

## MySQL8 InnoDB ReplicaSet 主从数据库安装教学指导书

【编写说明：聂耿青于 2021 年编写第一版；2023 年经李宇修订为第二版，主要修订内容：改为由两个沙漏实例构建副本集(ReplicaSet)，以方便删除重建，多次实验】

### 【预备工作】

确保数据库版本在 8.0.19 及以上，否则请升级版本：

如果 MySQL8 低于 19 版，可用如下命令升级：

```
yum update mysql-server [or yum update]
```

```
systemctl restart mysqld
```

一、安装至少两个 mysql 实例或者先安装一个，再通过 mysql shell 创建实验用的两个沙漏(sandbox)实例（仅用于测试实验）

192.168.X.2

centos7.x

```
mysql-community-server-8.0.19-1.el7.x86_64,mysql-router-community-
```

```
8.0.19-1.el7.x86_64
```

```
mysqlnode1, mysqlmonitor
```

192.168.X.3 (option or sandbox instance replace)

centos7.x

```
mysql-community-server-8.0.19-1.el7.x86_64,mysql-router-community-
```

```
8.0.19-1.el7.x86_64
```

```
mysqlnode2
```

```
#cd /software
```

```
wget http://mirrors.ustc.edu.cn/mysql-repo/mysql80-community-release-el7-3.noarch.rpm
```

```
#yum localinstall mysql80-community-release-el7-3.noarch.rpm
```

```
#yum install mysql-community-server
```

```
firewall-cmd --zone=public --permanent --add-port=3306/tcp
```

```
firewall-cmd --zone=public --permanent --add-port=3307/tcp
```

```
firewall-cmd --zone=public --permanent --add-port=3308/tcp
```

```
firewall-cmd --zone=public --permanent --add-port=3309/tcp
```

```
firewall-cmd --reload
```

```
systemctl start mysqld
```

```
grep 'temporary password' /var/log/mysqld.log
```

```
mysql -uroot -p
```

```
ALTER USER 'root'@'localhost' IDENTIFIED BY 'abc123!Test';
```

```
create user ngq@"%" identified by 'abc123!Test';
```

```
GRANT ALL privileges ON *.* TO "ngq"@"%";
```

```
FLUSH PRIVILEGES;
```

## 二、安装 mysql shell (Installing MySQL Shell):

Linux:

```
yum install mysql-shell
```

or

```
yum localinstall mysql_shell*.rpm
```

Windows: 解压即可

<https://dev.mysql.com/doc/mysql-shell/8.0/en/mysql-shell-install-windows-quick.html>

运行 mysqlsh

```
\connect ngq@192.168.0.21:3306
```

```
or mysqlsh [--uri] root@localhost:3306
```

```
or mysqlsh -uroot -p
```

<code>\status</code>	<code>\s</code>	显示当前的 MySQL Shell 状态。
<code>\js</code>		将执行模式切换为 JavaScript。
<code>\py</code>		将执行模式切换为 Python。
<code>\sql</code>		将执行模式切换为 SQL。

如果只有一个实例，用于测试的，可创建一个 sandbox 实例:

```
localhost:3306 ssl JS > dba.deploySandboxInstance(3307);
```

```
localhost:3306 ssl JS > dba.deploySandboxInstance(3308);
```

```
(可选) localhost:3306 ssl JS > dba.deploySandboxInstance(3309);
```

Use `shell.connect('root@localhost:3307')` to connect to the instance.

By default the sandbox is created in a directory named `$HOME/mysql-sandboxes/port` on Unix systems.

- To stop a sandbox instance use `dba.stopSandboxInstance(instance)`. This stops the instance gracefully, unlike `dba.killSandboxInstance(instance)`.
- To start a sandbox instance use `dba.startSandboxInstance(instance)`.
- To kill a sandbox instance use `dba.killSandboxInstance(instance)`. This stops the instance without gracefully stopping it and is useful in simulating unexpected halts.
- To delete a sandbox instance use `dba.deleteSandboxInstance(instance)`. This completely removes the sandbox instance from your file system.

## Configuring InnoDB ReplicaSet Instances

运行 mysqlsh

```
\connect ngq@192.168.0.21:3307  
or mysqlsh [--uri] root@localhost:3307
```

The administrator account must have the same user name and password across all instances of the same cluster or replica set.

要注意生成的实例名称（主机名），后面要用到，须一致。

```
MySQL 127.0.0.1:3307 ssl JS >  
dba.configureReplicaSetInstance('root@localhost:3307',{clusterAdmin: "rsadmin@localhost  
"});  
rsadmin@localhost 一个可用于全部节点的账户,  
如'rsadmin'@'%', 'rsadmin'@'192.168.0.%'  
dba.configureReplicaSetInstance('root@rs-1:3307', {clusterAdmin:  
"'rsadmin'@rs-%'", clusterAdminPassword: 'pwd'});  
The instance 'localhost.localdomain:3307' was configured to be used in an InnoDB ReplicaSet.
```

## Creating an InnoDB ReplicaSet

```
MySQL 127.0.0.1:3307 ssl JS > var rs = dba.createReplicaSet("ngqreplicaset");  
>rs.status()
```

```
Show the primary is { "localhost.localdomain:3307": {  
    "address": "localhost.localdomain:3307",  
    "instanceRole": "PRIMARY",  
    "mode": "R/W",  
    "status": "ONLINE" }
```

```
MySQL 127.0.0.1:3307 ssl JS > rs.addInstance('root@localhost:3308')
```

（可选）还可以再加 MySQL 127.0.0.1:3307 ssl JS >

```
rs.addInstance('root@localhost:3309')
```

Or rs.addInstance('technode:3308') 不指定用户

Select default Clone

```
[  
rs.addInstance('rsadmin@localhost3308')  
or  
rs.addInstance("admin@rs2:3308");  
]
```

The instance '127.0.0.1:3308' was added to the replicaset and is replicating from 127.0.0.1:3307.

```
MySQL 127.0.0.1:3307 ssl JS > rs.status()
```

```
{  
  "replicaSet": {
```

```
"name": "liyurs",
"primary": "127.0.0.1:3307",
"status": "AVAILABLE",
"statusText": "All instances available.",
"topology": {
  "127.0.0.1:3307": {
    "address": "127.0.0.1:3307",
    "instanceRole": "PRIMARY",
    "mode": "R/W",
    "status": "ONLINE"
  },
  "127.0.0.1:3308": {
    "address": "127.0.0.1:3308",
    "instanceRole": "SECONDARY",
    "mode": "R/O",
    "replication": {
      "applierStatus": "APPLIED_ALL",
      "applierThreadState": "Waiting for an event from Coordinator",
      "applierWorkerThreads": 4,
      "receiverStatus": "ON",
      "receiverThreadState": "Waiting for master to send event",
      "replicationLag": null
    },
    "status": "ONLINE"
  },
  "127.0.0.1:3309": {
    "address": "127.0.0.1:3309",
    "instanceRole": "SECONDARY",
    "mode": "R/O",
    "replication": {
      "applierStatus": "APPLIED_ALL",
      "applierThreadState": "Waiting for an event from Coordinator",
      "applierWorkerThreads": 4,
      "receiverStatus": "ON",
      "receiverThreadState": "Waiting for master to send event",
      "replicationLag": null
    },
    "status": "ONLINE"
  }
}
"client": {
  "type": "ASYNC"
}
```

客户端测试:

分别连接 3307 主, 3308 备, 3309 备

```
use nietest;
```

```
create table test ( id int not null auto_increment primary key,name varchar(50));
```

```
insert into test (name) values ('test name');
```

```
select * from information_schema.innodb_tables where name like 'nietest/%';
```

```
select * from nietest.test;
```

下一次连接需要输入接入集群:

```
var rs = dba.getReplicaSet()
```

```
rs.status()
```

```
show slave status\G
```

参考文档: [MySQL :: MySQL Shell 8.0 :: 9 MySQL InnoDB ReplicaSet](#)