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...

Server certificates should be verified during SSL/TLS connections
Vulnerability

Weak SSL/TLS protocols should not be used
Vulnerability

Disabling builder sandboxes is security-sensitive
Security Hotspot

Exposing administration services is security-sensitive
Security Hotspot

Recursively copying context directories is security-sensitive
Security Hotspot

Using clear-text protocols is security-sensitive
Security Hotspot

Using weak hashing algorithms is security-sensitive
Security Hotspot

Malformed JSON in Exec form leads to unexpected behavior
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

Server certificates should be verified during SSL/TLS connections

Analyze your code

Responsibility - Trustworthy | Security

Vulnerability | Critical | cwe privacy ssl

This vulnerability makes it possible that an encrypted communication is intercepted.

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

Transport Layer Security (TLS) provides secure communication between systems over the internet by encrypting the data sent between them. Certificate validation adds an extra layer of trust and security to this process to ensure that a system is indeed the one it claims to be.

When certificate validation is disabled, the client skips a critical security check. This creates an opportunity for attackers to pose as a trusted entity and intercept, manipulate, or steal the data being transmitted.

What is the potential impact?

Establishing trust in a secure way is a non-trivial task. When you disable certificate validation, you are removing a key mechanism designed to build this trust in internet communication, opening your system up to a number of potential threats.

Identity spoofing

If a system does not validate certificates, it cannot confirm the identity of the other party involved in the communication. An attacker can exploit this by creating a fake server and masquerading as a legitimate one. For example, they might set up a server that looks like your bank's server, tricking your system into thinking it is communicating with the bank. This scenario, called identity spoofing, allows the attacker to collect any data your system sends to them, potentially leading to significant data breaches.

Loss of data integrity

When TLS certificate validation is disabled, the integrity of the data you send and receive cannot be guaranteed. An attacker could modify the data in transit, and you would have no way of knowing. This could range from subtle manipulations of the data you receive to the injection of malicious code or malware into your system. The consequences of such breaches of data integrity can be severe, depending on the nature of the data and the system.

Available In:
sonarlint | sonarcloud | sonarqube