Лекция
21.02.2019

DBA1. p8.

Привилегии и роли

Ильшат Каразбаев
руководитель группы DBA
АО ТК Центр

Немного обо мне

Вместе со своей командой администрирую:

СУБД MySQL, Mariadb, galeracluster, Postgres

Главный по базам в ТК Центр

Повестка дня:

1. Вводная
2. Привилегии
3. Автоматизация
4. Демонстрация
5. Задания
6. Литература

Вводная.

1. GRANT используется для делегирование привилегий учетным записям или ролям
2. Для запуска команды GRANT необходима привилегия GRANT OPTION
3. REVOKE используется для отзыва делегированных ранее привилегий
4. С помощью команды GRANT можно создать пользователя и сразу делегировать ему привилегии (если не задана опция NO_AUTO_CREATE_USER в переменной SQL_MODE)
5. Справка по привилегиям SHOW PRIVILEGES

USAGE привилегия

USAGE не дает никаких привилегий. Ее можно исользовать для назначения GRANT OPTION, MAX_USER_CONNECTIONS и других опций без изменений привилегий аккаунта.

GRANT OPTION привилегия

Привилегия GRANT OPTION дает право делегировать назначение привилегий другому пользователю, но только на те сущности, куда имел доступ делегирующий привилегию.

Назначение привилегии GRANT OPTION на отдельные колонки не сработает - будет дана привилегия на всю таблицу.

WITH GRANT OPTION и GRANT OPTION эквивалентны

Управление привилегиями. Уровни привилегий

1. Глобальные, ‘*.*’. Глобальные привилегии включают привилегии для администрирования СУБД, пользователями и их доступами, функциями и процедурами, а также на операции со всеми базами СУБД. Привилегии хранятся в таблице mysql.user
2. Уровня базы данных, ‘db_name.*’. Хранятся в mysql.db
3. Уровня таблиц, ‘db_name.table_name’
4. Уровня колонок. Можно дать доступ на операции с определенным набором колонок в таблице.
5. Уровня функций, FUNCTION db_name.routine_name или FUNCTION routine_name
6. Уровня хранимых процедур, PROCEDURE db_name.routine_name или PROCEDURE routine_name

Глобальные привилегии

1. CREATE USER - управление пользователями и ролями
2. FILE - работа с файлами (LOAD DATA INFILE, LOAD_FILE(), CONNECT)
3. GRANT OPTION - управление привилегиями
4. PROCESS - просмотр SHOW PROCESSLIST
5. RELOAD - отсечение логов (FLUSH)
6. REPLICATION CLIENT - SHOW MASTER STATUS, SHOW SLAVE STATUS
7. REPLICATION SLAVE - получение обновлений с сервера
8. SHOW DATABASES
9. SHUTDOWN
10. SUPER - CHANGE MASTER TO, KILL, PURGE, SET GLOBAL

Уровня базы данных

1. CREATE - создание базы, можно дать привилегию CREATE на несуществующую БД
2. CREATE ROUTINE - CREATE PROCEDURE, CREATE FUNCTION
3. CREATE TEMPORARY TABLES
4. DROP - DROP DATABASE
5. EVENT - операции с event
6. GRANT OPTION
7. LOCK TABLES - для этой привилегии нужна еще и SELECT привилегия на таблицу

Уровня таблицы.

1. ALTER/ CREATE
2. CREATE VIEW/ SHOW VIEW
3. INSERT/ DELETE/ UPDATE
4. DELETE HISTORY - удаление версионных строк (начиная с 10.3.4)
5. DROP
6. GRANT OPTION
7. INDEX (можно создать индексы и без этой привилегии с помощью ALTER/CREATE)
8. TRIGGER

Уровня колонок

1. INSERT
2. SELECT
3. UPDATE

Уровня функций и хранимых процедур

1. ALTER ROUTINE - ALTER FUNCTION, ALTER PROCEDURE
2. EXECUTE - CALL
3. GRANT OPTION

Уровня прокси

1. PROXY - делегирование привилегий прокси

Чаще всего используется с PAM authentication plugin, не поддерживается mysql_native_password

Опции аутентификации

В открытом виде:

1. GRANT USAGE ON *.* TO ‘foo’@’ip’ IDENTIFIED BY ‘unencrypted_pass’;

С помощью хеша:

1. SELECT PASSWORD(‘unencrypted_pass’);
2. GRANT USAGE ON *.* TO ‘foo’@’ip’ IDENTIFIED BY PASSWORD ‘password_hash’;

С помощью плагинов:

1. SHOW PLUGINS;
2. GRANT USAGE ON *.* TO ‘foo’@’ip’ IDENTIFIED VIA pam;

Управление пользователями. Пароли

В открытом виде:

1. CREATE USER ‘foo’@’ip’ IDENTIFIED BY ‘unencrypted_pass’;

С помощью хеша:

1. SELECT PASSWORD(‘unencrypted_pass’);
2. CREATE USER ‘foo’@’ip’ IDENTIFIED BY PASSWORD ‘password_hash’;

С помощью плагинов:

1. SHOW PLUGINS;
2. CREATE USER ‘foo’@’ip’ IDENTIFIED VIA pam;

Управление привилегиями. Шифрование трафика

(как в прошлой презентации)

```
GRANT USAGE ON *.* TO 'someone'@'localhost'
   REQUIRE SUBJECT '/CN=www.mydom.com/O=My Dom, Inc./C=US/ST=Oregon/L=Portland'
   AND ISSUER '/C=FI/ST=Somewhere/L=City/ O=Some Company/CN=Peter Parker/emailAddress=p.parker@marvel.com'
   AND CIPHER 'SHA-DES-CBC3-EDH-RSA';
```

Option	Description
`REQUIRE NONE`	TLS is not required for this account, but can still be used.
`REQUIRE SSL`	The account must use TLS, but no valid X509 certificate is required.
`REQUIRE X509`	The account must use TLS and must have a valid X509 certificate.
`REQUIRE ISSUER 'issuer'`	The account must use TLS and must have a valid X509 certificate. Also, the Certificate Authority must be the one specified via the string `issuer`.
`REQUIRE SUBJECT 'subject'`	The account must use TLS and must have a valid X509 certificate. Also, the certificate's Subject must be the one specified via the string `subject`.
`REQUIRE CIPHER 'cipher'`	The account must use TLS and must have a valid X509 certificate. Also, the encryption used for the connection must use one of the methods specified in the string `cipher`.

Управление привилегиями. Ограничение ресурсов

Начиная с версии MariaDB 10.2

```
GRANT USAGE ON *.* TO 'someone'@'localhost' WITH
   MAX_USER_CONNECTIONS 0
   MAX_QUERIES_PER_HOUR 200;
```

Limit Type	Decription
MAX_QUERIES_PER_HOUR	Number of statements that the account can issue per hour (including updates)
MAX_UPDATES_PER_HOUR	Number of updates (not queries) that the account can issue per hour
MAX_CONNECTIONS_PER_HOUR	Number of connections that the account can start per hour
MAX_USER_CONNECTIONS	Number of simultaneous connections that can be accepted from the same account; if it is 0, `max_connections` will be used instead; if `max_connections` is 0, there is no limit for this account's simultaneous connections.
MAX_STATEMENT_TIME	Timeout, in seconds, for statements executed by the user. See also Aborting Statements that Exceed a Certain Time to Execute.

Управление привилегиями. Роли

GRANT role TO grantee [, grantee2 ...]
[WITH ADMIN OPTION]

GRANT journalist TO hulda;

GRANT journalist TO berengar WITH ADMIN OPTION;

WITH ADMIN OPTION - позволяет делегировать привилегии роли другому пользователю

Управление привилегиями. root-like

```
CREATE USER 'alexander'@'localhost';
GRANT ALL PRIVILEGES ON  *.* to 'alexander'@'localhost' WITH GRANT
OPTION;
```

Демонстрация

Задания

1. Создайте пользователя и назначьте ему привилегии из каждого уровня привилегий: глобальную, уровня бд, таблиц, колонок, функций и хранимых процедур.

Вопросы?

telegram: karazbaev

vk, instagram: barazbay

twitter: karazbay

Литература

1. https://mariadb.com/kb/en/library/user-account-management/
2. https://mariadb.com/kb/en/library/grant/
3.