FB.PATH\_TRAVERSAL\_OUT

Ошибка "Method writes to a file whose location is specified by user input" указывает на потенциальную уязвимость в коде, связанную с записью в файл, путь к которому указывается пользовательским вводом. Это может стать источником опасности, таких как запись в конфиденциальные файлы, выполнение нежелательных операций или атаки вида "Path Traversal" (перемещение по пути).

Таблица №1 – Основные характеристики детектора FB.PATH\_TRAVERSAL\_OUT

|  |  |
| --- | --- |
| Категория критичности | Minor |
| Надежность | Unknown |
| Источник | SpotBugs |
| Тип ошибки | Path Traversal Vulnerability |
| CWE | CWE22 |
| CVE | --- |
| Возможные последствия | Несанкционированная запись файлов в произвольные каталоги, повреждение системных файлов, компрометация системы |

Таблица №2 – Примеры срабатывания детектора и устранения уязвимости

|  |  |
| --- | --- |
| Потенциально небезопасное | Корректная конструкция |
| public static void incorrectTest(String fileName) {  try {  File file = new File("/var/tmp/export/" + fileName);  Files.writeString(file.toPath(), "malicious content");  System.out.println("File written (VULNERABLE)");  } catch (IOException e) {  e.printStackTrace();  }  } | public static void correctTest(String fileName) {  try {  Path basePath = Paths.get("/var/tmp/export/").normalize().toAbsolutePath();  Path userPath = Paths.get(fileName).normalize();  Path resolvedPath = basePath.resolve(userPath).normalize();    if (!resolvedPath.startsWith(basePath)) {  throw new SecurityException("Path traversal attempt detected");  }    Files.writeString(resolvedPath, "safe content");  System.out.println("File written securely");  } catch (IOException | SecurityException e) {  e.printStackTrace();  }  } |