FB.IM\_AVERAGE\_COMPUTATION\_COULD\_OVERFLOW

Детектор определяет ошибку, в ходе которой код вычисляет среднее значение двух целых чисел, используя либо деление, либо сдвиг вправо со знаком, а затем использует результат в качестве индекса массива. Если усредняемые значения очень велики, это может привести к переполнению (что приведет к вычислению отрицательного среднего). Предполагая, что результат должен быть неотрицательным, вместо этого вы можете использовать беззнаковый сдвиг вправо. Другими словами, вместо использования (низкий+высокий)/2 используйте (низкий+высокий) >>> 1.

Таблица №1 – Основные характеристики детектора

|  |  |
| --- | --- |
| Категория критичности | MAJOR |
| Надежность | Unknown |
| Источник | SpotBugs |
| Тип ошибки | Ошибка разработчика |
| CWE | — |
| CVE | — |
| Возможные последствия | При вычислении среднего двух больших целых чисел (например, low + high) может произойти переполнение, что приведет к отрицательному значению индекса массива и потенциальному ArrayIndexOutOfBoundsException. |
| Комментарий | Чтобы избежать переполнения, используйте беззнаковый сдвиг вправо (>>> 1) вместо деления. |

Таблица №2 – Примеры срабатывания детектора и устранения уязвимости

|  |  |
| --- | --- |
| Некорректное написание | Корректное написание |
| public static void incorrectTest() {  int low = Integer.MAX\_VALUE - 1;  int high = Integer.MAX\_VALUE;    // может привести к переполнению  int mid = (low + high) / 2;  System.out.println("Incorrect mid: " + mid);  } | public static void correctTest() {  int low = Integer.MAX\_VALUE - 1;  int high = Integer.MAX\_VALUE;  // предотвращает переполнение  int mid = (low + high) >>> 1;  System.out.println("Correct mid: " + mid);  } |