Docker static code analysis

Unique rules to find Vulnerabilities, Security Hotspots, and Code Smells in your DOCKER code

Secrets
ABAP
Apex
AzureResourceManager
C
C#
C++
CloudFormation
COBOL
CSS
Dart
Docker
Flex
Go
HTML
Java
JavaScript
JCL
Kotlin
Kubernetes
Objective C
PHP
PL/I
PL/SQL
Python
RPG
Ruby
Scala
Swift
Terraform
Text
TypeScript
T-SQL
VB.NET
VB6
XML

All rules 44 | 🔒 Vulnerability 4 | 🐛 Bug 4 | 🛡 Security Hotspot 15 | ◎ Code Smell 21

Tags ⌄ | Impact ⌄ | Clean code attribute ⌄ | Search by name...

Cache should be cleaned after package installation
◎ Code Smell

Deprecated instructions should not be used
◎ Code Smell

Consent flag should be set to avoid manual input
◎ Code Smell

Environment variables should not be unset on a different layer than they were set
◎ Code Smell

Expanded filenames should not become options
◎ Code Smell

Double quote to prevent globbing and word splitting
◎ Code Smell

Instructions should be upper case
◎ Code Smell

Allowing non-root users to modify resources copied to an image is security-sensitive
🛡 Security Hotspot

Automatically installing recommended packages is security-sensitive
🛡 Security Hotspot

Running containers as a privileged user is security-sensitive
🛡 Security Hotspot

Delivering code in production with debug features activated is security-sensitive
🛡 Security Hotspot

Cache should be cleaned after package installation

[Intentionality - Efficient] [Maintainability ⌃]

◎ Code Smell | 🔻 Major ⓘ

In Docker, when packages are installed via a package manager, an index is cached locally by default. This index should either be cleaned up or stored in a dedicated cache mount.

Why is this an issue? | How can I fix it? | **More Info**

Analyze your code

Documentation

- Dockefile Best Practices - RUN - Best practices for writing Dockerfiles
- apk man
- apt-get man
- aptitude man
- Ask Ubuntu - How do I remove the apt package index?
- Docker Build Cache - Use the dedicated RUN cache

Available In:

sonarlint ••• | sonarcloud ☁ | sonarqube 🔊

Sonar helps developers write Clean Code.
Privacy Policy | Cookie Policy