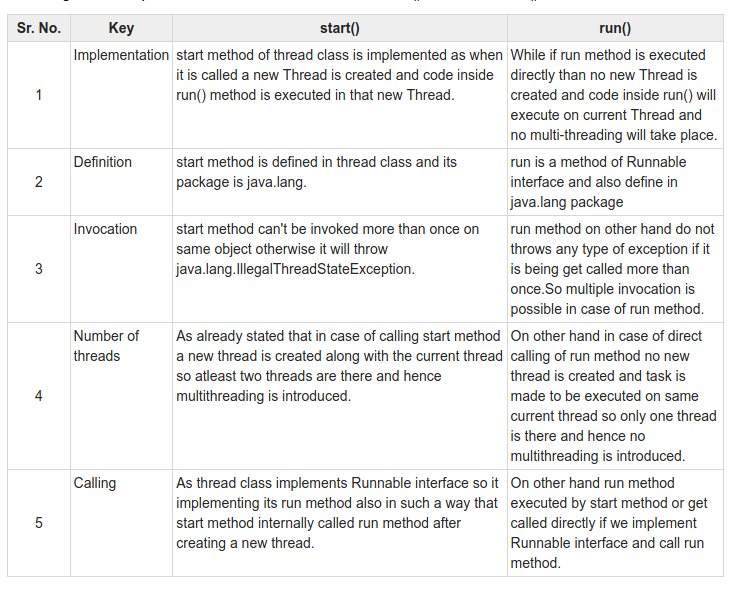
**Code Correctness: Call to Thread.run() Development Mitigation SOP**

Code correctness vulnerabilities occur when an Object API is not used properly or as intended. Code correctness vulnerabilities caused by Thread.run() are detected by Fortify because the run() method is used rather than start(). Typically the start() method is used to begin a new thread of control, but programmers often accidentally call run(),so the run()method will execute in the caller’s thread of control.

When a program calls the *start()* method, a new thread is created and then the *run()* method is executed. But if we directly call the *run()* method then no new thread will be created and *run()* method will be executed as a normal method call on the current calling thread itself and no multi-threading will take place. In Java’s multi-threading concept, another most important difference between *start()* and *run()* method is that we can’t call the *start()* method twice otherwise it will throw an *IllegalStateException* whereas *run()* method can be called multiple times as it is just a normal method calling.



**Defense Against Code Correctness: Call to Thread.run()**

In most cases, the call to the run() method can be replaced with the start() method.

**Examples**

**General Example**

Thread thr = new Thread() {

public void run() {

…

}

};

thr.run();

**Explanation**

The code above mistakenly calls run() instead of start()*.*

**Recommendation**

The code can be rewritten in the following way:

Thread thr = new Thread() {

public void run() {

…

}

};

thr.start();

**Example**

collectionManager = new CollectionManagerThread(parser);

SwingUtilities.invokeLater(new Runnable() {

public void run() {

collectionManager.run();

refreshTree();

}

});

**Explanation**

The code above mistakenly calls run() instead of start()*.*

**Recommendation**

collectionManager = new CollectionManagerThread(parser);

SwingUtilities.invokeLater(new Runnable() {

public void run() {

collectionManager.start();

refreshTree();

}

});

**Resources**

1. [HP Enterprise Security – Code Correctness: Call to Thread.run()](https://vulncat.fortify.com/en/detail?id=desc.structural.java.code_correctness_call_to_thread_run#Java%2fJSP)
2. <https://www.geeksforgeeks.org/difference-between-thread-start-and-thread-run-in-java/>