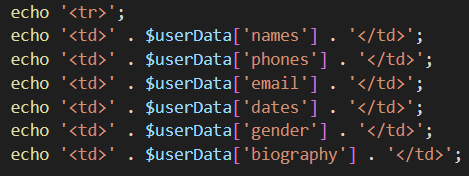
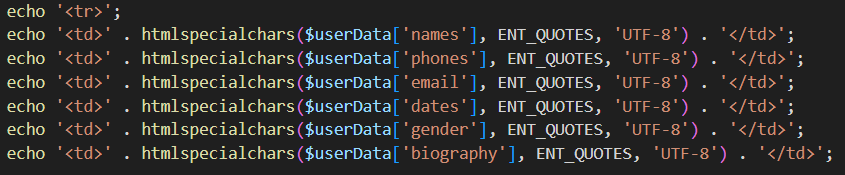
Отчет об аудите

1. Cross-Site Scripting (XSS)

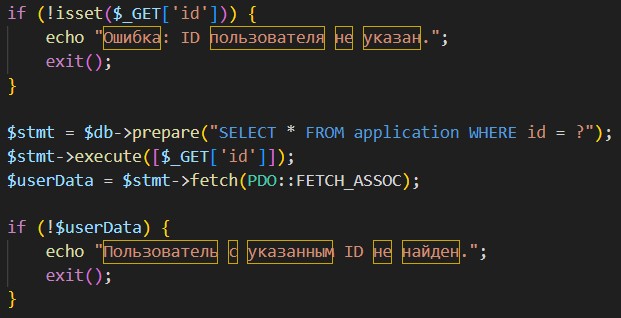
**XSS (Cross-Site Scripting)** – это когда злоумышленники "подкладывают" свой код на веб-страницы, чтобы выполнить недоброжелательные действия. Это может включать в себя кражу ваших данных, например, паролей или банковской информации. Представьте, что вы читаете книгу, и кто-то вклеивает туда страницу с ложной информацией – вот и XSS работает примерно так же.

Решение: использование функции htmlspecialchars для экранирования данных перед выводом. (admin.php)

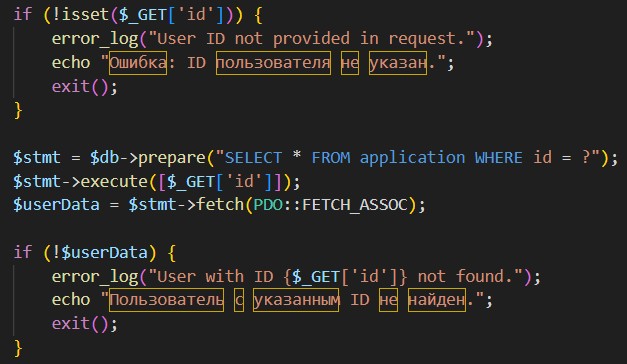


1. Information Disclosure

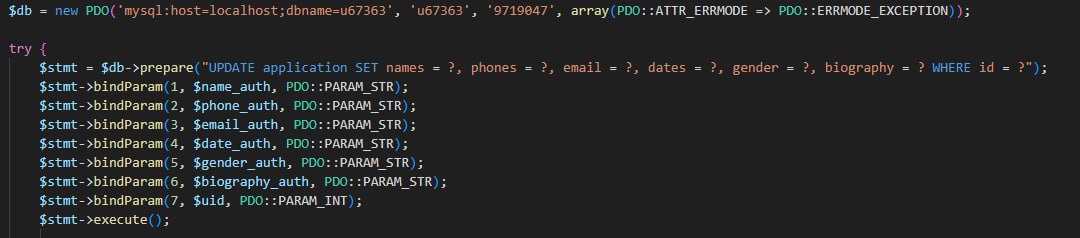
Уязвимость: приложение может раскрыть внутреннюю информацию (например, ошибки базы данных), которая может быть использована злоумышленниками для проведения атак.



Решение: включение пользовательских сообщений об ошибках вместо отображения внутренних ошибок. Логирование ошибок для внутреннего аудита вместо их отображения пользователю. (edit\_user.php)



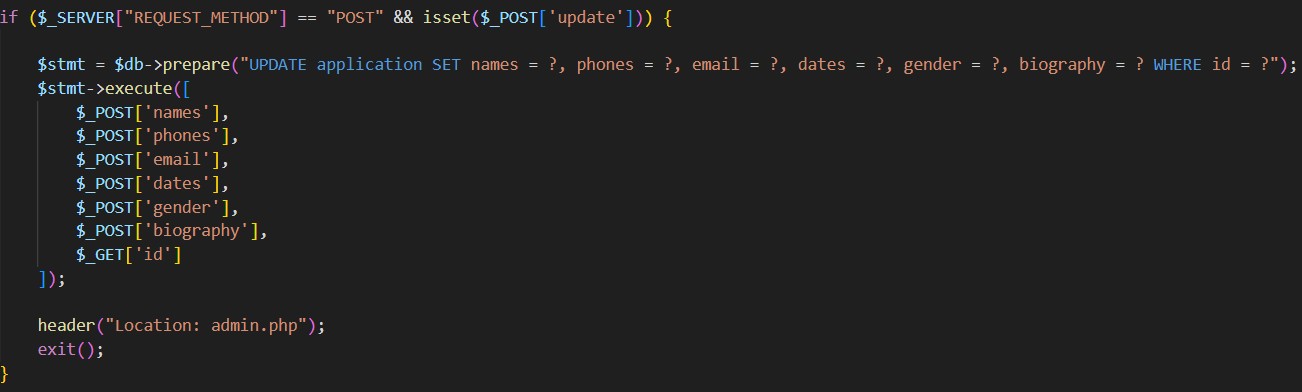
1. SQL Injection

Уязвимость: SQL Injection - это метод атаки, при котором злоумышленник может вставить или "инжектировать" свой собственный SQL-код в запросы к базе данных, что может привести к утечке данных, их изменению или удалению. 

Решение: в текущем коде уже используется подготовленный запрос для выборки данных пользователя по логину: (login.php)

Этот подход обеспечивает защиту от SQL Injection, так как параметры запроса передаются отдельно и обрабатываются драйвером базы данных.

1. CSRF (Cross-Site Request Forgery)

Уязвимость: отсутствие защиты от CSRF может позволить злоумышленнику выполнить нежелательные действия от имени пользователя.

Решение: использование CSRF-токенов для всех форм, выполняющих изменения данных. (admin.php)

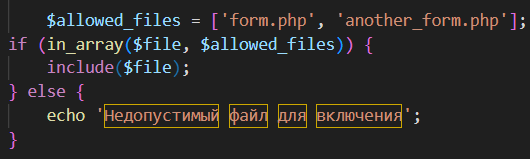


1. Include (Local File Inclusion)

Уязвимость: использование функции include с внешними данными может привести к атакам LFI (Local File Inclusion), что позволяет злоумышленникам включить произвольные файлы.



Решение: Использование жестко заданных путей для включаемых файлов. Например, при динамическом включении файлов, убедитесь, что включаются только файлы из разрешенного списка. (index.php)



1. Upload

Уязвимость: возникает, когда пользователи могут загружать файлы на сервер без должной проверки и ограничения. Это может привести к загрузке и выполнению вредоносного кода на сервере.

Решение: в текущем коде функциональность загрузки файлов отсутствует. Однако, если такая функциональность будет добавлена, необходимо реализовать следующие меры безопасности: ограничить типы файлов, которые могут быть загружены (например, только изображения), проверять MIME-тип файла и его расширение, использовать уникальные имена файлов для предотвращения перезаписи существующих файлов, хранить загруженные файлы вне корневого каталога веб-сервера, установить ограничения на размер загружаемых файлов.