FB.DC\_PARTIALLY\_CONSTRUCTED

Похоже, этот метод использует ленивую инициализацию поля с двойной проверкой блокировки. Хотя поле правильно объявлено как изменчивое, возможно, что внутренняя структура объекта изменится после назначения поля, поэтому другой поток может увидеть частично инициализированный объект.

Чтобы решить эту проблему, стоит рассмотреть возможность сначала сохранить объект в локальной переменной и сохранить его в изменчивом поле только после того, как он будет полностью создан.

Таблица №1 – Основные характеристики детектора FB.DC\_PARTIALLY\_CONSTRUCTED

|  |  |
| --- | --- |
| Категория критичности | Minor |
| Надежность | Unknown |
| Источник | SpotBugs |
| Тип ошибки | Concurrency Vulnerability |
| CWE | CWE609 |
| CVE | --- |
| Возможные последствия | * Частично инициализированные объекты * Доступ к неинициализированным полям * Непредсказуемое поведение в многопоточной среде |

Таблица №2 – Примеры срабатывания детектора и устранения уязвимости

|  |  |
| --- | --- |
| Потенциально небезопасное | Корректная конструкция |
| public static void incorrectTest() {  class ResourceHolder {  private final Resource resource;    // Некорректно: "побег" this из конструктора  public ResourceHolder() {  this.resource = new Resource();  register(this); // Опасная публикация  }    public Resource getResource() {  return resource;  }  }    class Resource {}    // Может получить частично инициализированный объект  new ResourceHolder();  } | public static void correctTest() {  class SafeResourceHolder {  private final Resource resource;    // Корректно: полная инициализация перед публикацией  public SafeResourceHolder() {  this.resource = new Resource();  }    // Регистрация после полного создания объекта  public SafeResourceHolder register() {  Registry.register(this);  return this;  }    public Resource getResource() {  return resource;  }  }    class Resource {}    // Безопасное создание и регистрация  SafeResourceHolder holder = new SafeResourceHolder().register();  }  } |