FB.VA\_PRIMITIVE\_ARRAY\_PASSED\_TO\_OBJECT\_VARARG

Детектор определяет ошибку, в ходе которой код передает примитивный массив функции, которая принимает переменное количество аргументов объекта. Это создает массив длиной один для хранения примитивного массива и передает его функции.

Таблица №1 – Основные характеристики детектора

|  |  |
| --- | --- |
| Категория критичности | MAJOR |
| Надежность | Unknown |
| Источник | SpotBugs |
| Тип ошибки | Ошибка разработчика |
| CWE | — |
| CVE | — |
| Возможные последствия | Передача примитивного массива в функцию с переменным числом аргументов, ожидающую объект, может привести к неожиданным результатам или ошибкам, поскольку примитивный массив будет обёрнут в объект, что может изменить ожидаемое поведение функции. |
| Комментарий | Для корректной передачи примитивного массива следует использовать подходящие способы упаковки, чтобы избежать создания лишнего массива и сохранить правильную работу функции. |

Таблица №2 – Примеры срабатывания детектора и устранения уязвимости

|  |  |
| --- | --- |
| Некорректное написание | Корректное написание |
| public static void incorrectTest() {  int[] numbers = {1, 2, 3};  // программа выведет адрес массив  printObjects(numbers);  }  public static void printObjects(Object... args) {  for (Object arg : args) {  System.out.println(arg);  }  } | public static void correctTest() {  int[] numbers = {1, 2, 3};  printObjects(Arrays.toString(numbers));  }  public static void printObjects(Object... args) {  for (Object arg : args) {  System.out.println(arg);  }  } |

Методы устранения уязвимости при срабатывании детектора:

1. Если функция предназначена для обработки элементов примитивного массива, измените сигнатуру функции, чтобы она принимала vararg-параметр соответствующего примитивного типа. Например, int... args вместо Object... args.
2. В месте вызова функции, если вы хотите передать элементы примитивного массива как отдельные аргументы, разверните массив вручную. Это можно сделать с помощью цикла или других средств языка (если применимо).
3. Используйте методы утилитарных классов, такие как Arrays.toString() (для вывода) или другие методы, которые преобразуют примитивный массив в объект, который ожидает vararg-параметр Object. Это подходит, если функция обрабатывает массив как единый объект.
4. Если функция должна обрабатывать массив как список объектов, вручную преобразуйте примитивный массив в массив оберток (например, Integer[] для int[]) перед передачей.