Практическая работа №8

SQL-инъекции и способы защиты от них

Цель работы

Ознакомление с механизмами SQL-инъекций, их практическим применением для компрометации веб-приложений и методами защиты с использованием параметризованных запросов.

Ход выполнения работы

1. Подготовка и запуск тестового окружения

Запущено тестовое веб-приложение с использованием Docker Compose. Приложение содержит форму аутентификации, уязвимую к SQL-инъекциям. После успешного запуска по адресу `http://localhost:8080` открывается страница входа с полями для имени пользователя и пароля.

2. Проверка стандартной аутентификации

Сначала выполнена попытка входа с неверными учетными данными (user1/22221), которая закончилась неудачей с соответствующим сообщением об ошибке. Затем использованы корректные данные (user1/password1), что привело к успешной аутентификации и отображению приветственного сообщения.

3. Демонстрация SQL-инъекции

Проведена атака с использованием SQL-инъекции. В поле имени пользователя введена строка `' OR '1'='1' --`, поле пароля оставлено пустым. Данная инъекция привела к успешному обходу аутентификации без знания пароля.

Механизм работы инъекции:

- Кавычка (') завершает строковый параметр имени пользователя

- `OR '1'='1'` добавляет условие, которое всегда истинно
- `--` комментирует оставшуюся часть SQL-запроса, включая проверку пароля

4. Анализ серверных логов

Просмотр логов сервера показал, как изменяется SQL-запрос при атаке. В корректном запросе выполняется проверка и имени пользователя, и пароля. При инъекции запрос преобразуется таким образом, что проверка пароля игнорируется, а условие `'1'='1'` гарантирует возврат хотя бы одной записи из базы данных.

5. SQL-инъекция для изменения пароля

Выполнена вторая SQL-инъекция для модификации пароля пользователя `admin`.

6. Реализация защиты

Для защиты приложения от SQL-инъекций модифицирован исходный код, заменена конкатенация строк параметризованными запросами. Параметризованные запросы разделяют код SQL и данные, предотвращая интерпретацию пользовательского ввода как части SQL-команды.