FB.DM\_FP\_NUMBER\_CTOR

Использование new Double(double) гарантированно всегда приводит к созданию нового объекта, тогда как Double.valueOf(double) позволяет кэшировать значения, выполняемые компилятором, библиотекой классов или JVM. Использование кэшированных значений позволяет избежать выделения объектов, и код будет работать быстрее.

Если класс не должен быть совместим с JVM, предшествующими Java 5, следует использовать либо автоупаковку, либо метод valueOf() при создании экземпляров Double и Float.

Таблица №1 – Основные характеристики детектора FB.DM\_FP\_NUMBER\_CTOR

|  |  |
| --- | --- |
| Категория критичности | Minor |
| Надежность | Unknown |
| Источник | SpotBugs |
| Тип ошибки | Performance Issue |
| CWE | --- |
| CVE | --- |
| Возможные последствия | Избыточное создание объектов-обёрток; снижение производительности в циклах; увеличение нагрузки на сборщик мусора |

Таблица №2 – Примеры срабатывания детектора и устранения уязвимости

|  |  |
| --- | --- |
| Потенциально небезопасное | Корректная конструкция |
| public static void incorrectTest() {  // Неэффективное создание объектов-обёрток через конструктор  Integer i = new Integer(42);  Double d = new Double(3.14);  Long l = new Long(100L);    System.out.println(i + ", " + d + ", " + l);  } | public static void correctTest() {  // Использование valueOf() или автоупаковки  Integer i = Integer.valueOf(42);  Double d = 3.14; // автоупаковка  Long l = 100L; // автоупаковка    System.out.println(i + ", " + d + ", " + l);  } |