

МИНОБРНАУКИ РОССИИ

**Федеральное государственное бюджетное образовательное учреждение высшего образования**

**«МИРЭА – Российский технологический университет»**

**РТУ МИРЭА**

Институт кибербезопасности и цифровых технологий

Кафедра КБ-3 «Безопасность программных решений»

**ОТЧЕТ по дисциплине**

**«Разработка безопасного программного обеспечения»**

**Тема практической работы №4**  
«Использование статического анализатора кода для обеспечения безопасности»

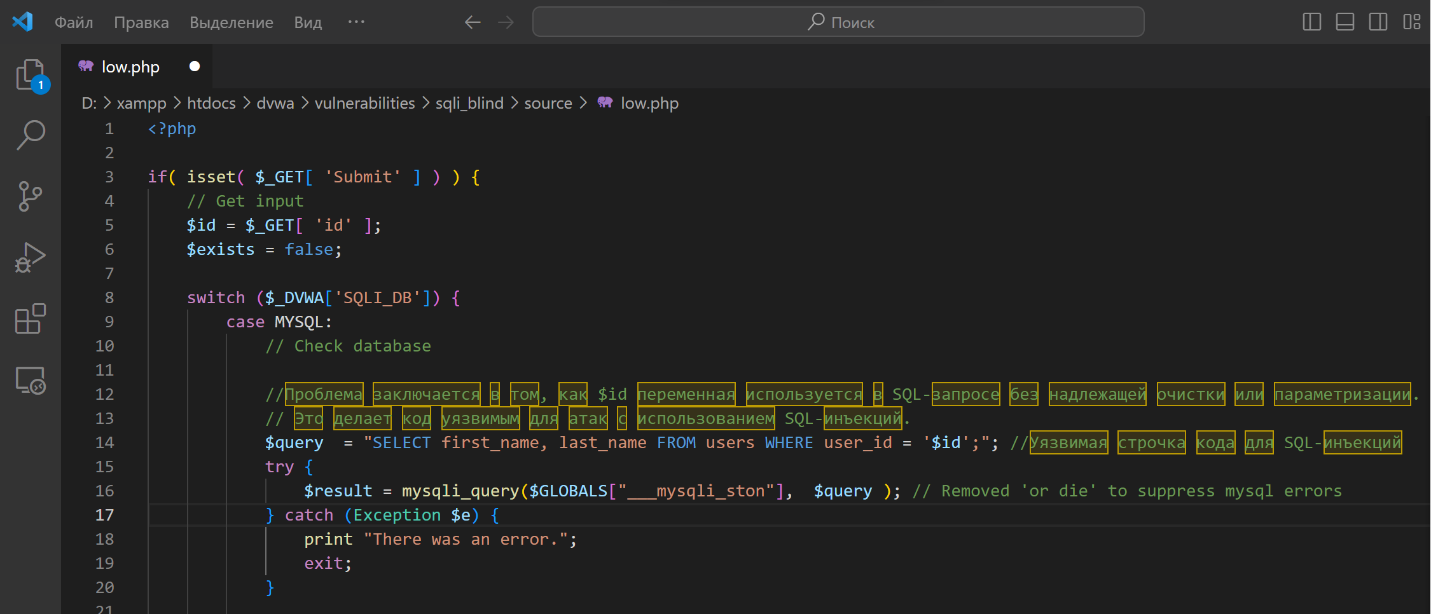
Студент: Королёв Николай Юрьевич

Шифр учебной группы: БСБО-03-20

Руководитель: Латыпов Ильдар Танисович

Москва 2023г.

**Задание №1**



**Задание №2**

Код, который был предоставлен, содержит несколько проблем, которые могут повлиять на безопасность и стабильность приложения. Вот некоторые из слабых мест:

1. SQL Инъекции:

$getid = "SELECT first\_name, last\_name FROM users WHERE user\_id = '$id';";

Ваш запрос подвержен SQL инъекциям, так как вы вставляете значение $id напрямую в запрос без проверки. Лучше всего использовать подготовленные запросы или хотя бы функцию mysqli\_real\_escape\_string для очистки ввода.

Пример использования подготовленного запроса:

$getid = "SELECT first\_name, last\_name FROM users WHERE user\_id = ?"; $stmt = mysqli\_prepare($GLOBALS["\_\_\_mysqli\_ston"], $getid); mysqli\_stmt\_bind\_param($stmt, "s", $id); mysqli\_stmt\_execute($stmt); $result = mysqli\_stmt\_get\_result($stmt);

1. Отсутствие валидации ввода:

$id = $\_GET[ 'id' ];

Здесь нет никакой валидации или фильтрации ввода. Вы должны проверять и очищать ввод, чтобы предотвратить атаки.

1. Отсутствие проверки типа данных:

$num = @mysqli\_num\_rows( $result );

Нет проверки типа данных, что может привести к нежелательным последствиям. Лучше использовать строгое сравнение, например, if ($num > 0).

1. Неполное закрытие соединения с базой данных:

((is\_null($\_\_\_mysqli\_res = mysqli\_close($GLOBALS["\_\_\_mysqli\_ston"]))) ? false : $\_\_\_mysqli\_res);

Это может не сработать в случае ошибок, и соединение не будет закрыто. Лучше использовать mysqli\_close($GLOBALS["\_\_\_mysqli\_ston"]) без оберток.

1. Использование устаревших функций MySQL:

$result = mysqli\_query($GLOBALS["\_\_\_mysqli\_ston"], $getid);

Рекомендуется использовать объектно-ориентированный способ работы с MySQLi, а не процедурный.

1. Отсутствие обработки ошибок:

$result = mysqli\_query($GLOBALS["\_\_\_mysqli\_ston"], $getid);

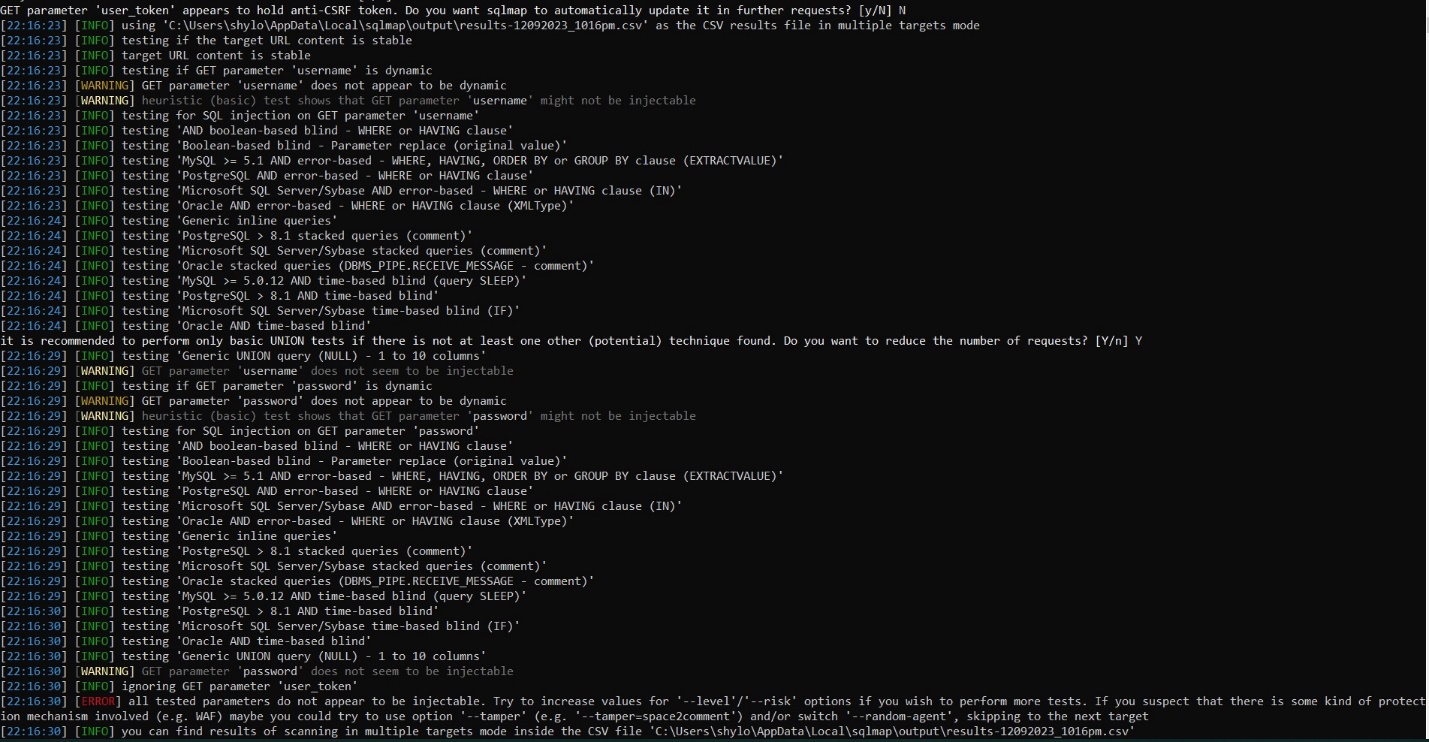
Ваш код не обрабатывает возможные ошибки, которые могут произойти при выполнении запроса. Обязательно добавьте обработку ошибок для более надежного кода.

Обратите внимание, что приведенные рекомендации могут помочь улучшить безопасность и надежность кода, но они не обязательно решают все возможные проблемы. Рекомендуется также изучить текущие стандарты безопасности для работы с базами данных в PHP.

**Задание №3**

https://github.com/byKol9n/Lab\_4\_dvwa

**Задание №4**



**Задание №5**

