- 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

Docker static code analysis

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

All rules 44 | Vulnerability 4 | Bug 4 | Security Hotspot 15 | Code Smell 21

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

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

Use ADD instruction to retrieve remote resources

Code Smell

Arguments in long RUN instructions should be sorted

Code Smell

Track uses of "TODO" tags

Consent flag should be set to avoid manual input

Analyze your code

Intentionality - Logical | Maintainability

Code Smell | Major | shell

Various package and software management applications require manual input for execution confirmation by default. This confirmation is usually required when installing, updating, or removing programs and packages. General consent can be given to execute the command without manual input.

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

Code examples

Noncompliant code example

```
RUN apt-get install ca-certificates
RUN aptitude install ca-certificates
RUN apt install ca-certificates
```

Here each line represents a package installation command command for the most popular package managers. Each of them is trying to perform an installation in interactive mode, it will wait for prompt that will never come, so it will result in aborted execution.

Compliant solution

```
RUN apt-get -y install ca-certificates
RUN aptitude -y install ca-certificates
RUN apt -y install ca-certificates
```

Here in each line we added the option `-y`, it will assume yes to all prompts and continue execution.

How does this work?

If the `-y` flag is set, no manual input is expected, and the package manager can run non-interactively. For `apt` and `apt-get`, the long versions `--yes` and `--assume-yes` also exist. For `aptitude`, the long version `--assume-yes` exists.

Available In:

sonarlint | sonarcloud | sonarqube