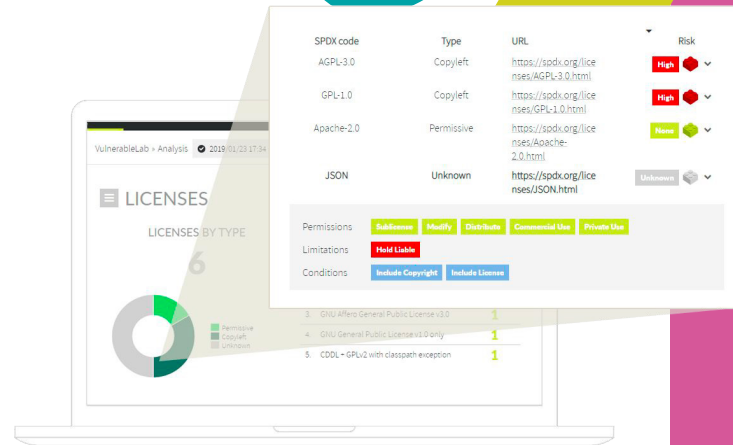


INSIGHTS (SCA)

Presque tous les développeurs s'appuient dans une certaine mesure sur des logiciels open source, et il est difficile de battre la flexibilité de l'utilisation ouverte et des licences de distribution. Cependant, il est également essentiel que tous les développeurs comprennent comment contrôler les composants open source. Il existe un certain nombre de problèmes de dépendance, de sécurité et de compatibilité de licence liés aux solutions open source qui nécessitent une réflexion avant de lancer une application.



Il est extrêmement important de **générer un inventaire complet et détaillé** des composants open-source et tiers utilisé pendant les builds ou dans les applications.

Vos composants open source font autant partie de votre application que le code que vous avez construit vous-même.

Le rôle, les fonctions, les fonctionnalités, le comportement et la licence de chaque composant open source doivent être parfaitement compris afin de gérer correctement le déploiement de votre application et de vous conformer aux réglementations sur les licences logicielles.

Éliminez le processus fastidieux et sujet aux erreurs de compilation manuelle de l'inventaire - y compris leurs dépendances - dans le but de déterminer si vous êtes touché par une nouvelle alerte de vulnérabilité de sécurité ou pour vérifier les problèmes de licence. Rédigez une politique officielle qui couvrira tous les développeurs de vos équipes.

KIUWAN INSIGHTS SE DISTINGUE PAR

- ✓ Pas de configuration particulière
- ✓ Entièrement personnalisable – visuellement et conceptuellement
- ✓ Demandez une démonstration gratuite ou un devis et découvrez comment Kiuwan Insights peut vous aider à rationaliser les composants externes de votre application

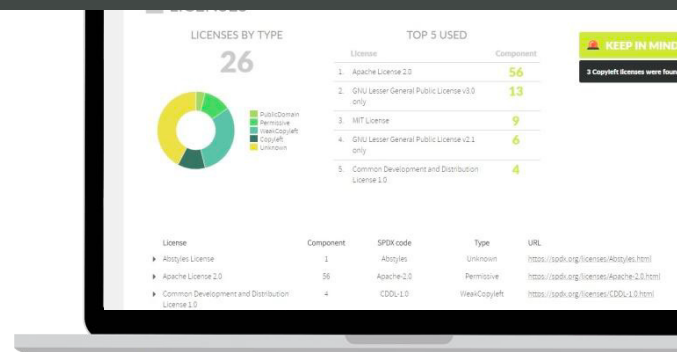
QUELQUES VULNERABILITIES DETECTEES

- Application Misconfiguration
- Credential/Session Prediction
- Directory Indexing
- Insufficient Authorization/Authentication
- Automatic Reference Counting
- Cross Site Request Forgery
- Information Leakage
- Insufficient Transport Layer Protection
- Insufficient Binary Protection
- Cross Site Scripting
- Injection Attacks
- Interprocess Communication
- OS Commanding
- Insecure Cryptography
- Cryptographic Related Attacks
- Buffer Overrun
- Free Non-Heap Variable
- Use After-Free
- Double Free/Close
- Format String Vulnerability
- Return Pointer To Local
- SQL injection
- ... and more

Kiuwan Insights scanne en permanence les **NIST National Vulnerability Database** pour les nouvelles vulnérabilités, en plus d'utiliser notre propre base de connaissances et les recherches par des experts en sécurité.

DEPENDENCES

Les projets open source sont souvent accompagnés de dépendances spécifiques par version. Au fur et à mesure de la mise à jour des environnements de déploiement, les composants open source de la base de code peuvent être affectés par les dépendances des versions précédentes. Étant donné que de nombreux développeurs s'appuient sur une pile importante de composants open source, il peut être difficile d'isoler la source des dépendances. Kiuwan Insights vous aide à suivre l'architecture des applications et à surveiller la qualité du code, aidant ainsi les développeurs à réduire le nombre de dépendances dans leur base de code. Cela n'élimine pas complètement les dépendances, mais le maintien de ce niveau de compréhension du code rend les développeurs beaucoup plus efficaces pour résoudre les problèmes de dépendance à mesure qu'ils surviennent.



LICENSE COMPLIANCE

La plupart des projets open source permettent pleinement aux utilisateurs de répliquer, modifier, distribuer et même vendre leurs logiciels à des fins lucratives.

Ce n'est pas le cas de tous les projets open source, c'est pourquoi les développeurs ont parfois des problèmes en utilisant des composants open source dans leurs produits. La plupart des produits nécessitent un verbiage spécifique dans les licences des distributions de logiciels dérivés pour créditer correctement l'utilisation des bibliothèques open source.

LANGAGES

REFERENTIELS

BUILD SYSTEMS

Go	GitHub	go.mod Gopkg.lock
Java	Maven Gradle	nt (*.xml files) Maven (pom.xml files) Gradle (*.gradle files) *.jar, *.war, *.ear files
Javascript	Npm Bower	Npm (package.json files) Bower (bower.json files) Yarn (package.json files)
.Net	Nuget	Nuget (*.csproj, project.json, global.json, *.vbproj files)
Python	PyPI GitHub	PyPI (setup.py files) Requirements (txt file with declared dependencies)
Scala	Maven	SBT (build.sbt)
Swift	Cocoapods GitHub	Podspec (*.podspec, Podfile.lock files) Package (Package.swift files)
Php	Packagist	Composer (composer.json, composer.lock files)
Ruby	RubyGems	Gemfile, Gemfile.lock and *.gemspec files

EN RESUME

✓ Détecter les menaces

Enquêtez sur les risques de sécurité liés à vos composants open source et traitez chacun d'eux.

✓ Inventaire des composants

Générez un inventaire complet et précis de tous les composants open source et tiers utilisés lors des builds ou dans les applications.

✓ Éliminer les activités chronophages

Éliminez le processus fastidieux et sujet aux erreurs de compilation manuelle de l'inventaire dans le but de déterminer si vous êtes touché par une nouvelle alerte de vulnérabilité de sécurité ou pour vérifier les problèmes de licence.

✓ Eviter l'obsolescence

Gérez l'obsolescence de vos bibliothèques : mises à jour, versions et problèmes de sécurité. Recevez des alertes d'obsolescence.

✓ Détecter les risques de sécurité

Étudiez les risques de sécurité liés à vos composants open source et traitez chacun d'eux en fonction de votre application.

✓ Isoler les dépendances

Les déploiements open source incluent souvent de nombreuses fonctionnalités inutilisées qui causent des problèmes de dépendance. L'analyse de la qualité du code Kiuwan permet d'identifier le code inutilisé et de le supprimer, ce qui réduit encore le risque de rencontrer des problèmes de dépendance.

