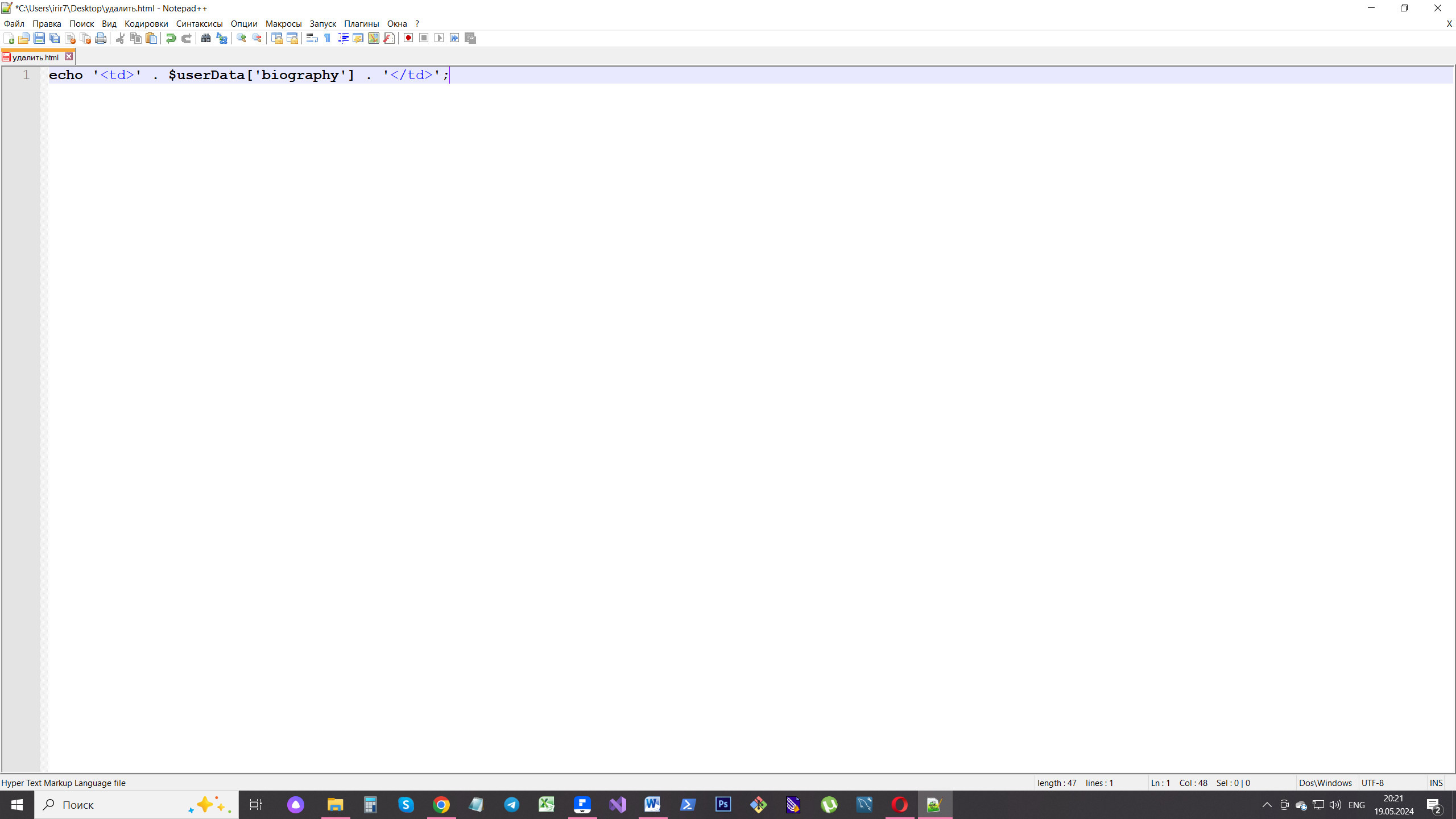
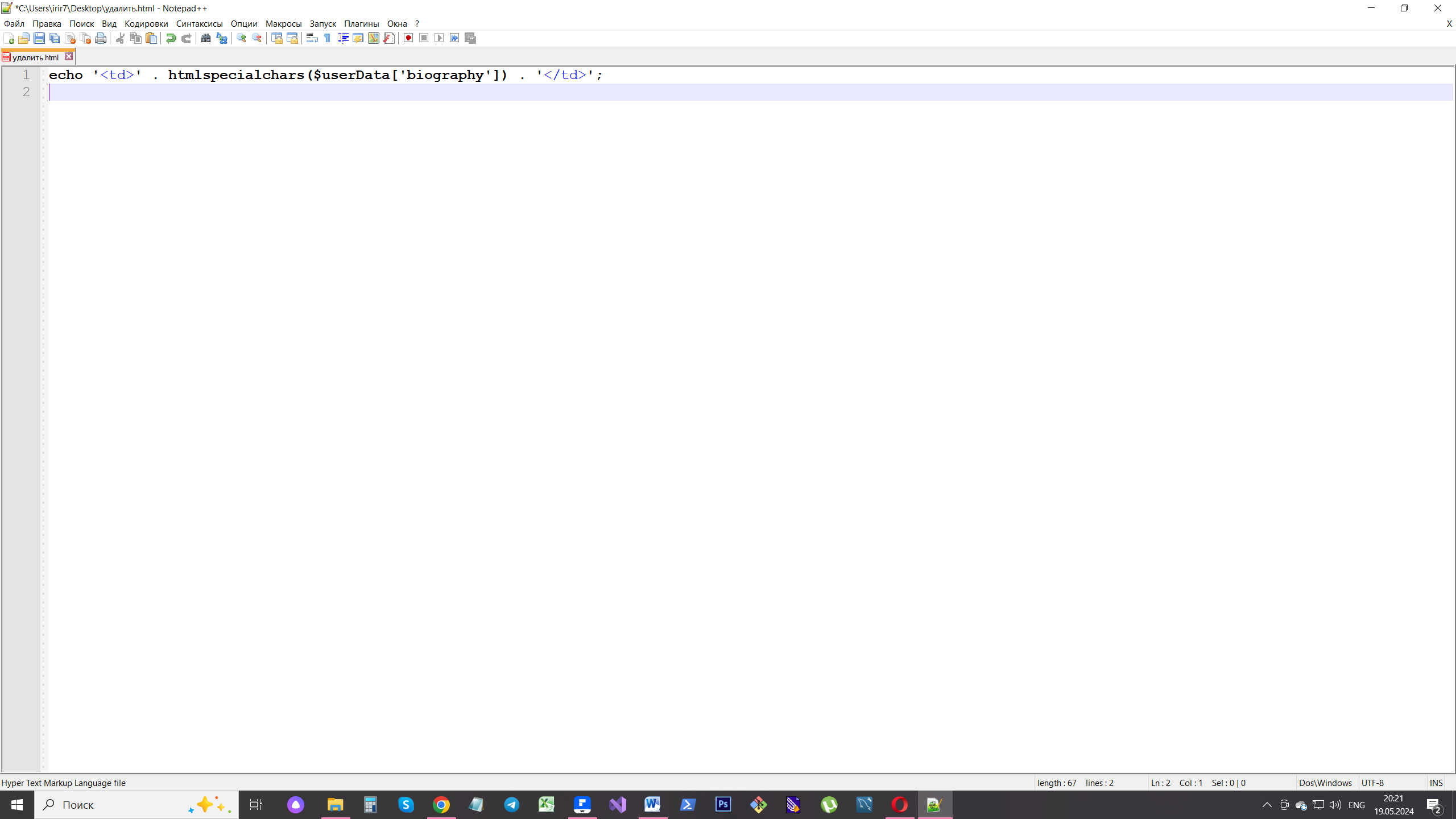
**Отчет об аудите безопасности**

1. Уязвимости XSS (межсайтовый скриптинг) возникают, когда веб-приложение доверяет непроверенным данным от пользователя и позволяет им внедрять и выполнять скрипты на стороне клиента. Это может привести к различным видам атак, таким как кража сеансов, изменение содержимого страницы, перенаправление пользователя и другие.

Пример проблемного кода:



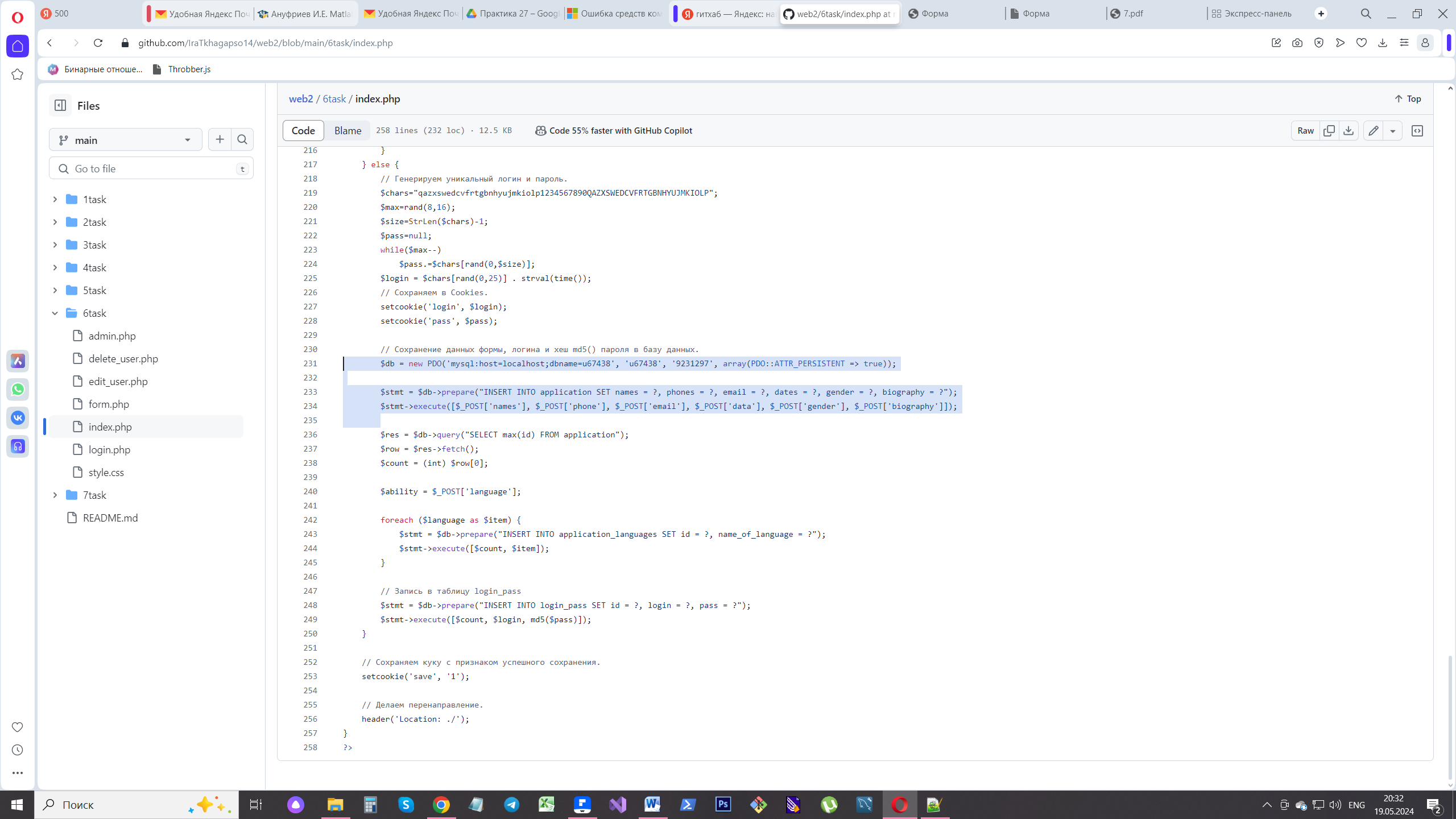
Пример непроблемного кода:



2. Уязвимости раскрытия информации (Information Disclosure) возникают, когда система или приложение случайно или намеренно раскрывает конфиденциальные данные пользователя или другую чувствительную информацию, которая не должна быть доступна широкой публике. Это может включать в себя такие данные, как личная информация, пароли, ключи API и т. д.

3. Уязвимости SQL Injection возникают, когда злоумышленник может внедрить зловредный SQL-код в SQL запрос, выполняемый веб-приложением. Это может привести к различным атакам, включая обход аутентификации, получение конфиденциальной информации, изменение или удаление данных из базы данных и другие вредоносные действия.

Пример проблемного кода:



Пример исправленного кода:

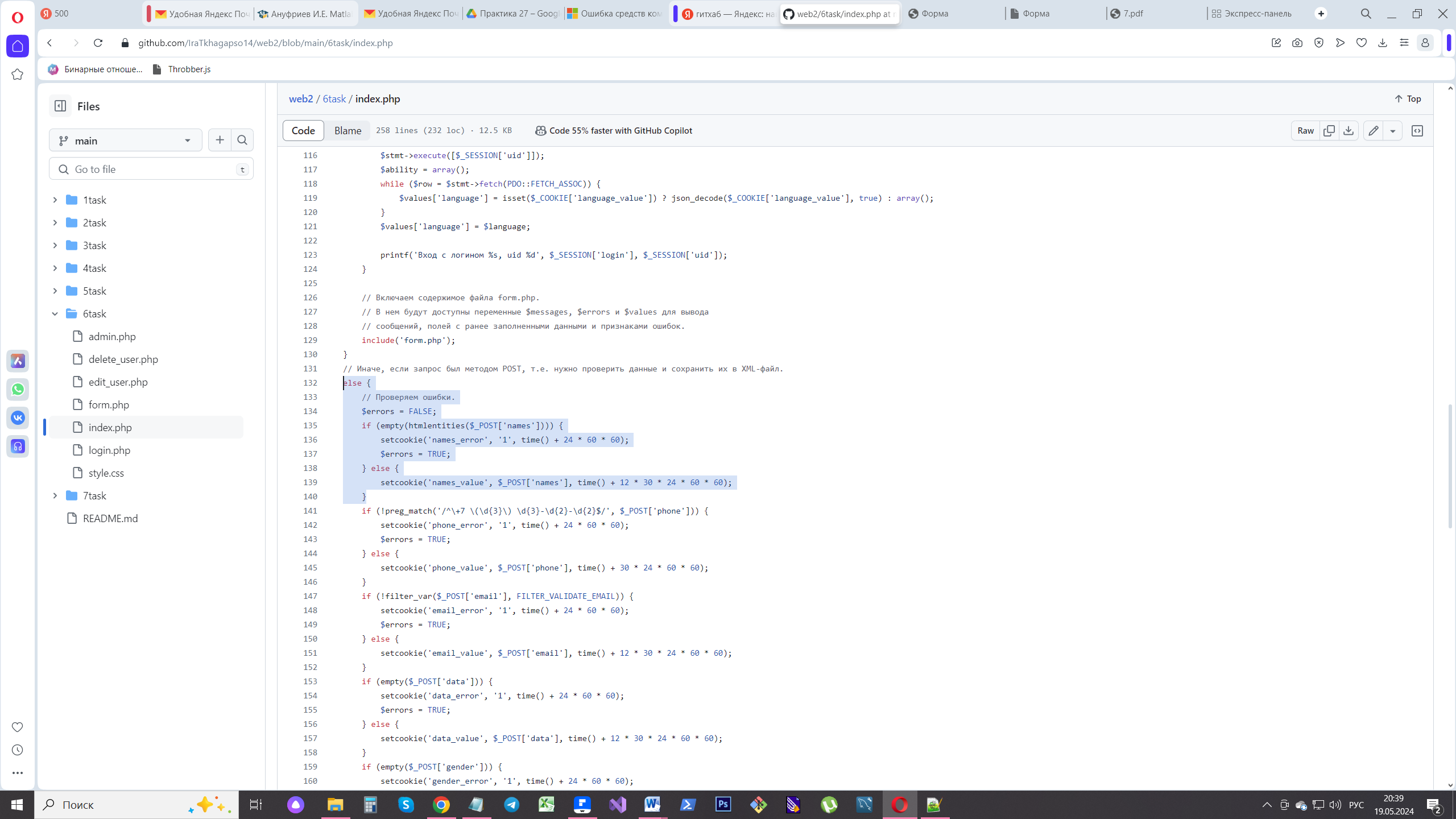
$stmt = $db->prepare("SELECT \* FROM application WHERE id = ?");

$stmt->execute([$\_SESSION['uid']]);

4. Уязвимости CSRF (межсайтовая подделка запроса) могут возникать в веб-приложениях, которые не надлежащим образом защищены от таких атак. Основные моменты, которые могут привести к возникновению уязвимости CSRF:

* Отсутствие проверки источника запроса: Если веб-приложение не проверяет источник запроса, то злоумышленники могут создать поддельный запрос, который будет отправлен от имени аутентифицированного пользователя.
* Использование сессионных cookie: Если сессионные cookie используются для аутентификации пользователя и не привязаны к конкретному IP-адресу или другим характеристикам сессии, злоумышленник может выполнить запрос от имени аутентифицированного пользователя.

Проблемный ход



Исправленный код:

session\_start();

if ($\_SERVER['REQUEST\_METHOD'] == 'POST') {

if (!isset($\_POST['csrf\_token']) || $\_POST['csrf\_token'] !== $\_SESSION['csrf\_token']) {

exit('Invalid CSRF token!');

}

unset($\_SESSION['csrf\_token']);

} else {

$\_SESSION['csrf\_token'] = bin2hex(random\_bytes(32));

}

5. Уязвимости, связанные с функциями include и upload, могут возникать в веб-приложениях из-за неправильного обращения к ним или недостаточной проверки данных, поступающих от пользователей