

# MGR集群搭建攻略

【IT老齐】VIP 五大权益：

<https://www.itlaoqi.com/chapter.html?sid=124&cid=2584>

一、仅需498开通永久VIP，课程永久更新，永远没有二次收费

二、永久VIP（简称VIP），从Java多线程到分布式架构上百小时精品课，所有课程提供完整笔记课件，不说垃圾话，不做垃圾课

三、VIP，加老齐微信 itlaoqi001，学习问题微信1对1答疑解惑

四、VIP，1对1制定学习计划、优化简历、offer建议、职业规划

五、VIP，架构选型、技术方案，只要我有经验一定知无不言

PS：

创业不易，设置VIP也属实为了养家糊口，望兄弟们海涵。

希望通过我的经验能帮你有效提高收入，找到更好的工作。

-----  
IT老齐官网：www.itlaoqi.com

公众号：IT老齐

如有任何问题都欢迎加我微信 itlaoqi001  
-----

这份文档你早晚用的到，下载保存好，未来一定用的上！！

为这份文档老齐掉了不少头发，从早8点红（害羞）到晚10点14小时，兄弟们收藏投个币吧！

这份文档面向对MySQL有一定使用经验的童鞋，直接看文档是最快的。

如果你经验少，我还拍了片儿，人挺少的那种，目前正在剪辑，三天后上传。

文档下载地址详情留意介绍或评论部分

## 虚拟机环境准备

- 3台CentOS7系统虚拟机，4G内存2核CPU，10G硬盘
- CentOS7系统要求安装Yum与Vim，采用“服务器”模式安装已内置

## 单主MGR拓扑：

- 192.168.31.230-Master服务器 (hostname:n0)
- 192.168.31.231-Slave服务器1 (hostname:n1)
- 192.168.31.232-Slave服务器2 (hostname:n2)

## 系统环境脚本

开头结尾带 \* 号的注释说明下行代码需要根据环境进行调整

请将下面代码片段在每一台节点上运行

```
1 #更改IP地址脚本
2 cd /etc/sysconfig/network-scripts
3 sed -i 's/dhcp/static/g' ifcfg-ens33
4 sudo cat >> ifcfg-ens33 <<- 'EOF'
5
6 # *三台服务器都要运行这段脚本，依次将下面230/231/232*
7 IPADDR=192.168.31.230
8 GATEWAY=192.168.31.1
9 NETMASK=255.255.255.0
10 DNS1=114.114.114.114
11 DNS2=8.8.8.8
```

```
12 EOF
13
14 #重启网络将IP地址生效
15 service network restart
16
17 #将Hosts文件写入n0/n1/n2节点与IP对应关系，后面配置采用域名访问
18 sudo cat > /etc/hosts <<- 'EOF'
19 192.168.31.230 n0
20 192.168.31.231 n1
21 192.168.31.232 n2
22 EOF
23 #*修改当前主机的名字，230->n0，231->n1，232->n2*
24 hostnamectl set-hostname n0
25
26 #大坑1:
27 #临时关闭selinux防火墙，开启后端口将无法访问，开发环境可以这样用
28 #生产环境这块安全配置更麻烦点，咱们后面再说
29 setenforce 0
30
31 #放行端口，3306MySQL默认端口，24901MGR通信端口
32 firewall-cmd --zone=public --add-port=3306/tcp --permanent
33 firewall-cmd --zone=public --add-port=24901/tcp --permanent
34 firewall-cmd --reload
35
```

## 安装部署MySQL

我们采用MySQL官方YUM仓库方式安装，亲测下载速度11mb/s，非常快

### 安装MySQL官方 Yum 安装源

请将下面代码片段在每一台节点上运行

```
1 cd /home/
2 wget --no-check-certificate https://repo.mysql.com/mysql80-community-release-el7-5.noarch.rpm
3 yum localinstall -y mysql80-community-release-el7-5.noarch.rpm
4
```

### 安装MySQL 8.0.28

```
1 #自动安装MySQL 8.0.28
2 yum install -y mysql-community-server
3
```

### 调整配置文件

主从三个节点配置文件是不同的，请分别复制粘贴

#### n0-230-Master节点

```
1 #安装完成，追加mysql
2 sudo cat >> /etc/my.cnf <<- 'EOF'
3 #使用mysql_native_password密码策略，防止navicat连不上mysql8
4 default_authentication_plugin=mysql_native_password
5 #设置MySQL插件所在目录，因为MGR基于插件，所以必须设置插件路径
6 plugin_dir=/usr/lib64/mysql/plugin
7 #服务器编号，Master=1
8 server_id=1
9 #开启binlog的GTID模式
```

```

10 gtid_mode=ON
11 #开启后MySQL只允许能够保障事务安全，并且能够被日志记录的SQL语句被执行
12 enforce_gtid_consistency=ON
13 #关闭binlog校验
14 binlog_checksum=NONE
15
16 #定义用于事务期间哈希写入提取的算法，组复制模式下必须设置为 XXHASH64。
17 transaction_write_set_extraction=XXHASH64
18 #确定组复制恢复时是否应该应用 SSL，通常设置为“开”，但默认设置为“关”。
19 loose-group_replication_recovery_use_ssl=ON
20 #该服务器的实例所在复制组的名称，必须是有效的 UUID，所有节点必须相同。
21 loose-group_replication_group_name="bbbbbbbb-bbbb-cccc-dddd-eeeeeeeeeeee"
22 #确定服务器是否应该在服务器启动期间启动组复制。
23 loose-group_replication_start_on_boot=OFF
24
25 #大坑2:
26 #为复制组中其他的成员提供的网络地址，指定为“主机:端口”的格式化字符串。
27 #很多人想当然认为端口应该是3306，起始不然，MGR需要开启新端口24901同步交换
28 #所以这里不要写错，同时，前面我们配置了hosts文件做了主机名与IP的映射，这里直接写主机名即可
29 loose-group_replication_local_address="n0:24901"
30
31 #用于建立新成员到组的连接组成员列表。
32 #这个列表指定为由分隔号间隔的组成员网络地址列表，类似 host1:port1、host2:port2 的格式。
33 #同样采用n0~n2的主机名替代
34 loose-group_replication_group_seeds="n0:24901,n1:24901,n2:24901"
35
36 #配置此服务器为引导组，这个选项必须仅在一台服务器上设置，
37 #并且仅当第一次启动组或者重新启动整个组时。成功引导组启动后，将此选项设置为关闭。
38 loose-group_replication_bootstrap_group=OFF
39 EOF
40
41 #启动MySQL服务器
42 systemctl start mysqld
43

```

## n1-231-Slave节点1

从节点和主节点的配置变化部分我通过注释标出

```

1 sudo cat >> /etc/my.cnf <<- 'EOF'
2 default_authentication_plugin=mysql_native_password
3 plugin_dir=/usr/lib64/mysql/plugin
4
5 #设置唯一的服务器编号
6 server_id=2
7 gtid_mode=ON
8 enforce_gtid_consistency=ON
9 binlog_checksum=NONE
10
11 #这个参数决定primary节点到secondary节点的请求是否为基于 RSA 密钥对的密码交换所需的公钥
12 loose-group_replication_recovery_get_public_key=ON
13
14 loose-group_replication_recovery_use_ssl=ON
15 loose-group_replication_group_name="bbbbbbbb-bbbb-cccc-dddd-eeeeeeeeeeee"
16 loose-group_replication_start_on_boot=OFF

```

```

17
18 #设置本机地址n1:24901
19 loose-group_replication_local_address="n1:24901"
20 loose-group_replication_group_seeds="n0:24901,n1:24901,n2:24901"
21 loose-group_replication_bootstrap_group=OFF
22 EOF
23
24
25 #启动MySQL服务器
26 systemctl start mysqld
27
28

```

## n2-232-Slave节点2

从节点和主节点的配置变化部分我通过注释标出

```

1 sudo cat >> /etc/my.cnf <<- 'EOF'
2 default_authentication_plugin=mysql_native_password
3 plugin_dir=/usr/lib64/mysql/plugin
4
5 #设置唯一的服务器编号
6 server_id=3
7 gtid_mode=ON
8 enforce_gtid_consistency=ON
9 binlog_checksum=NONE
10
11 #这个参数决定primary节点到secondary节点的请求是否为基于 RSA 密钥对的密码交换所需的公钥
12 loose-group_replication_recovery_get_public_key=ON
13 loose-group_replication_recovery_use_ssl=ON
14 loose-group_replication_group_name="bbbbbbbb-bbbb-cccc-dddd-eeeeeeeeeeee"
15 loose-group_replication_start_on_boot=OFF
16
17 #设置本机地址n2:24901
18 loose-group_replication_local_address="n2:24901"
19 loose-group_replication_group_seeds="n0:24901,n1:24901,n2:24901"
20 loose-group_replication_bootstrap_group=OFF
21 EOF
22
23
24 #启动MySQL服务器
25 systemctl start mysqld
26

```

到这里所有MySQL配置已完成，服务器已启动

## 初始化MySQL数据库

请在每一台服务器上运行下面脚本

### 获取初始密码

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

运行结果：

[Note] [MY-010454] [Server] A temporary password is generated for

root@localhost: =vjvyerPc5o=

## 连接到MySQL

每一次生成的初始密码都不同，进行灵活调整即可

```
1 mysql -uroot -p='vjvyerPc5o='
```

## 修改初始密码并设置主从同步账户

```
1 #修改root密码为asAS123456!
2 ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY 'asAS123456!';
3 #创建rpl_user账户，此账户用于实现主从数据同步
4 CREATE USER rpl_user@'%' IDENTIFIED BY 'asAS123456!';
5 #赋予主从同步权限
6 GRANT REPLICATION SLAVE ON *.* TO rpl_user@'%';
7
8 #创建一个远程连接用户，这个用户用在Navcate、JDBC登录的时候，直接用root登录不好
9 create user 'remote'@'%' identified with mysql_native_password by 'asAS123456!';
10 #为remote用户赋予所有数据库资源的访问权限，熟悉grant的小伙伴可以自己调整
11 grant all privileges on *.* to remote@'%';
12
13 #让刚才的修改生效
14 FLUSH PRIVILEGES;
15
16 #大坑3:
17 #删除已产生的Binlog
18 #一定要RESET MASTER，它会删除刚才已产生的Binlog
19 #因为刚才Binglog包含创建用户这种高权限操作，用于主从同步的rpl_user账户是没有权限执行的
20 #这就会导致RelayLog重放无法正确执行，导致从属服务器卡死在"RECEIVING"状态
21 #利用RESET MASTER删除这些无法执行的binlog，就没问题了
22 #就是这个卡了我3个小时 5555
23 RESET MASTER;
24
```

## 安装MGR插件

每一台节点，在MySQL提示符状态下，使用INSTALL PLUGIN安装MGR插件

```
1 INSTALL PLUGIN group_replication SONAME 'group_replication.so';
```

运行下面的SQL，看到结果出现PLUGIN\_STATUS=ACTIVE，说明安装成功了

```
1 SELECT * FROM INFORMATION_SCHEMA.PLUGINS WHERE PLUGIN_NAME = 'group_replication' \G;
2 ***** 1. row *****
3  PLUGIN_NAME: group_replication
4  PLUGIN_VERSION: 1.1
5  PLUGIN_STATUS: ACTIVE
6  PLUGIN_TYPE: GROUP REPLICATION
7  PLUGIN_TYPE_VERSION: 1.4
8  PLUGIN_LIBRARY: group_replication.so
9  PLUGIN_LIBRARY_VERSION: 1.10
10  PLUGIN_AUTHOR: Oracle Corporation
11  PLUGIN_DESCRIPTION: Group Replication (1.1.0)
12  PLUGIN_LICENSE: GPL
13  LOAD_OPTION: ON
14  1 row in set (0.00 sec)
```

## 进行组网数据同步

组网同步需要主从运行不同的脚本

### n0-230-Master节点

我们在 primary.cnf 配置文件中把 group\_replication\_bootstrap\_group 参数设置成 OFF，在 primary 服务器启动时并不会直接启动复制组，通过下面的命令动态的开启复制组是我们的集群更安全。

```
1 SET GLOBAL group_replication_bootstrap_group=ON;
2 START GROUP_REPLICATION;
3 SET GLOBAL group_replication_bootstrap_group=OFF;
```

运行成功，可以在linux命令提示符下

```
1 tail -f /var/log/mysqld.log
```

看到下面的数据代表Master启动成功

```
1 2022-03-04T03:15:12.930686Z 34 [System] [MY-011510] [Repl]
2 Plugin group_replication reported: 'This server is working as primary member.'
```

## n1-231-Slave节点1与n2-232-Slave节点2

从节点运行下面的命令

```
1 #指定主从账户与指定通信频道
2 CHANGE MASTER TO MASTER_USER="rpl_user", MASTER_PASSWORD="rpl_pass" FOR CHANNEL 'group_replication_recovery';
3 #开启组网数据同步
4 START GROUP_REPLICATION;
```

当两个从节点都运行完毕后，运行下面结果进行验证。

```
1 SELECT * FROM performance_schema.replication_group_members;
```

CHANNEL_NAME	MEMBER_ID	MEMBER_HOST	MEMBER_PORT	MEMBER_STATE	MEMBER_ROLE	MEMBER_VERSION	MEMBER_COMMUNICATION_STACK
group_replication_applier	e596599b-9b93-11ec-a071-000c29c169a0	n0	3306	ONLINE	PRIMARY	8.0.28	XCom
group_replication_applier	e78b2ad1-9b93-11ec-a2af-000c29485b60	n1	3306	ONLINE	SECONDARY	8.0.28	XCom
group_replication_applier	e9a2751e-9b93-11ec-83e9-000c29864b5a	n2	3306	ONLINE	SECONDARY	8.0.28	XCom

看到一主两从都是“ONLINE”状态，代表组网成功

## 踩坑分享：

### 坑1：网络不通

tail -f /var/log/mysqld.log 看日志

```
1 [ERROR] [MY-011735] [Repl] Plugin group_replication reported:
2 '[GCS] Error joining the group while waiting for the network layer to become ready.'
```

解决办法：

- ping n0、ping n1、ping n2确保网络畅通
- 所有节点千万不要忘记执行：setenforce 0

### 坑2：事务进度不一致

tail -f /var/log/mysqld.log 看日志

```
1 [Warning] Plugin group_replication reported:
2 This member has more executed transactions than those present in the group.
3 Local transactions: f9418f39-8f6a-11ea-8e60-005056841c10:1 > Group transactions: 6c0d784d-8f66-11ea-bd2e-00505684742a:1,
4 aaaaaaaaa-aaaa-aaaa-aaaa-aaaaaaaaaaaa:1-5'
```

解决办法：从属服务器停止，RESET binlog，重新同步

```
1 STOP GROUP_REPLICATION;
2 RESET MASTER;
3 CHANGE MASTER TO MASTER_USER="rpl_user", MASTER_PASSWORD="asAS123456!" FOR CHANNEL 'group_replication_recovery';
4 START GROUP_REPLICATION;
```

### 坑3：无法执行create user..

tail -f /var/log/mysql.log 看日志

```
1 如果Slave节点日志出现 Can not execute Query :“create user ...”
```

代表某一条SQL的执行权限已经超过rpl\_user用户的执行权限，会导致从节点同步状态卡死在"RECEIVING"状态，这是新人最容易犯的错误

解决办法：主节点执行REST MASTER，重新同步。这适用于刚开始的情况，运行中出现可以使用指令忽略这条无法同步的binlog