FB.MS\_MUTABLE\_COLLECTION\_PKGPROTECT

Экземпляр изменяемой коллекции присваивается конечному статическому полю, поэтому он может быть изменен вредоносным кодом или случайно из другого пакета. Поле можно сделать защищенным пакетом, чтобы избежать этой уязвимости. В качестве альтернативы можно обернуть это поле в Collections.unmodifyingSet/List/Map/etc. чтобы избежать этой уязвимости.

Таблица №1 – Основные характеристики детектора FB.MS\_MUTABLE\_COLLECTION\_PKGPROTECT

|  |  |
| --- | --- |
| Категория критичности | Minor |
| Надежность | Unknown |
| Источник | SpotBugs |
| Тип ошибки | Security Vulnerability |
| CWE | CWE218 |
| CVE | --- |
| Возможные последствия | * Несанкционированная модификация коллекции из других классов пакета * Нарушение инкапсуляции * Потенциальные race conditions |

Таблица №2 – Примеры срабатывания детектора и устранения уязвимости

|  |  |
| --- | --- |
| Потенциально небезопасное | Корректная конструкция |
| public static void incorrectTest() {  // Некорректно: package-protected изменяемая коллекция  class SharedData {  final List<String> data = new ArrayList<>(); // Видна в пакете    void addData(String item) {  data.add(item);  }  }    SharedData shared = new SharedData();  shared.addData("initial");    // Другой класс в пакете может модифицировать коллекцию  shared.data.clear();  System.out.println("Data compromised: " + shared.data);  } | public static void correctTest() {  // Корректно: защита коллекции от модификации  class SecureSharedData {  private final List<String> data = new ArrayList<>();    void addData(String item) {  data.add(item);  }    // Возвращаем неизменяемое представление  List<String> getData() {  return Collections.unmodifiableList(data);  }    // Или package-private метод для контролируемого доступа  List<String> getDataForPackage() {  return new ArrayList<>(data); // Защитная копия  }  }    SecureSharedData secure = new SecureSharedData();  secure.addData("secure");    // Попытка модификации через getData() вызовет исключение  try {  secure.getData().clear();  } catch (UnsupportedOperationException e) {  System.out.println("Collection properly protected");  }  } |