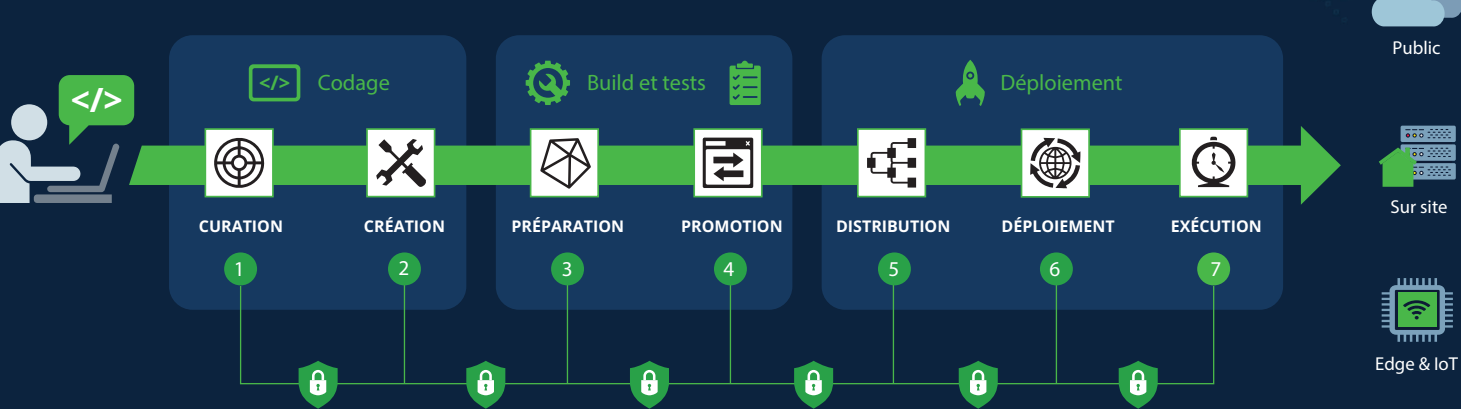


Comment JFrog sécurise la chaîne d'approvisionnement logicielle

Nous protégeons vos logiciels d'application contre les menaces connues et inconnues, ce qui vous donne l'assurance de livrer des versions fiables rapidement et à grande échelle.

Les étapes du développement d'un logiciel

La chaîne d'approvisionnement logicielle peut être décomposée en 3 phases et 7 étapes.



Codage

1 Curation

- Activité** : Vérification de la qualité, de la fiabilité, du respect des normes et des exigences de sécurité des packages tiers et du code interne.
- Menace** : Vulnérabilités des logiciels open source, pratiques de codage non sécurisées, packages malveillants, découverte de CVE, violations de licences.
- Réponse** : Élaborer des politiques et automatiser les contrôles de qualité pour le code et les packages OSS dès le début.

2 Création

- Activité** : Concevoir et développer le logiciel, y compris la rédaction du code, la création d'algorithmes et la mise en œuvre de fonctionnalités.
- Menace** : Vulnérabilités du code interne.
- Réponse** : Rechercher les CWE connues dans les artefacts logiciels à l'aide de méthodes statiques et dynamiques afin de vérifier le code interne pour détecter les erreurs de configuration et la divulgation de secrets.

Build et de tests

3 Préparation

- Activité** : Le logiciel est présenté sous une forme distribuable avec les packages tiers, les dépendances et la documentation nécessaires.
- Menace** : Non-respect de la conformité, fuites de secrets, erreurs de configuration.
- Réponse** : Instaurer des analyses continues de la sécurité et de la conformité dans l'ensemble des fichiers binaires et des dépendances, fournir une analyse contextuelle pour la priorisation des vulnérabilités.

4 Promotion

- Activité** : Promouvoir les builds dans des environnements de staging ou de test afin de garantir leur fonctionnalité, leur compatibilité et leur performance.
- Menace** : Exposition des informations d'authentification et des secrets.
- Réponse** : Analyser les conteneurs, les packages et les artefacts associés.

Déploiement

5 Distribution

- Activité** : Le logiciel testé est prêt à être distribué aux utilisateurs finaux, à l'aide d'outils CI/CD pour les mises à jour continues et le déploiement automatisé.
- Menace** : Corruption de builds, attaques de type « man-in-the-middle », dépôts compromis, mises à jour contrefaites.
- Réponse** : Mise en œuvre de la signature des builds avec vérification par l'utilisateur final, maintien de dépôts sécurisés et bien surveillés avec des mécanismes robustes de contrôle d'accès et d'authentification.

6 Déploiement

- Activité** : Le logiciel est installé et configuré dans l'environnement de l'utilisateur final avec toutes les ressources et dépendances nécessaires.
- Menace** : Erreurs de configuration de l'infrastructure.
- Réponse** : Recherche des erreurs de configuration de l'infrastructure.

7 Exécution

- Activité** : Exploitation du logiciel avec toutes ses fonctionnalités dans l'environnement d'exécution de l'utilisateur final.
- Menace** : Attaques ciblées, vulnérabilités exploitables.
- Réponse** : Analyse des anomalies d'exécution pour détecter les attaques ciblées, priorisation des CVE avec une traçabilité complète de l'ensemble des artefacts et dépendances pour permettre des temps de réponse rapides et une remédiation intelligente.

Sécurisez votre chaîne d'approvisionnement logicielle

Profitez des avantages d'une solution complète basée sur une plateforme :



Sécurité renforcée et risques réduits



Protection contre les menaces connues et inconnues



Conformité et reporting rationalisés



Flexibilité et évolutivité avec moins de ressources

Voyez par vous-même comment JFrog assure votre protection à chaque étape du développement.

Visite guidée

Réserver une démo

Démarrer un essai



Copyright 2024 JFrog Ltd. | JFrog.com