FB.DM\_STRING\_TOSTRING

Детектор определяет ошибку, в ходе которой вызов String.toString() является избыточной операцией. Просто используйте строку.

Таблица №1 – Основные характеристики детектора

|  |  |
| --- | --- |
| Категория критичности | MAJOR |
| Надежность | Unknown |
| Источник | SpotBugs |
| Тип ошибки | Ошибка разработчика |
| CWE | — |
| CVE | — |
| Возможные последствия | Использование конструктора new String(String) или вызова String.toString() создаёт ненужные объекты, что приводит к избыточному расходу памяти и снижению производительности. |
| Комментарий | Рекомендуется использовать строку напрямую без лишних преобразований, так как строки в Java неизменяемы, и дополнительное создание объектов не имеет смысла. |

Таблица №2 – Примеры срабатывания детектора и устранения уязвимости

|  |  |
| --- | --- |
| Некорректное написание | Корректное написание |
| public static void incorrectTest() {  String message = "Hello, world!";  String duplicate = message.toString();  System.out.println(duplicate);  } | public static void correctTest() {  String message = "Hello, world!";  System.out.println(message);  } |

Методы устранения уязвимости при срабатывании детектора:

1. **Удалите вызов .toString():** Просто удалите вызов метода toString() у любого объекта, который уже имеет тип String.
2. **Проверьте весь код:** Найдите все места, где вызывается toString() у строковых переменных или литералов, и удалите этот вызов.
3. **Избегайте ненужных преобразований:** Понимание того, что объект типа String уже является строкой, помогает писать более чистый и эффективный код.
4. **Сосредоточьтесь на читаемости:** Удаление избыточных вызовов методов делает код более лаконичным и легким для понимания.