FB.SA\_LOCAL\_SELF\_COMPUTATION

Детектор определяет ошибку, в ходе которой метод выполняет бессмысленное вычисление локальной переменной с другой ссылкой на ту же переменную (например, x&x или x-x). Из-за характера вычислений эта операция кажется бессмысленной и может указывать на опечатку или логическую ошибку. Дважды проверьте расчет.

Таблица №1 – Основные характеристики детектора

|  |  |
| --- | --- |
| Категория критичности | MAJOR |
| Надежность | Unknown |
| Источник | SpotBugs |
| Тип ошибки | Ошибка разработчика |
| CWE | — |
| CVE | — |
| Возможные последствия | Бессмысленные вычисления (например, x - x, x & x, x ^ x) не изменяют значение переменной и могут указывать на логическую ошибку, приводящую к некорректной работе программы. |
| Комментарий | Необходимо проверить формулу вычислений, возможно, была допущена опечатка или использована неверная переменная. |

Таблица №2 – Примеры срабатывания детектора и устранения уязвимости

|  |  |
| --- | --- |
| Некорректное написание | Корректное написание |
| public static void incorrectTest(int value) {  boolean test = (value & value) != 0;  } | public static void correctTest(int value) {  int value1 = 1;  boolean test = (value & value1) != 0;  } |

Методы устранения уязвимости при срабатывании детектора:

1. **Проанализируйте строку кода, содержащую бессмысленное вычисление.** Определите, какую операцию предполагалось выполнить и с какой другой переменной или значением.
2. **Замените одну из ссылок на локальную переменную на ссылку на другую переменную, поле класса, параметр метода или константу, которая должна участвовать в вычислении.**
3. **Проверьте, не является ли бессмысленное вычисление частью более сложного выражения.** Убедитесь, что после исправления вся логика выражения остается корректной.
4. **Если вычисление не имеет никакого эффекта и его результат не используется, удалите эту строку кода.**
5. **Рассмотрите возможность использования отладчика для проверки значений переменных и пошагового выполнения кода, чтобы лучше понять предполагаемую логику вычисления.**
6. **Проверьте, не является ли бессмысленное вычисление результатом ошибки копирования и вставки кода, который не был должным образом отредактирован.**
7. **Убедитесь, что используемый оператор соответствует предполагаемой операции.** Например, если вы хотели выполнить побитовое И с другой переменной, убедитесь, что используется &, а не && или & с той же переменной.