Java static code analysis

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

"wait(...)" should be used instead of "Thread.sleep(...)" when a lock is held

Analyze your code

Bug Blocker multi-threading performance cert

If `Thread.sleep(...)` is called when the current thread holds a lock, it could lead to performance and scalability issues, or even worse to deadlocks because the execution of the thread holding the lock is frozen. It's better to call `wait(...)` on the monitor object to temporarily release the lock and allow other threads to run.

Noncompliant Code Example

```
public void doSomething(){
  synchronized(monitor) {
    while(notReady()){
      Thread.sleep(200);
    }
    process();
  }
  ...
}
```

Compliant Solution

```
public void doSomething(){
  synchronized(monitor) {
    while(notReady()){
      monitor.wait(200);
    }
    process();
  }
  ...
}
```

See

- CERT, LCK09-J. - Do not perform operations that can block while holding a lock

Available In:
sonarlint | sonarcloud | sonarqube

Server hostnames should be verified during SSL/TLS connections

🔓 Vulnerability

Insecure temporary file creation methods should not be used

🔓 Vulnerability

Passwords should not be stored in plain-text or with a fast hashing algorithm

🔓 Vulnerability

Server certificates should be verified during SSL/TLS connections

🔓 Vulnerability