**Отчет: аудит безопасности**

**Задание 7**

Уязвимости:

|  |  |
| --- | --- |
| 1. Cross Site Scripting (XSS) | **Фильтрация (preg\_match)**  Она определяет данные, вводимые пользователем на соответствие заданным требованиям.  **Приведение к типу (is\_numeric)**  Проверяет, является ли переменная числом или же числовой строкой. |
| 1. SQL Injection | Использование подготовленных запросов (PDO), что позволяет отделить данные пользователя от данных запроса.  Экранирование(**addslashes**)  Возвращает строку с обратным слешом перед символами, которые нужно экранировать. Экранируются одиночная кавычка ('), двойная кавычка ("), обратный слеш (\) и NUL (NULL байт). |
| 1. Cross Site Request Forgery (CSRF) | Использование методов POST и GET по назначению.  XNR токены. |
| 1. Upload и Include | Использовался полный путь к файлу, чтобы избавиться от вероятности, что файл будет подменен (**dirname**) |

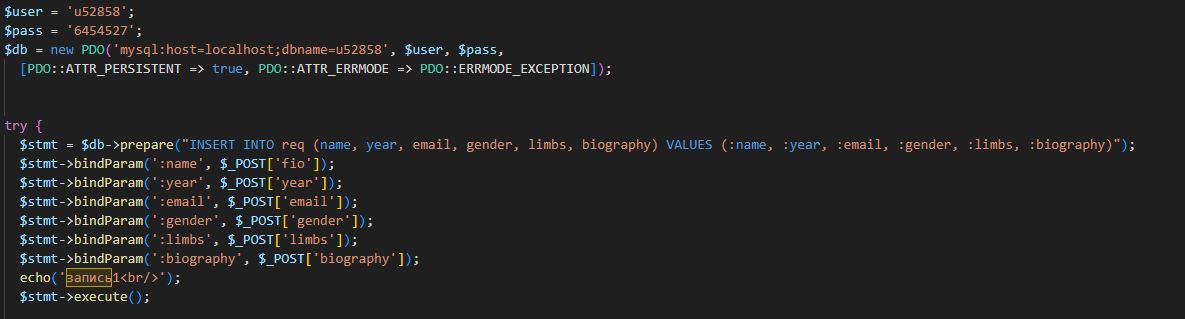
***Примеры из кода:***

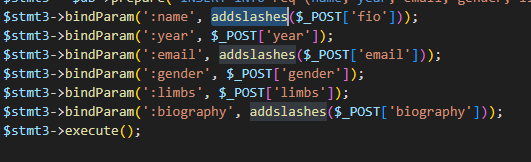
1. XSS:

C:\Users\user\Desktop\7\1.1.JPG

C:\Users\user\Desktop\7\1.3.JPG

1. SQL Injection





1. CSRF

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описаниеИзображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

1. Upload и Include

