GNU/Linux-MariaDB

安全管理

一、理解访问控制

MariaDB服务器的基本安全策略：用户应该对他们需要的数据具有适当的访问权限。我们需要对用户提供其需要的访问权限，并且只提供他们需要的访问权限。这就是所谓的访问控制（access control)，并且管理访问控制就需要创建和管理用户账户。

在首次安装时，MariaDB创建了一个名为root的账户，其对MariaDB服务器具有完全的控制权限。但在现实的工作生产环境中，觉不应该使用root。解决方案是，创建一些列账户，有的供管理员使用，有的给用户，有的提供给开发者等。

GNU/Linux-MariaDB

安全管理

二、用户管理

MariaDB用户账户和信息都存储在名为mysql的MariaDB数据库中。当我们需要获取所有用户账户列表的时候，可以使用以下语句：

```
use mysql;
select user from user;
```

注：这个mysql数据库中拥有一个包含所有账户的名为user的表，user表有一个名为user的列，包含用户的登录名。

GNU/Linux-MariaDB

安全管理

1.创建用户账户

为了创建一个新的用户账户，使用create user语句：

create user ben IDENTIFIED BY ‘p@$$w0rd’;

注：1）create user 创建了一个新的用户账户，这个例子中使用 identified by ‘p@$$w0rd’指定了密码。

2）如果再次列示出用户账户，将会在输出中看到新创建的账户

3）identified by 以纯文本方式指定密码，MariaDB将其存储到user表之前会先对其进行加密。为了将散列值作为密码指定，可以使用identified by password

GNU/Linux-MariaDB

安全管理

4)使用grant或者insert语句创建用户：使用grant语句也可以创建用户账户。另外，直接插入行到user表的方式添加用户，但这种做法不推荐，因为MariaDB用于存储用户账户信息的表（及表的框架）十分重要，对他们的任何损害都会严重影响MariaDB服务器，因此，使用标记和函数操作这些表，会比直接操作他们更好。

GNU/Linux-MariaDB

安全管理

2.重命名用户账户：

rename user ben to bforta;

3.删除用户账户：

drop user bforta;

注：删除用户账户也会删除其相关权限。

GNU/Linux-MariaDB

安全管理

4.设置访问权限

新创建的用户完全没有任何访问权限，他们可以登录MariaDB，但是不能看到数据且不能对任何数据库执行操作。可以使用下面的语句查看用户的权限：

show grants for bforta;

注：结果中显示用户bforta有一个权限“USAGE ON *.* ”代表该用户对任何数据库和任意表上的任何数据都没有权限。

为了设置权限，可以使用GRANT语句，这个语句可以指定以下信息：

授予的权限

授予权限的数据库和表

用户名

如：

grant select on crashcourse.* to bforta;

GNU/Linux-MariaDB

安全管理

注：1）上述语句允许用户对crashcourse.*（即crashcourse数据库中的所有表）进行select操作，即用户bforta拥有对crashcourse数据库所有的可读情况。可用show grants查看更改后的权限。

2）每个grant对用户添加或者更新一个权限，MariaDB读取所有的授权，并且基于此判定他们的权限。

5.撤销权限

使用REVOKE语句来撤销权限：

revoke select on crashcourse.* from bforta;

注：1）在撤销权限时，撤销的权限必须存在。Grant和revoke可以多次层次的控制访问权限：

整个服务器，使用grant all和revoke all

整个数据库，使用on database.*

特定的表，使用no database.table

特定的列

特定的存储过程

附录中列出了可被赋予和撤销的权限。

GNU/Linux-MariaDB

安全管理

2）当使用grant和revoke是，用户账户必须存在，而设计到的对象可以不存在，这样管理员可以在数据库更改之前甚至是创建表之前就可以设计安全策略。

3）可以使用,分隔，将多个grant语句结合起来用以简化多次授权：

```
grant select,insert on crashcourse.* to bforta;
```

5.修改密码

使用set password语句修改密码，新密码必须按照如下方式加密：

```
set password for bforta = password('n3w pa$$w0rd');
```

注：1) set password更新用户密码，新密码必须传递给password()函数加密。

2）set password也可以用来设置你自己的密码：

```
set password = password('n3w pa$$w0rd');
```

即没有指定用户时，set password更新当前登录的用户密码。

GNU/Linux-MariaDB

安全管理

附录：权限

权限	说明
ALL	除了grant option之外所有权限
ALTER	使用alter table
ALTER ROUTINE	使用alter procedure和drop procedure
CREATE	使用create table
CREATE TEMPORAY TABLE	使用create temporary table
CREATE ROUTING	使用create procedure
CREATE USER	使用create user,drop user,rename user和revoke,privileges
CREATE VIEW	使用create view
DELETE	使用delete
DROP	使用drop table
EXECUTE	使用call和存储过程
FILE	使用select into outfile和load data infile

GNU/Linux-MariaDB

安全管理

权限	说明
GRANT OPTION	使用grant和revoke
INDEX	使用create index和drop index
INSERT	使用insert
LOCK TABLES	使用lock tables
PROCESS	使用show full processlist
RELORD	使用flush
RELICATION CLIENT	访问本地服务器
RELICATION SLAVE	有复制从属使用
SELECT	使用select
SHOW DATABASES	使用show databases
SHOW VIEW	使用show create view
SHUTDOWN	使用mysqladmin shutdomn

GNU/Linux-MariaDB

安全管理

权限	说明
SUPER	使用change master,kill,logs,purge,master和set globle,还允许mysqladmin测试登录
UPDATE	使用Update
USAGE	没有访问权限