

пользователей [1]. Запретить им никто не может, потому что все находятся в равных условиях, обладают равными правами и могут в равной степени исполнять и даже нарушать свои обязанности. Все пользователи blockchain образуют собой сеть компьютеров, на каждом из которых хранится копия данных blockchain. Обычно это полная копия всех блоков, но, в принципе, можно хранить лишь нужные на конкретном компьютере данные. Благодаря этому выключить или сломать blockchain практически невозможно, поскольку для этого надо выключить или сломать все компьютеры. Пока есть хоть один пользователь, blockchain существует [1]. Каждый новый пользователь расширяет и укрепляет эту сеть. Причем все компьютеры равноправны, там нет организаторов, модераторов, контролеров и менеджеров. Каждый отвечает за себя сам. Криптография – это основа blockchain, которая обеспечивает работу системы. Криптография в blockchain гарантирует безопасность, причем основанную на прозрачности и проверяемости всех операций. Различные криптографические техники гарантируют неизменность журнала транзакций blockchain, решают задачу аутентификации и контролируют доступ к сети и данным в blockchain в целом. Исходя из вышеизложенного, можно сделать вывод о том, что использование описанной технологии позволяет достаточно надежно осуществлять защиту информации. Рассмотренная технология достаточно успешно используется авторами работы.

Литература

1. Antonopoulos, A. Mastering Bitcoin: Unlocking Digital Cryptocurrencies. O'Reilly Media, 2014. 298 p.

МЕТОДЫ ЗАЩИТЫ ОТ SQL-ИНЪЕКЦИЙ

М.П. Хоронко, М.А. Медунецкий, А.С. Летохо

SQL-инъекции были и остаются наиболее критичными уязвимостями приложений. Обычно атаки типа SQL-инъекций рассматривают относительно web-приложений, однако данной уязвимости подвержены любые клиент-серверные и сервис-ориентированные приложения, работающие с системами управления базами данных. SQL-инъекция – это атака, при которой злоумышленник производит вставку вредоносного кода в строки, передающиеся на сервер СУБД для синтаксического анализа и выполнения. Данная уязвимость состоит в том, что веб-приложение некорректно проверяет данные от пользователя, которые в дальнейшем используются для генерации SQL-запросов к базе данных.

SQL-инъекции можно классифицировать следующим образом:

- 1) по месту нахождения в запросе (инъекции в строковом параметре и в числовом параметре);
- 2) по способу внедрения инъекции (обычные и «слепые»).

Для успешного внедрения SQL-инъекций необходимо знать следующие параметры: способ передачи данных в веб-приложении, тип и версию СУБД, имя текущего пользователя, назначенные роли и системные привилегии.

В зависимости от назначенных текущему пользователю ролей и привилегий злоумышленник может попытаться извлечь хэши паролей пользователей СУБД, получить структуру БД и конфиденциальные данные или выполнить команды ОС, а также доступ к функциональным возможностям СУБД, а в некоторых случаях – к операционной системе сервера, на котором функционирует СУБД.

Приведем основные методы защиты веб-приложений:

- максимально возможная фильтрация данных;
- использовать при сравнении кавычки SELECT ...WHERE name=''\$name';
- фильтрация символов “%” и “_” при использовании SQL-функции LIKE.
- создание белого списка запросов на которые будет отвечать приложение.

ОБЕСПЕЧЕНИЕ БЕЗОПАСНОСТИ ANGULAR ПРИЛОЖЕНИЙ

А.Л. Хотеев, А.С. Дроздов, Н.В. Харитонов, Е.И. Нехведович

Мир программного обеспечения эволюционирует с огромной скоростью. Всего пару лет назад настольные компьютеры и ноутбуки являлись основными устройствами, под которые