АНАЛИЗ БЕЗОПАСНОСТИ В СУБД MYSQL КАК ХРАНИЛИЩЕ ДАННЫХ ДЛЯ СИСТЕМ ДИСТАНЦИОННОГО ОБУЧЕНИЯ

И.П. Навицкий[1], С.С. Куликов[2]

[1] *Белорусский государственный университет информатики и радиоэлектроники, Минск, Беларусь; roocster@gmail.com*

[2] *Белорусский государственный университет информатики и радиоэлектроники, Минск, Беларусь; kulikov@bsuir.by*

Abstract. At the present stage of development of information technology, databases are an integral part of the software. In this regard, in the process of training, including distance, it-professionals should receive in-depth knowledge in the field of databases.

В настоящее время довольно часто встречаются ситуации, когда на одном сервере располагается несколько баз данных, которые управляются одной локальной клиент-серверной СУБД (системой управления базами данных). СУБД располагается на сервере вместе с БД и осуществляет доступ к БД непосредственно, в монопольном режиме. Все клиентские запросы на обработку данных обрабатываются клиент-серверной СУБД централизованно.

Подключаться к серверу для работы с базами данных могут различные ресурсы, причём никак не связанные между собой. В связи с этим, становятся актуальными вопросы обеспечения безопасности и распределения доступа к данным.

Используемая в MySQL система безопасности для всех подключений, запросов и иных операций, которые может пытаться выполнить пользователь, базируется на списках контроля доступа ACLs (Access Control Lists) [1].

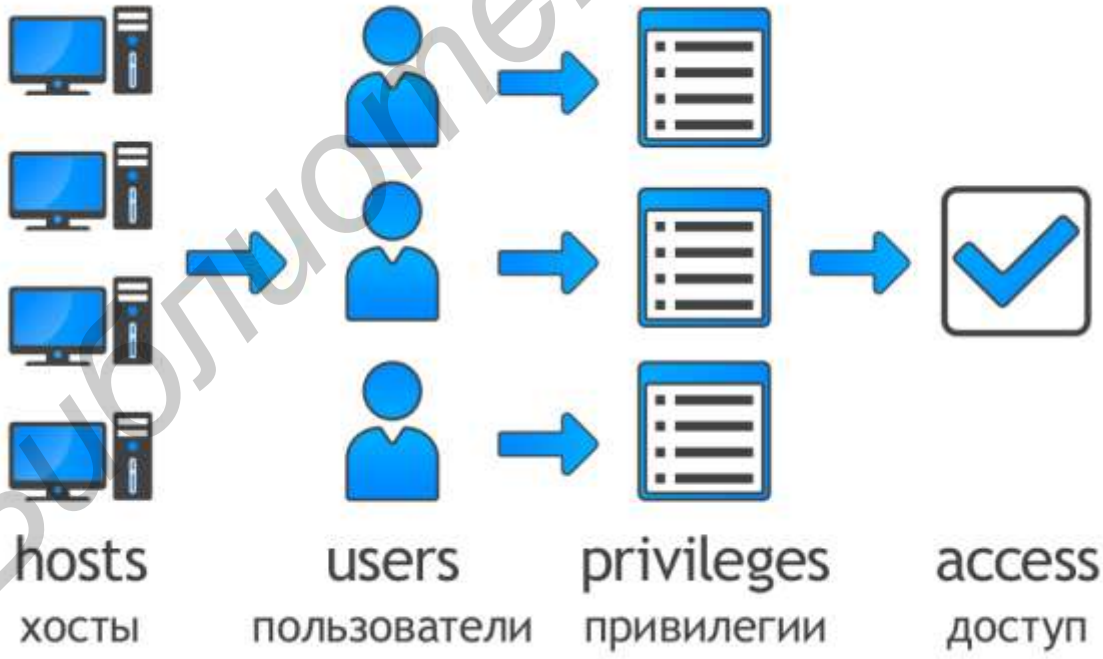

Рисунок 1 – Порядок получения доступа

При обсуждении вопросов безопасности акцентируется внимание на необходимости защиты всего серверного хоста (а не одного лишь сервера MySQL) от

всех возможных типов атак: перехвата, внесения изменений, считывания и отказа в обслуживании.

Основной функцией системы привилегий MySQL является аутентификация пользователя, подключающегося с указанного хоста, и ассоциирование его с привилегиями базы данных, такими как SELECT, INSERT, UPDATE и DELETE. Контроль доступа осуществляется с помощью трёх полей контекста таблицы user (Host, User и Password) [2]. На рисунке 1 представлена схема порядка проверки привилегий и получения доступа к базе данных.

Для каждого пользователя в СУДБ MySQL задаётся его доступность с различных хостов. В качестве хоста может указываться его имя, IP-адрес, диапазон IP-адресов, либо значение 'localhost' (доступ с локального хоста). Привилегии пользователей выставляются на 4-х уровнях: глобальный уровень, уровень баз данных, уровень таблиц и уровень столбцов.

Исходя из вышеописанного, анализ безопасности в рамках СУБД MySQL может осуществляться путём проверки доступности пользователей, а также выявления небезопасных привилегий, которыми они обладают. Нарушением безопасности можно считать ситуацию, когда пользователь root доступен вне сервера (localhost) или, по крайней мере, вне локальной сети. Доступ к обычным пользователям (не root), в большинстве случаев, требуется обеспечить только с одного хоста или локальной сети, однако возможность доступа к пользователям с нескольких хостов нельзя трактовать как нарушение безопасности, но не будет лишним сформировать список с такими пользователями для изучения администратором.

СУБД MySQL допускает отсутствие пароля у пользователей. В связи с этим, необходимо выявить всех пользователей у которых не установлен пароль. При анализе привилегий пользователей, в первую очередь следует проверить привилегии глобального уровня. В большинстве случаев, наличие глобальных привилегий требуется только у пользователя root. Наличие глобальных привилегий у других пользователей представляет потенциальную опасность для системы. Наиболее опасны такие глобальные привилегии как: DELETE, DROP, SHUTDOWN, GRANT, ALTER, PROCESS, SUPER, LOCK_TABLES, EXECUTE, ALTER_ROUTINE, CREATE_USER. Доступ к системным таблицам, таким как mysql.users например, следует предоставлять только для пользователя root.

В качестве дополнительной меры обеспечения безопасности можно просканировать порты с помощью утилиты типа nmap. MySQL использует по умолчанию порт 3306. Этот порт должен быть недоступен с неблагонадёжных компьютеров. Также для проверки открыт порт или нет, можно попытаться установить соединение через Telnet. Если соединение будет установлено, это будет означать, что порт открыт, и его следует закрыть на брандмауэре или маршрутизаторе (если, конечно, нет действительно веских причин держать его открытым) [1].

Литература

1. mysql.ru [Электронный ресурс]. – Электронные данные. – Режим доступа: http://www.mysql.ru/.

2. Кузнецов, М. В. MySQL 5 / М. В. Кузнецов, И. В. Симдянов. – СПб: БХВ-Петербург, 2010. – 1024 с.