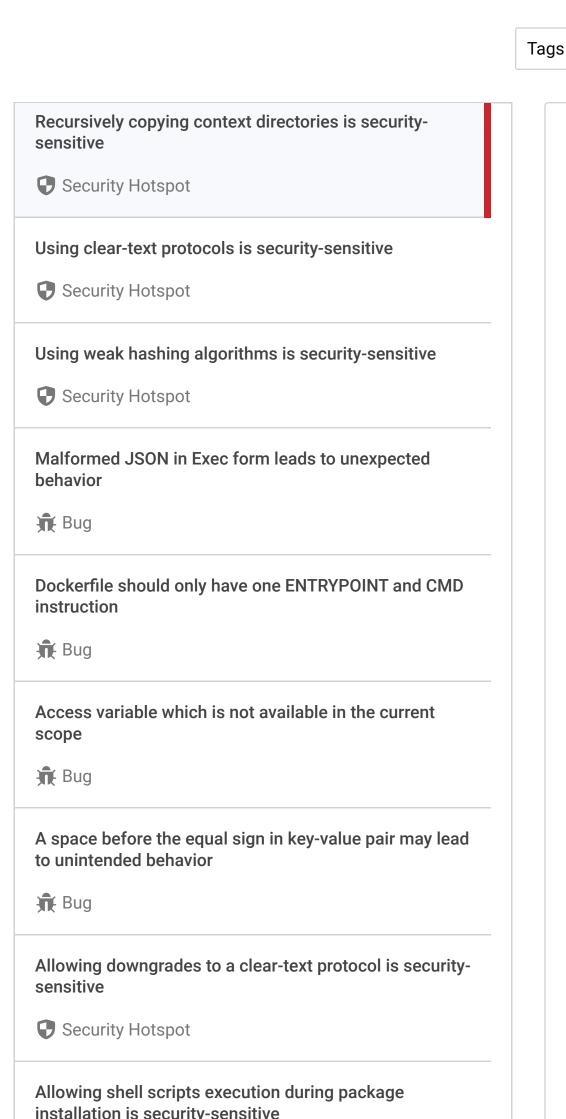


Docker static code analysis

Unique rules to find Vulnerabilities, Security Hotspots,

Security Hotspot (15) Code Smell (21)



Recursively copying context directories is securitysensitive

Analyze your code

Search by name...

Q

• dockerfile cwe When building a Docker image from a Dockerfile, a context directory is used and sent to the Docker daemon before the actual build starts. This context directory usually contains the Dockerfile itself, along with all the files that will be necessary

Clean code attribute

- the source code of applications to set up in the container.
- · configuration files for other software components.

Impact

• other necessary packages or components.

for the build to succeed. This generally includes:

Intentionality - Complete Security (2)

The COPY and ADD directives in the Dockerfiles are then used to actually copy content from the context directory to the image file system.

When COPY or ADD are used to recursively copy entire top-level directories or multiple items whose names are determined at build-time, unexpected files might get copied to the image filesystem. It could affect their confidentiality.

Ask Yourself Whether

- The copied files and directories might contain sensitive data that should be kept confidential.
- The context directory contains files and directories that have no functional purpose for the final container image.

There is a risk if you answered yes to any of those questions.

Keep in mind that the content of the context directory might change depending on the build environment and over time.

Recommended Secure Coding Practices

- Limit the usage of globbing in the COPY and ADD sources definition.
- · Avoid copying the entire context directory to the image filesystem.
- Prefer providing an explicit list of files and directories that are required for the image to properly run.

Sensitive Code Example

Copying the complete context directory:

```
FROM ubuntu:22.04
# Sensitive
COPY . .
CMD /run.sh
```

Copying multiple files and directories whose names are expanded at build time:

```
FROM ubuntu:22.04
# Sensitive
COPY ./example* /
COPY ./run.sh /
CMD /run.sh
```

Compliant Solution

```
FROM ubuntu:22.04
COPY ./example1 /example1
COPY ./example2 /example2
COPY ./run.sh /
CMD /run.sh
```

See

- Dockerfile reference COPY directive
- Dockerfile reference ADD directive
- CWE CWE-668 Exposure of Resource to Wrong Sphere
- CWE CWE-497 Exposure of Sensitive System Information to an Unauthorized Control Sphere

Available In:

sonarlint ⊕ | sonarcloud ↔ | sonarqube

