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...

Bug

Dockerfile should only have one ENTRYPOINT and CMD instruction

Bug

Access variable which is not available in the current scope

Bug

A space before the equal sign in key-value pair may lead to unintended behavior

Bug

Allowing downgrades to a clear-text protocol is security-sensitive

Security Hotspot

Allowing shell scripts execution during package installation is security-sensitive

Security Hotspot

Using host operating system namespaces is security-sensitive

Security Hotspot

Setting loose POSIX file permissions is security-sensitive

Security Hotspot

Reduce the amount of consecutive RUN instructions

Code Smell

Prefer COPY over ADD for copying local resources

Code Smell

WORKDIR instruction should only be used with absolute path

Code Smell

Too long RUN instruction should be split into multiple lines

Malformed JSON in Exec form leads to unexpected behavior

Analyze your code

Consistency - Conventional | Maintainability | Reliability

Bug | Major

In Dockerfiles, commands can be specified either as a single string or as a JSON array of strings. The latter is called an "exec form". However, when exec form is not a valid JSON, it will be silently treated as shell form. Usually, this will lead to a crash, but sometimes it can pass silently and lead to unexpected behavior.

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

Code examples

Noncompliant code example

```
FROM alpine:3.14
ENTRYPOINT ["/app", "-c", "config.json"] -D
```

```
FROM alpine:3.14
ENTRYPOINT [ "/bin/bash”, “-c” ]
```

Compliant solution

```
FROM alpine:3.14
ENTRYPOINT ["/app", "-c", "config.json", "-D"]
```

```
FROM alpine:3.14
ENTRYPOINT [ "/bin/bash", "-c" ]
```

Available In:

sonarlint | sonarcloud | sonarqube