FB.ODR\_OPEN\_DATABASE\_RESOURCE

Метод создает ресурс базы данных (например, подключение к базе данных или набор строк), не присваивает его каким-либо полям, не передает другим методам и не возвращает его, и, по-видимому, не закрывает объект на всех путях выхода из метода. Неспособность закрыть ресурсы базы данных по всем путям вне метода может привести к снижению производительности и вызвать проблемы при взаимодействии приложения с базой данных.

Таблица №1 – Основные характеристики детектора FB.ODR\_OPEN\_DATABASE\_RESOURCE

|  |  |
| --- | --- |
| Категория критичности | Minor |
| Надежность | Unknown |
| Источник | SpotBugs |
| Тип ошибки | Open Database Resource |
| CWE | --- |
| CVE | --- |
| Возможные последствия | Утечка соединений с БД, исчерпание пула подключений, снижение производительности приложения, возможные сбои при длительной работе |

Таблица №2 – Примеры срабатывания детектора и устранения уязвимости

|  |  |
| --- | --- |
| Потенциально небезопасное | Корректная конструкция |
| public static void incorrectTest() {  try {  Connection conn = DriverManager.getConnection("jdbc:mysql://localhost/test", "user", "pass");  Statement stmt = conn.createStatement();  ResultSet rs = stmt.executeQuery("SELECT \* FROM users");    while (rs.next()) {  System.out.println(rs.getString("username"));  }  // ОШИБКА: ресурсы не закрываются явно  } catch (SQLException e) {  e.printStackTrace();  }  } | public static void correctTest() {  try (Connection conn = DriverManager.getConnection("jdbc:mysql://localhost/test", "user", "pass");  Statement stmt = conn.createStatement();  ResultSet rs = stmt.executeQuery("SELECT \* FROM users")) {    while (rs.next()) {  System.out.println(rs.getString("username"));  }  } catch (SQLException e) {  e.printStackTrace();  }  } |