[Top](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN201904/ENGINEER/DAY03/CASE/01/index.html" \l "page_top_case)

# NSD ENGINEER DAY03

1. [案例1：发布iSCSI网络磁盘](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN201904/ENGINEER/DAY03/CASE/01/index.html" \l "case1)
2. [案例2：搭建mariadb数据库系统](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN201904/ENGINEER/DAY03/CASE/01/index.html" \l "case2)
3. [案例3：配置一个数据库](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN201904/ENGINEER/DAY03/CASE/01/index.html" \l "case3)
4. [案例4：使用数据库查询](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN201904/ENGINEER/DAY03/CASE/01/index.html" \l "case4)

## 1 案例1：发布iSCSI网络磁盘

### 1.1 问题

本例要求配置 server0 提供 iSCSI 服务，要求如下：

1. 磁盘名为iqn.2016-02.com.example:server0
2. 服务端口为 3260
3. 使用 iscsi\_store 作其后端卷，其大小为 3GiB
4. 此磁盘服务只能被 desktop0.example.com 访问

方案

首先利用parted或fdisk工具划分分区

安装targetcli服务端软件，执行targetcli命令进行配置

1. 创建后端存储
2. 创建iqn对象
3. 将iqn与后端存储绑定
4. 授权客户端的iqn
5. 指定监听地址及本机端口

### 1.2 步骤

实现此案例需要按照如下步骤进行。

步骤一：准备磁盘空间，划分分区

1）利用parted命令划分/dev/vdb

1. [root@server0 /]# parted /dev/vdb
2. (parted) mktable gpt //指定分区模式为gpt
3. (parted) mkpart primary ext4 0 3GiB
4. 警告: The resulting partition is not properly aligned for best performance.
5. 忽略/Ignore/放弃/Cancel? Ignore
6. (parted) print
7. ……
8. Number Start End Size File system Name 标志
9. 1 17.4kB 3221MB 3221MB primary
10. (parted) quit

2）命令行查看分区信息

1. [root@server0 /]# lsblk
2. NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
3. ……
4. └─vdb1 252:17 0 3G 0 part ln -s

3）查看分区设备文件

1. [root@server0 /]# ls /dev/vdb1
2. /dev/vdb1

步骤二：安装与配置iSCSI磁盘

1）安装软件包targetcli

1. [root@server0 ~]# yum -y install targetcli

2）命令行运行targetcli配置iSCSI磁盘

1. [root@server0 /]# targetcli
2. /> backstores/block create iscsi\_store /dev/vdb1 //创建后端存储
3. /> iscsi/ create iqn.2016-02.com.example:server0 //创建iqn对象
4. /> iscsi/iqn.2016-02.com.example:server0/tpg1/luns create /backstores/block/iscsi\_store //将iqn对象与后端存储绑定
5. /> iscsi/iqn.2016-02.com.example:server0/tpg1/acls create iqn.2016-02.com.example:desktop0 //设置客户端的iqn
6. /> iscsi/iqn.2016-02.com.example:server0/tpg1/portals/ create 0.0.0.0 3260 //开启本地监听的IP地址与端口
7. /> exit

3）启动target服务，并设置为开机自启

1. [root@server0 /]# systemctl restart target
2. [root@server0 /]# systemctl enable target
3. Created symlink from /etc/systemd/system/multi-user.target.wants/target.service to /usr/lib/systemd/system/target.service.

## 2 案例2：搭建mariadb数据库系统

### 2.1 问题

本例要求在虚拟机server0上安装 MariaDB 数据库系统：

1. 安装 mariadb-server、mariadb 软件包
2. 启动 mariadb 服务，并确认监听状态

然后在客户端访问此数据库服务：

1. 使用 mysql 命令访问本机的数据库服务，用户名为 root，密码为空
2. 执行 SHOW DATABASES; 指令列出有哪些库
3. 退出 mysql 交互界面

### 2.2 方案

数据库表及相关软件的基本知识：

* 数据（记录）：用来表示一个事物（实体）的一些信息（属性）的文字/图片文件等，例如字符串“：tedu.cn”
* 数据表：存放很多条数据记录的容器，例如学员联系信息表、学员月考成绩表
* 数据表的每一行：存放一条记录
* 数据表的每一列/字段：很多个事物的同一个属性
* 数据库：存放很多个相互关联的表格的容器，例如NSD1609学员档案库
* 数据库管理系统（DBMS）：用来管理（创建库/添加/查询/删除/授权等）数据库信息的软件平台

MariaDB服务端：软件包mariadb-server、系统服务mariadb

MariaDB客户端：软件包mariadb、管理工具mysql

MariaDB服务端配置文件：/etc/my.cnf

传输协议及端口：TCP 3306

mysql命令的简单用法：

1. mysql [-u用户名] [-p[密码]]

### 2.3 步骤

实现此案例需要按照如下步骤进行。

步骤一：搭建MariaDB数据库服务器

1）安装软件包mariadb-server、mariadb

1. [root@server0 ~]# yum -y install mariadb-server mariadb
2. .. ..

2）启动系统服务mariadb，并设置开机自启

1. [root@server0 ~]# systemctl restart mariadb
2. [root@server0 ~]# systemctl enable mariadb
3. ln -s '/usr/lib/systemd/system/mariadb.service' '/etc/systemd/system/multi-user.target.wants/mariadb.service'

3）检查监听状态

1. [root@server0 ~]# netstat -antpu | grep :3306
2. tcp 0 0 0.0.0.0:3306 0.0.0.0:\* LISTEN 2922/mysqld

步骤二：访问本机的MariaDB数据库系统

1）以用户root连接本机的mariadb（或mysqld）数据库服务

1. [root@server0 ~]# mysql -uroot
2. Welcome to the MariaDB monitor. Commands end with ; or \g.
3. Your MariaDB connection id is 3
4. Server version: 5.5.35-MariaDB MariaDB Server
5. Copyright (c) 2000, 2013, Oracle, Monty Program Ab and others.
6. Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
7. MariaDB [(none)]>

2）查看当前数据库系统内有哪些库

1. MariaDB [(none)]> SHOW DATABASES;
2. +--------------------+
3. | Database |
4. +--------------------+
5. | information\_schema |
6. | mysql |
7. | performance\_schema |
8. | test |
9. +--------------------+
10. 4 rows in set (0.00 sec)

3）退出操作环境

1. MariaDB [(none)]> QUIT
2. Bye
3. [root@server0 ~]#

## 3 案例3：配置一个数据库

### 3.1 问题

本例要求在虚拟机server0上部署 MariaDB 数据库，具体要求如下：

1. 此数据库系统只能被 localhost 访问
2. 新建一个数据库名为 Contacts，其中应该包含来自数据库复制的内容，复制文件的 URL 为：http://classroom/pub/materials/users.sql
3. 除了 root 用户，此数据库只能被用户 Raikon 查询，此用户的密码为atenorth
4. root用户的密码为 atenorth

### 3.2 方案

为数据库账号修改密码：

1. mysqladmin [-u用户名] [-p[旧密码]] password '新密码'

导入/恢复到数据库：

1. mysql [-u用户名] [-p[密码]] 数据库名 < 备份文件.sql

为数据库用户授权/撤销权限：

1. grant 权限1,权限2... on 库名.表名 to 用户名@客户机地址 identified by '密码';
2. revoke 权限1,权限2... on 库名.表名 from 用户名@客户机地址;

### 3.3 步骤

实现此案例需要按照如下步骤进行。

步骤一：禁止mariadb服务提供网络监听（只服务于本机）

1）修改配置文件

1. [root@server0 ~]# vim /etc/my.cnf
2. [mysqld]
3. skip-networking                                         //跳过网络

2）重启mariadb服务

1. [root@server0 ~]# systemctl restart mariadb             //重启服务

3）确认结果

1. [root@server0 ~]# netstat -anptu | grep :3306         //已经不提供端口监听
2. [root@server0 ~]# pgrep -l mysqld                     //但进程仍在
3. 3127 mysqld\_safe
4. 3297 mysqld

步骤二：配置数据库管理密码

1）使用mysqladmin为用户root设置密码

原管理账号root的密码为空，因此无需验证旧密码：

1. [root@server0 ~]# mysqladmin -u root password 'atenorth'

2）验证新密码是否可用

root使用空密码从本机连接将会失败：

1. [root@server0 ~]# mysql -uroot
2. ERROR 1045 (28000): Access denied for user 'root'@'localhost' (using password: NO)

必须指定正确的新密码才能连接成功：

1. [root@server0 ~]# mysql -uroot -patenorth
2. Welcome to the MariaDB monitor. Commands end with ; or \g.
3. Your MariaDB connection id is 4
4. Server version: 5.5.35-MariaDB MariaDB Server
5. Copyright (c) 2000, 2013, Oracle, Monty Program Ab and others.
6. Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
7. .. ..

步骤三：建Contacts库并导入备份数据

1）创建新库Contacts，并退出操作环境

1. MariaDB [(none)]> CREATE DATABASE Contacts;
2. Query OK, 1 row affected (0.00 sec)
3. MariaDB [(none)]> QUIT
4. Bye

2）下载指定的数据库备份

1. [root@server0 ~]# wget http://classroom.example.com/pub/materials/users.sql
2. --2016-11-26 19:00:37-- http://classroom.example.com/pub/materials/users.sql
3. Resolving classroom.example.com (classroom.example.com)... 172.25.254.254
4. Connecting to classroom.example.com (classroom.example.com)|172.25.254.254|:80... connected.
5. HTTP request sent, awaiting response... 200 OK
6. Length: 2634 (2.6K) [application/sql]
7. Saving to: ‘users.sql’
8. 100%[=====================>] 2,634 --.-K/s in 0s
9. 2016-11-26 19:00:37 (269 MB/s) - ‘users.sql’ saved [2634/2634]
10. [root@server0 ~]# ls -lh users.sql                     //确认下载的文件
11. -rw-r--r--. 1 root root 2.6K Mar 31 2016 users.sql

3）导入数据库

1. [root@server0 ~]# mysql -uroot -patenorth Contacts < users.sql

4）重新连入操作环境，确认导入结果

1. [root@server0 ~]# mysql -uroot -patenorth
2. .. ..
3. MariaDB [(none)]> USE Contacts;                     //使用指定库
4. Database changed
5. MariaDB [Contacts]> SHOW TABLES;                 //列出有哪些表
6. +--------------------+
7. | Tables\_in\_Contacts |
8. +--------------------+
9. | base |
10. | location |
11. +--------------------+
12. 2 rows in set (0.00 sec)

步骤四：为Contacts库授权

1）允许用户Raikon从本机访问，具有查询权限，密码为atenorth

1. MariaDB [Contacts]> GRANT select ON Contacts.\* TO Raikon@localhost IDENTIFIED BY 'atenorth';
2. Query OK, 0 rows affected (0.00 sec)

2）退出操作环境

1. MariaDB [Contacts]> QUIT
2. Bye
3. [root@server0 ~]#

## 4 案例4：使用数据库查询

### 4.1 问题

本例要求配置MariaDB数据库，完成以下任务：

1. 禁止空密码root用户访问mariadb数据库
2. 在系统server0上使用数据库Contacts，通过SQL查询回答下列问题：密码是solicitous的人的名字？有多少人的姓名是Barbara同时居住在 Sunnyvale？

### 4.2 方案

表记录增删改查：

1. insert into [库名.]表名 values(值1,值2,值3);
2. delete from [库名.]表名 where ...;
3. update [库名.]表名 set 字段名=字段值 where ....;
4. select 字段列表 from [库名.]表名 where 字段名1=值 and|or 字段名2=值;

统计查询结果的数量：

1. select count(\*) from [库名.]表名 where .. ..;

### 4.3 步骤

实现此案例需要按照如下步骤进行。

步骤一：清理空密码root用户

1）确认空密码root用户记录

MariaDB服务端默认的mysql库user表保存了用户授权记录。

使用DESC指令查看表结构，以便了解相关字段名：

1. MariaDB [(none)]> DESC mysql.user;
2. +------------------------+-----------------------------------+------+-----+---------+-------+
3. | Field | Type | Null | Key | Default | Extra |
4. +------------------------+-----------------------------------+------+-----+---------+-------+
5. | Host | char(60) | NO | PRI | | |
6. | User | char(16) | NO | PRI | | |
7. | Password | char(41) | NO | | | |

列出user表中的Host、User、Password字段，限定密码为空的root用户：

1. MariaDB [(none)]> SELECT Host,User,Password FROM mysql.user WHERE User='root' AND Password='';
2. +---------------------+------+----------+
3. | Host | User | Password |
4. +---------------------+------+----------+
5. | server0.example.com | root | |
6. | 127.0.0.1 | root | |
7. | ::1 | root | |
8. +---------------------+------+----------+
9. 3 rows in set (0.00 sec)
10. MariaDB [(none)]>

2）删除空密码root用户记录

使用DELETE指令删除掉需要清除的授权记录：

1. MariaDB [(none)]> DELETE FROM mysql.user WHERE User='root' AND Password='';
2. Query OK, 3 rows affected (0.00 sec)

再次查询，确认删除结果：

1. MariaDB [(none)]> SELECT Host,User,Password FROM mysql.user WHERE User='root' AND Password='';
2. Empty set (0.00 sec)

步骤二：按条件查询表记录

1）按单个条件查询

找出密码是solicitous的人的名字？

1. MariaDB [(none)]> SELECT name FROM Contacts.base WHERE Password='solicitous';
2. +-------+
3. | name |
4. +-------+
5. | James |
6. +-------+
7. 1 row in set (0.00 sec)

2）按多个条件在关联的两张表中查询

有多少人的姓名是Barbara同时居住在 Sunnyvale？

1. MariaDB [(none)]> USE Contacts;
2. .. ..
3. Database changed
4. MariaDB [Contacts]> SELECT COUNT(\*) FROM base,location WHERE base.name='Barbara' AND location.city='Sunnyvale' AND base.id=location.id;
5. +----------+
6. | COUNT(\*) |
7. +----------+
8. | 1 |
9. +----------+
10. 1 row in set (0.00 sec)