

NSD ENGINEER DAY05

1. [案例1：发布iSCSI网络磁盘](#)
2. [案例2：访问iSCSI网络磁盘](#)
3. [案例3：搭建mariadb数据库系统](#)
4. [案例4：配置一个数据库](#)
5. [案例5：使用数据库查询](#)

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

[Top](#)

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

```
01. [ root@server0 /] # parted /dev /vdb
02. ( parted) mktable gpt           //指定分区模式为gpt
03. ( parted) mkpart primary ext4 0 3GiB
04. 警告: The resulting partition is not properly aligned for best performance.
05. 忽略/ignore/放弃/Cancel? ignore
06. ( parted) print
07. .....
08. Number  Start   End     Size   File system  Name    标志
09.   1      17.4kB 3221MB 3221MB                primary
10.
11. ( parted) quit
```

2) 命令行查看分区信息

```
01. [ root@server0 /] # lsblk
02. NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
03. .....
04. └─vdb1       252:17  0   3G  0 part ln -s
```

3) 查看分区设备文件

```
01. [ root@server0 /] # ls /dev /vdb1
```

[Top](#)

02. /dev/vdb1

步骤二：安装与配置iSCSI磁盘

1) 安装软件包targetcli

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

2) 命令行运行targetcli配置iSCSI磁盘

```
01.     [root@server0 /]# targetcli
02.     /> backstores/block create iscsi_store /dev/vdb1     //创建后端存储
03.     /> iscsi/ create iqn.2016-02.com.example:server0     //创建iqn对象
04.     /> 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
05.     /> iscsi/iqn.2016-02.com.example:server0/tpg1/acls create iqn.2016-02.com.example:desktop0     //设置客户端的iqn
06.     /> 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
07.     /> exit
```

3) 启动target服务，并设置为开机自启

```
01.     [root@server0 /]# systemctl restart target
02.     [root@server0 /]# systemctl enable target
03.     Created symlink from /etc/systemd/system/multi-user.target.wants/target.service to /usr/lib/systemd/system/target.service.
```

[Top](#)

2 案例2：访问iSCSI网络磁盘

2.1 问题

配置 desktop0 使用 iSCSI 磁盘，要求如下：

1. 目标磁盘组为 iqn.2016-02.com.example:server0
2. 此 iSCSI 设备在系统启动的期间自动加载
3. 此 iSCSI 块设备上包含一个大小为 2100MiB 的分区，并将其格式化为 ext4 文件系统
4. 此分区应该挂载到 /mnt/data 上，同时在系统启动的期间自动挂载

方案

首先按住客户端软件iscsi-initiator-utils

1. 设置客户端iqn
2. 重启iscsid服务刷新客户端iqn
3. 发现服务端iSCSI网络磁盘
4. 使用iSCSI共享磁盘空间

2.2 步骤

实现此案例需要按照如下步骤进行。

步骤一：安装客户端软件iscsi-initiator-utils

1) 利用yum安装软件

- ```
01. [root@desktop0 /] # yum repolist //生成yum仓库信息缓存
02. [root@desktop0 /] # yum -y install iscsi-initiator-utils.i686 //在生成yum仓库信息缓存后，软件包名可以补全 (tab)
```

#### 步骤二：设置客户端iqn名称

1 ) 修改配置文件

[Top](#)

01. [ root@desktop0 /] # vim /etc/iscsi/initiatorname.iscsi
02. InitiatorName=iqn.2016-02.com.example:desktop0

## 2) 重启iscsid服务刷新iqn名称

01. [ root@desktop0 /] # systemctl restart iscsid
02. Warning: iscsid.service changed on disk. Run 'systemctl daemon-reload' to reload units.
03. [ root@desktop0 /] # systemctl daemon-reload
04. [ root@desktop0 /] # systemctl restart iscsid
05. [ root@desktop0 /] # systemctl enable iscsid
06. Created symlink from /etc/systemd/system/multi-user.target.wants/iscsid.service to /usr/lib/systemd/system/iscsid.service.

## 步骤三：发现服务端iSCSI网络磁盘

### 1) 执行发现iSCSI网络磁盘命令

01. [ root@desktop0 /] # iscsiadm --mode discoverydb --type sendtargets --portal 172.25.0.11 --discover //此命令较长可以通过man iscsiadm查:
- 02.
03. 172.25.0.11:3260,1 iqn.2016-02.com.example:server0



### 2) 重启iscsi服务加载iSCSI网络磁盘

01. [ root@desktop0 /] # systemctl restart iscsi
02. [ root@desktop0 /] # lsblk

[Top](#)

```

03. NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
04. sda 8:0 0 3G 0 disk
05. [root@desktop0 /] # systemctl enable iscsi

```

## 步骤四：客户端使用iSCSI网络磁盘

### 1) 划分分区

```

01. [root@desktop0 /] # parted /dev /sda
02. (parted) mktable gpt
03. (parted) mkpart primary ext4 0 2100MB
04. 警告: The resulting partition is not properly aligned for best performance.
05. 忽略/Ignore/放弃/Cancel? Ignore
06. (parted) print
07. (parted) exit
08. [root@desktop0 /] # partprobe /dev /sda //刷新分区表
09. [root@desktop0 /] # lsblk
10. NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
11. sda 8:0 0 3G 0 disk
12. └─sda1 8:1 0 2.1G 0 part

```

### 2) 格式化使用分区，利用UUID实现开机自动挂载

```

01. [root@desktop0 /] # mkfs.ext4 /dev /sda1
02. [root@desktop0 /] # blkid /dev /sda1 #查看分区UUID值
03. /dev /sda1: UUID="402d1304- 5781- 4349- 8c43- ba45eba8b427" TYPE="ext4" PARTLABEL="primary" PARTUUID="2e41f84d- 5123- 4511- aaca- 7e7790

```

[Top](#)

```
04. [root@desktop0 /] # mkdir /mnt/data
05. [root@desktop0 /] # vim /etc/fstab
06. UUID="402d1304-5781-4349-8c43-ba45eba8b427" /mnt/data ext4 defaults,_netdev 0 0
07. [root@desktop0 /] # mount -a
08. [root@desktop0 /] # df -h
09. 文件系统 容量 已用 可用 已用% 挂载点
10.
11. /dev/sda1 2.0G 6.2M 1.9G 1% /mnt/data
```

## 3 案例3：搭建mariadb数据库系统

### 3.1 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

### 3.2 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

[Top](#)

- 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 :

```
01. my sql [- u用户名] [- p[密码]]
```

### 3.3 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

```
01. [root@server0 ~] # yum -y install mariadb-server mariadb
02.
```

2 ) 启动系统服务mariadb , 并设置开机自启

```
01. [root@server0 ~] # systemctl restart mariadb
02. [root@server0 ~] # systemctl enable mariadb
03. ln -s '/usr/lib/systemd/system/mariadb.service' '/etc/systemd/system/multi-user.target.wants/mariadb.service'
```

[Top](#)



### 3 ) 检查监听状态

```
01. [root@server0 ~]# netstat -antpu | grep :3306
02. tcp 0 0 0.0.0.0:3306 0.0.0.0:* LISTEN 2922/mysql
```

### 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

```
01. [root@server0 ~]# mysql -uroot
02. Welcome to the MariaDB monitor. Commands end with ; or \g.
03. Your MariaDB connection id is 3
04. Server version: 5.5.35-MariaDB MariaDB Server
05.
06. Copyright (c) 2000, 2013, Oracle, Monty Program Ab and others.
07.
08. Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
09.
10. MariaDB [(none)]>
```

#### 2 ) 查看当前数据库系统内有哪些库

```
01. MariaDB [(none)]> SHOW DATABASES;
02. +-----+
03. | Database |
04. +-----+
```

[Top](#)

```
05. | information_schema |
06. | my sql |
07. | performance_schema |
08. | test |
09. +-----+
10. 4 rows in set (0.00 sec)
```

### 3) 退出操作环境

```
01. MariaDB [(none)]> QUIT
02. Bye
03. [root@server0 ~]#
```

## 4 案例4：配置一个数据库

### 4.1 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

### 4.2 方案

为数据库账号修改密码：

[Top](#)

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

导入/恢复到数据库：

```
01 my sql [- 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：

```
01 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';
02 revoke 权限1,权限2... on 库名.表名 from 用户名@客户机地址;
```

## 4.3 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

```
01 [root@server0 ~] # vim /etc/my.cnf
02 [mysqld]
03 skip-networking //跳过网络
```

2 ) 重启mariadb服务

```
01 [root@server0 ~] # systemctl restart mariadb //重启服务
```

[Top](#)

### 3 ) 确认结果

```
01. [root@server0 ~]# netstat -anptu | grep :3306 //已经不提供端口监听
02. [root@server0 ~]# pgrep -l mysqld //但进程仍在
03. 3127 mysqld_safe
04. 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：

```
01. [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：

```
01. [root@server0 ~]# mysql -uroot
02. 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：

```
01. [root@server0 ~]# mysql -uroot -patenorth
```

[Top](#)

02. Welcome to the MariaDB monitor. Commands end with ; or \g.
03. Your MariaDB connection id is 4
04. Server version: 5.5.35-MariaDB MariaDB Server
- 05.
06. Copyright (c) 2000, 2013, Oracle, Monty Program Ab and others.
- 07.
08. Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
- 09.
10. ...

### 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

01. MariaDB [(none)]> CREATE DATABASE Contacts;
02. Query OK, 1 row affected (0.00 sec)
- 03.
04. MariaDB [(none)]> QUIT
05. Bye

#### 2) 下载指定的数据库备份

01. [root@server0 ~]# wget http://classroom.example.com/pub/materials/users.sql
02. -- 2016-11-26 19:00:37 -- http://classroom.example.com/pub/materials/users.sql
03. Resolving classroom.example.com (classroom.example.com)... 172.25.254.254
04. Connecting to classroom.example.com (classroom.example.com) | 172.25.254.254 | 80... connected.

[Top](#)

```

05. HTTP request sent, awaiting response... 200 OK
06. Length: 2634 (2.6K) [application/sql]
07. Saving to: ' users.sql'
08.
09. 100%[=====>] 2,634 --.- K/s in 0s
10.
11. 2016-11-26 19:00:37 (269 MB/s) - ' users.sql' saved [2634/2634]
12.
13. [root@server0 ~] # ls -lh users.sql //确认下载的文件
14. -rw-r--r--. 1 root root 2.6K Mar 31 2016 users.sql

```

### 3 ) 导入数据库

```

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

```

### 4 ) 重新连入操作环境，确认导入结果

```

01. [root@server0 ~] # mysql -uroot -patenorth
02. ...
03. MariaDB [(none)]> USE Contacts; //使用指定库
04. Database changed
05.
06. MariaDB [Contacts]> SHOW TABLES; //列出有哪些表
07. +-----+
08. | Tables_in_Contacts |
09. +-----+

```

[Top](#)

```
10. | base |
11. | location |
12. +-----+
13. 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

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

2) 退出操作环境

```
01. MariaDB [Contacts] > QUIT
02. Bye
03. [root@server0 ~] #
```

## 5 案例5：使用数据库查询

### 5.1 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？

[Top](#)

## 5.2 方案

表记录增删改查：

01. insert into [库名.]表名 values(值1,值2,值3);
02. delete from [库名.]表名 where ...;
03. update [库名.]表名 set 字段名=字段值 where ...;
04. 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：

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

## 5.3 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：

01. MariaDB [(none)]> DESC mysql.user;
02. +-----+-----+-----+-----+-----+-----+
03. | Field | Type | Null | Key | Default | Extra |
04. +-----+-----+-----+-----+-----+-----+

[Top](#)



|     |          |           |    |     |  |  |  |
|-----|----------|-----------|----|-----|--|--|--|
| 05. | Host     | char( 60) | NO | PRI |  |  |  |
| 06. | User     | char( 16) | NO | PRI |  |  |  |
| 07. | 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：

```

01. MariaDB [(none)]> SELECT Host,User,Password FROM mysql.user WHERE User='root' AND Password='';
02. +-----+-----+-----+
03. | Host | User | Password |
04. +-----+-----+-----+
05. | server0.example.com | root | |
06. | 127.0.0.1 | root | |
07. | ::1 | root | |
08. +-----+-----+-----+
09. 3 rows in set (0.00 sec)
10.
11. 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：

```

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

```

[Top](#)

再次查询，确认删除结果：

01. MariaDB [(none)]> SELECT Host,User,Password FROM mysql.user WHERE User='root' AND Password='';
02. 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？

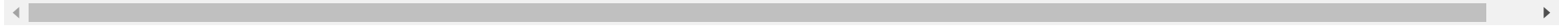
01. MariaDB [(none)]> SELECT name FROM Contacts.base WHERE Password='solicitous';
02. +-----+
03. | name |
04. +-----+
05. | James |
06. +-----+
07. 1 row in set (0.00 sec)

### 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 ？

01. MariaDB [(none)]> USE Contacts;
02. ....
03. Database changed
04. MariaDB [Contacts]> SELECT COUNT(\*) FROM base,location WHERE base.name='Barbara' AND location.city='Sunnyvale' AND base.id=location.id;
05. +-----+
06. | COUNT(\*) |

- 07. +-----+
- 08. | 1|
- 09. +-----+
- 10. 1 row in set (0.00 sec)

[Top](#)