PASS\_TO\_PROC\_AFTER\_CHECK

Когда проверка значения происходит, но после этой проверки значение передается в процедуру или функцию без дополнительной проверки. Это может быть потенциально опасно, если функция или процедура не предполагает получение некорректных или неожиданных значений и не выполняет свои собственные проверки. Это может привести к ошибкам выполнения, сбоям программы или уязвимостям безопасности.

Таблица №1 – Основные характеристики детектора

|  |  |
| --- | --- |
| Категория критичности | Normal |
| Надежность | High |
| Источник | SvEng |
| Тип ошибки | Ошибка разработчика |
| CWE | CWE129;CWE394;CWE606 |
| CVE | — |
| Возможные последствия | Передача некорректных данных в функцию может привести к сбоям программы или неправильному поведению. |
| Комментарий | Проверка значений до их передачи в функции помогает предотвратить неожиданные ошибки и защитить программу от возможных уязвимостей. |

Таблица №2 – Примеры срабатывания детектора и устранения уязвимости

|  |  |
| --- | --- |
| Некорректное написание | Корректное написание |
| public static void incorrectTest() {  int value = getValueFromInput();  if (value < 0 || value > 100) {  System.out.println("Invalid value in incorrectTest: " + value);  }  // Передача значения в функцию без дополнительной проверки  processValue(value);  } | public static void correctTest() {  int value = getValueFromInput();  if (value < 0 || value > 100) {  System.out.println("Invalid value in correctTest: " + value);  return;  }  // Передача значения в функцию только после проверки  processValue(value);  } |

Методы устранения уязвимости при срабатывании детектора:

1. Убедитесь, что проверка значения является исчерпывающей и покрывает все недопустимые или потенциально опасные случаи, которые могут вызвать проблемы в вызываемой процедуре.
2. Если проверка обнаруживает недопустимое значение, не передавайте его в процедуру. Вместо этого, обработайте ошибку (например, выведите сообщение, вернитесь из текущей функции, выбросьте исключение).