

Gestionnaire de paquets



Table des matières

01

Définition

02

Fonctionnement

03

Types

04

**Génie
logiciel**

05

Limites

06

Conclusion





01

Définition

C'est quoi?

Un outil qui automatise :

- Installation
- Mise à jour
- Configuration
- Suppression

Source

Pourquoi l'utiliser?

Simplicité

- **Installer/mettre à jour** en une seule commande.
- **Dépendances et versions** gérées sans effort.



Sécurité & fiabilité

- **Téléchargement** depuis des dépôts officiels vérifiés.
- **Vérification** des signatures et de l'intégrité des paquets.

Gestion automatique

- **Télécharge** toutes les librairies nécessaires.
- **Résout les conflits** de versions et **maintient** la cohérence de l'environnement.

Reproductibilité

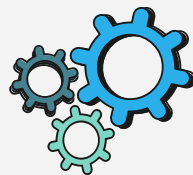
- **Sauvegarder** la liste des dépendances dans un fichier de config.
- **Recréer** le même environnement à partir de ce fichier.



02

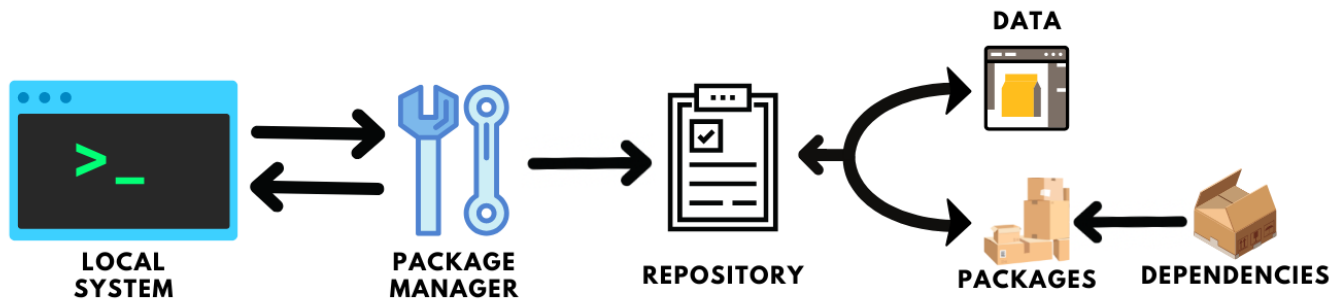
Fonctionnement

Processus



1. **Utilisateur** : Lance une commande pour installer un paquet.
2. **Gestionnaire de paquets** : Analyse la demande, résout les dépendances et vérifie la compatibilité.
3. **Dépôt** : Contient les paquets et leurs métadonnées (nom, version...), fournit le logiciel et ses dépendances (il peut être officiel, communautaire, privé).
4. **Téléchargement** : Récupère les fichiers, vérifie l'intégrité et la signature des paquets, et prépare les paquets pour l'installation.
5. **Installation** : Installe les fichiers aux bons endroits et configure automatiquement.

Processus



Source



03

Types

Types



Systèmes d'exploitation	<u>apt</u> , <u>pacman</u> , <u>brew</u>
Langages de programmation	<u>pip</u> , <u>npm</u> , <u>maven</u>
Général / multi-environnements	<u>conda</u> , <u>spack</u> , <u>nix</u>
App Stores	Mac App Store, Microsoft Store, Google play

Source



04

Génie logiciel

Intégration dans le CI/CD



- Automatisation des builds
- Cohérence entre les environnements
- **Déploiement reproductible** : Dans les images Docker
- Gestion centralisée des configurations
- **Support des dépôts d'artefacts** : Artifactory

Best Practices For Managing Docker Dependencies



05

Limites

Limites



Dependency Hell

Conflits entre les versions et arbres de dépendances

Vulnérabilités

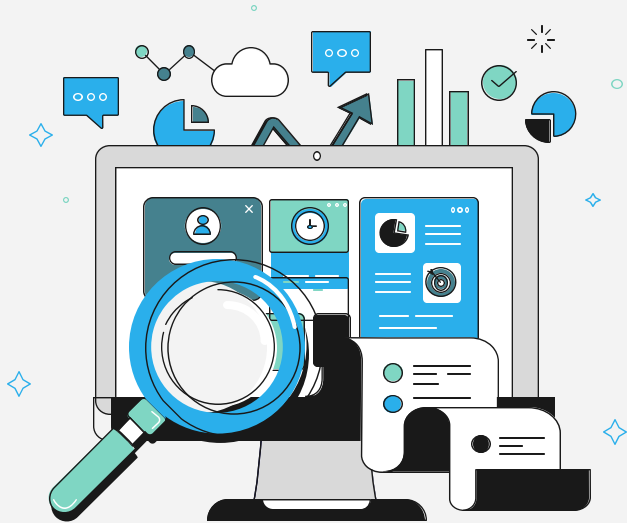
Une librairie tierce vulnérable peut affecter le projet

Dépendance au Dépôt

Indisponibilité ou paquets retirés

Taille des builds

Accumulations des paquets et dépendances



06

Conclusion

Conclusion

Les gestionnaires de paquets simplifient l'installation et la gestion des dépendances, mais présentent un coût : risques de vulnérabilités et conflits de versions.

Faut-il faire confiance aveuglément aux gestionnaires de paquets, ou garder le contrôle sur ses dépendances ?