FB.SA\_FIELD\_DOUBLE\_ASSIGNMENT

Детектор определяет ошибку, в ходе которой метод содержит двойное присвоение поля.

Таблица №1 – Основные характеристики детектора

|  |  |
| --- | --- |
| Категория критичности | MAJOR |
| Надежность | Unknown |
| Источник | SpotBugs |
| Тип ошибки | Ошибка разработчика |
| CWE | — |
| CVE | — |
| Возможные последствия | Двойное присваивание значения полю в одном методе может указывать на логическую ошибку или избыточные операции, что может привести к потере данных или неожиданному поведению. |
| Комментарий | Если поле дважды изменяется в одном методе, убедитесь, что оба присваивания необходимы и выполняются в правильном порядке. |

Таблица №2 – Примеры срабатывания детектора и устранения уязвимости

|  |  |
| --- | --- |
| Некорректное написание | Корректное написание |
| private static int value;  public static void incorrectTest(int newValue) {  value = 10;  value = newValue;  System.out.println("Некорректное значение: " + value);  } | private static int value;  public static void correctTest(int newValue) {  if (value != newValue) {  value = newValue;  }  System.out.println("Корректное значение: " + value);  } |

Методы устранения уязвимости при срабатывании детектора:

1. Проанализируйте код метода, в котором обнаружено двойное присваивание поля. Определите, используется ли значение, присвоенное полю в первый раз, до второго присваивания.
2. Удалите первое присваивание, если его значение нигде не используется до того, как поле перезаписывается. Это сделает код более чистым и эффективным.
3. Проверьте, не является ли двойное присваивание результатом ошибки копирования и вставки кода. Убедитесь, что обе операции присваивания предназначены для одного и того же поля и выполняются с нужными значениями.
4. Если двойное присваивание является частью сложной логики обновления состояния объекта, тщательно проверьте порядок операций и убедитесь, что промежуточное значение поля не является важным.