FB.RI\_REDUNDANT\_INTERFACES

Этот класс заявляет, что реализует интерфейс, который также реализуется суперклассом. Это избыточно, поскольку, как только суперкласс реализует интерфейс, все подклассы по умолчанию также реализуют этот интерфейс. Это может указывать на то, что иерархия наследования изменилась с момента создания этого класса, и следует уделить внимание владельцу реализации интерфейса.

Таблица №1 – Основные характеристики детектора FB.RI\_REDUNDANT\_INTERFACES

|  |  |
| --- | --- |
| Категория критичности | Minor |
| Надежность | Unknown |
| Источник | SpotBugs |
| Тип ошибки | Code Quality Issue |
| CWE | --- |
| CVE | --- |
| Возможные последствия | Избыточность кода; Усложнение поддержки; Введение в заблуждение разработчиков |

Таблица №2 – Примеры срабатывания детектора и устранения уязвимости

|  |  |
| --- | --- |
| Потенциально небезопасное | Корректная конструкция |
| public static void incorrectTest() {  // Некорректно: избыточное наследование интерфейсов  interface Animal {  void eat();  }    interface Mammal extends Animal { // Избыточно, если не добавляет новых методов  void eat(); // Тот же метод, что и в Animal  }    class Dog implements Mammal {  @Override  public void eat() {  System.out.println("Dog is eating");  }  }    new Dog().eat();  } | public static void correctTest() {  // Корректно: использование только нужных интерфейсов  interface Animal {  void eat();  }    class Dog implements Animal {  @Override  public void eat() {  System.out.println("Dog is eating");  }  }    new Dog().eat();  } |