NO\_CATCH.STRICT

Ошибка "Top-level function 'Main.main' doesn't catch exceptions of types 'java/lang/Error\*, java/lang/OutOfMemoryError\*', which could be thrown by callees" указывает на то, что в коде отсутствует обработка исключений типа java/lang/Error и java/lang/OutOfMemoryError. Эти ошибки представляют собой серьезные проблемы, которые могут возникнуть в процессе выполнения программы и указывают на фундаментальные проблемы среды выполнения.

Класс java/lang/Error является базовым классом для всех ошибок времени выполнения, которые обычно связаны с проблемами виртуальной машины Java (JVM) или среды выполнения. java/lang/OutOfMemoryError возникает, когда JVM не может выделить достаточное количество памяти для выполнения операции.

Таблица №1 – Основные характеристики детектора NO\_CATCH.STRICT

|  |  |
| --- | --- |
| Категория критичности | Minor |
| Надежность | Unknown |
| Источник | SvEng |
| Тип ошибки | Uncaught Exception |
| CWE | --- |
| CVE | --- |
| Возможные последствия | Аварийное завершение программы, необработанные ошибки, потеря данных, нестабильность работы приложения |

Таблица №2 – Примеры срабатывания детектора и устранения уязвимости

|  |  |
| --- | --- |
| Потенциально небезопасное | Корректная конструкция |
| public static void incorrectTest() throws FileNotFoundException {  FileInputStream fis = new FileInputStream("missing.txt"); // Может выбросить исключение  System.out.println("Файл открыт (но мы этого не увидим при ошибке)");  } | public static void correctTest() {  try {  FileInputStream fis = new FileInputStream("missing.txt");  System.out.println("Файл открыт");  } catch (FileNotFoundException e) {  System.err.println("Ошибка: Файл не найден");  // Логирование или обработка ошибки  } finally {  System.out.println("Блок finally выполнен");  }  } |