Что такое SQLI атака и как она работает

Данный тип атак направлен на формы для пользовательского ввода, чтобы получить доступ к базе данных на стороне сервера с целью ее удаления, изменения, копирования. В строки ввода злоумышленник пишет специальный код на «Языке структурированных запросов», то есть SQL.

Существуют два типа вредоносных строчек на SQL. Первый из них самый простой.

Представим, что на стороне сервера есть кусок кода, который подключается к базе данных (например, к файловой СУБД SQLite) и без какой-либо проверки подставляет введенные пользователем значения логина и пароля. Представим, что пользователь действительно ввел в поле логина и пароля такую пару: user:pass. Такой запрос будет выглядит примерно так:

SELECT username, password FROM users WHERE username='user' AND password='pass';

Так как в этом примере нет элементарной проверки и экранирования пользовательского ввода, то злоумышленник может ввести в поле логина такую запись: ' or 1=1 -- -, а вместо пароля любые символы, например, 123.

Получается следующий запрос в базу данных:

SELECT username, password FROM users WHERE username='' OR 1=1;-- - AND password='123';

Поскольку символы «-- -« считаются комментарием в языке SQL, то все, что следует после них можно откинуть, так как сервер это не читает. Когда этот запрос обрабатывается, сервер видит, что 1=1, что истинно верно и в ответ на это открывает доступ к аккаунту без авторизации.

Второй способ чуть сложнее и он использует оператор UNION, который объединяет команды. Он позволяет также проникнуть в систему, а также выгрузить столбцы из базы данных. Оператор UNION требует, чтобы в обоих запросах было одинаковое количество столбцов. Злоумышленник подбирает это число, добавляя в запрос дополнительные поля, часто через null

Чтобы это сработало, надо знать, какую бд использует это веб-приложение. Есть специальная таблица запросов, позволяющая это узнать. Если ввести

' union select sqlite_version(),null; -- -

То мы получаем версию базы данных sqlite.

Реальные риски от этой атаки

Уязвимость SQL-Injection может не только послужить инструментом для незаконного доступа к базе данных (это может повлечь за собой удалению клиентской базы, слив пользовательских данных, потерю доверия и репутации, взлом аккаунтов), но и в качестве помощи в осуществление других атак (Dos, XXS).

Как защититься

- Использование параметризованных запросов (Prepared Statements): это самый главный и эффективный метод. Серверный код отправляет запрос и данные раздельно, что исключает возможность их смешивания и интерпретации введенных данных как кода.
- Экранирование пользовательского ввода: хотя это менее надежно, чем параметризованные запросы, для некоторых случаев можно применять функции экранирования специальных символов.
- Принцип наименьших привилегий: у учетной записи приложения для доступа к БД должны быть минимально необходимые права (только SELECT, INSERT и т.д., но не DROP).
- Хранение паролей правильно: пароли должны хешироваться с использованием современных алгоритмов (bcrypt, Argon2) и "соли" (salt), чтобы защитить их даже в случае утечки.