**1.搭建环境：**

OS：Linux.el6uek.x86\_64

MySQL：5.7.26

**2.模式：**

1 主 2 从

192.168.65.2 test2

192.168.65.3 test3

192.168.65.4 test4

**3.必要参数：**

gtid\_mode=on

enforce-gtid-consistency=on

binlog\_gtid\_simple\_recovery=1

log-slave-updates=1

binlog\_checksum=NONE

master\_info\_repository=TABLE

relay\_log\_info\_repository=TABLE

transaction\_write\_set\_extraction = MURMUR32

loose-group\_replication\_group\_name = '9ac06b4e-13aa-11e7-a62e-5254004347f9' #组名，有效地UUID即可

#loose-group\_replication\_ip\_whitelist = '127.0.0.1/8,192.168.65.0/24'

loose-group\_replication\_start\_on\_boot = OFF #开机自动启动

loose-group\_replication\_local\_address = '192.168.65.2:33061' #本机ip+port，端口不能跟mysql的端口一样

loose-group\_replication\_group\_seeds = '192.168.65.3:33061,192.168.65.4:33061,192.168.65.2:33061' # group 中所有机器list

loose-group\_replication\_bootstrap\_group = OFF # 初始化，默认关闭

loose-group\_replication\_single\_primary\_mode = true # 单主模式

loose-group\_replication\_enforce\_update\_everywhere\_checks = false #强制检查 ，多主模式时需要

**PS：除了MySQL安装所需的参数外，上面是搭建 Group Replication 必须要的参数；**

**4.建立复制用的账号（ALL）：**

mysql>grant replication slave on \*.\* to rpl@'%' identified by '123';

**5.安装插件（ALL）：**

mysql> INSTALL PLUGIN group\_replication SONAME 'group\_replication.so';

**6.检查是否安装完成（ALL）：**

mysql> show plugins;

如果出现 group\_replication 即为成功

**7.Change master**

mysql> CHANGE MASTER TO MASTER\_USER='rpl', MASTER\_PASSWORD='123' FOR CHANNEL 'group\_replication\_recovery';

**8.初始化**

mysql> set global group\_replication\_bootstrap\_group = on;

**9.start group\_replication**

mysql> start group\_replication;

**10.关闭初始化**

mysql> set global group\_replication\_bootstrap\_group = OFF;

**11.检查状态：**

mysql> select \* from performance\_schema.replication\_group\_members;

+---------------------------+--------------------------------------+--------------+-------------+--------------+

| CHANNEL\_NAME | MEMBER\_ID | MEMBER\_HOST | MEMBER\_PORT | MEMBER\_STATE |

+---------------------------+--------------------------------------+--------------+-------------+--------------+

| group\_replication\_applier | b5ab9bc1-7cb0-11e9-816c-000c29dbfcd1 | 192.168.65.2 | 3306 | ONLINE |

+---------------------------+--------------------------------------+--------------+-------------+--------------+

**12.从节点操作**

mysql> CHANGE MASTER TO MASTER\_USER='rpl', MASTER\_PASSWORD='123' FOR CHANNEL 'group\_replication\_recovery';

mysql> start group\_replication;

每个从节点都执行上面2个操作

**13.查看状态：**

mysql> select \* from performance\_schema.replication\_group\_members;

+---------------------------+--------------------------------------+--------------+-------------+--------------+

| CHANNEL\_NAME | MEMBER\_ID | MEMBER\_HOST | MEMBER\_PORT | MEMBER\_STATE |

+---------------------------+--------------------------------------+--------------+-------------+--------------+

| group\_replication\_applier | 82dfcffa-a939-11e9-bb5c-000c29f8b2a5 | test4 | 3308 | ONLINE |

| group\_replication\_applier | b5ab9bc1-7cb0-11e9-816c-000c29dbfcd1 | 192.168.65.2 | 3306 | ONLINE |

| group\_replication\_applier | f3755962-a189-11e9-a5f7-000c298f0ec5 | 192.168.65.3 | 3306 | ONLINE |

+---------------------------+--------------------------------------+--------------+-------------+--------------+

mysql> show global status like 'group%';

+----------------------------------+--------------------------------------+

| Variable\_name | Value |

+----------------------------------+--------------------------------------+

| group\_replication\_primary\_member | b5ab9bc1-7cb0-11e9-816c-000c29dbfcd1 |

+----------------------------------+--------------------------------------+

至此，MySQL Group Replication搭建完成

**14.slave 宕机测试：**

a.模拟192.168.65.3宕机

直接 Kill -9 mysql 的进程

b.master 查看状态

mysql> select \* from performance\_schema.replication\_group\_members;

+---------------------------+--------------------------------------+--------------+-------------+--------------+

| CHANNEL\_NAME | MEMBER\_ID | MEMBER\_HOST | MEMBER\_PORT | MEMBER\_STATE |

+---------------------------+--------------------------------------+--------------+-------------+--------------+

| group\_replication\_applier | 82dfcffa-a939-11e9-bb5c-000c29f8b2a5 | test4 | 3308 | ONLINE |

| group\_replication\_applier | b5ab9bc1-7cb0-11e9-816c-000c29dbfcd1 | 192.168.65.2 | 3306 | ONLINE |

+---------------------------+--------------------------------------+--------------+-------------+--------------+

发现 192.168.65.3 已经退出复制

c.当 192.168.65.3 恢复后，如果重新加入：

mysql> CHANGE MASTER TO MASTER\_USER='rpl', MASTER\_PASSWORD='123' FOR CHANNEL 'group\_replication\_recovery';

mysql> start group\_replication;

mysql> select \* from performance\_schema.replication\_group\_members;

+---------------------------+--------------------------------------+--------------+-------------+--------------+

| CHANNEL\_NAME | MEMBER\_ID | MEMBER\_HOST | MEMBER\_PORT | MEMBER\_STATE |

+---------------------------+--------------------------------------+--------------+-------------+--------------+

| group\_replication\_applier | 82dfcffa-a939-11e9-bb5c-000c29f8b2a5 | test4 | 3308 | ONLINE |

| group\_replication\_applier | b5ab9bc1-7cb0-11e9-816c-000c29dbfcd1 | 192.168.65.2 | 3306 | ONLINE |

| group\_replication\_applier | f3755962-a189-11e9-a5f7-000c298f0ec5 | 192.168.65.3 | 3306 | ONLINE |

+---------------------------+--------------------------------------+--------------+-------------+--------------+

**15. master 宕机测试：**

a.模拟 192.168.65.2 宕机

直接 Kill -9 mysql 的进程

b.slave 上查看状体

mysql> select \* from performance\_schema.replication\_group\_members;

+---------------------------+--------------------------------------+--------------+-------------+--------------+

| CHANNEL\_NAME | MEMBER\_ID | MEMBER\_HOST | MEMBER\_PORT | MEMBER\_STATE |

+---------------------------+--------------------------------------+--------------+-------------+--------------+

| group\_replication\_applier | 82dfcffa-a939-11e9-bb5c-000c29f8b2a5 | test4 | 3308 | ONLINE |

| group\_replication\_applier | f3755962-a189-11e9-a5f7-000c298f0ec5 | 192.168.65.3 | 3306 | ONLINE |

+---------------------------+--------------------------------------+--------------+-------------+--------------+

mysql> show global status like 'group\_replication\_primary\_member';

+----------------------------------+--------------------------------------+

| Variable\_name | Value |

+----------------------------------+--------------------------------------+

| group\_replication\_primary\_member | 82dfcffa-a939-11e9-bb5c-000c29f8b2a5 |

+----------------------------------+--------------------------------------+

可见，当 master 宕机后，将 test4 自动建立成了新的 master。

c.当 192.168.65.2 恢复后，如果重新加入：

mysql> CHANGE MASTER TO MASTER\_USER='rpl', MASTER\_PASSWORD='123' FOR CHANNEL 'group\_replication\_recovery';

mysql> start group\_replication;

mysql> select \* from performance\_schema.replication\_group\_members;

+---------------------------+--------------------------------------+--------------+-------------+--------------+

| CHANNEL\_NAME | MEMBER\_ID | MEMBER\_HOST | MEMBER\_PORT | MEMBER\_STATE |

+---------------------------+--------------------------------------+--------------+-------------+--------------+

| group\_replication\_applier | 82dfcffa-a939-11e9-bb5c-000c29f8b2a5 | test4 | 3308 | ONLINE |

| group\_replication\_applier | b5ab9bc1-7cb0-11e9-816c-000c29dbfcd1 | 192.168.65.2 | 3306 | ONLINE |

| group\_replication\_applier | f3755962-a189-11e9-a5f7-000c298f0ec5 | 192.168.65.3 | 3306 | ONLINE |

+---------------------------+--------------------------------------+--------------+-------------+--------------+

mysql> show global status like 'group\_replication\_primary\_member';

+----------------------------------+--------------------------------------+

| Variable\_name | Value |

+----------------------------------+--------------------------------------+

| group\_replication\_primary\_member | 82dfcffa-a939-11e9-bb5c-000c29f8b2a5 |

+----------------------------------+--------------------------------------+

**16.但主修改为多主模式**

多主模式如何配置呢，其实跟 单主模式的流程一模一样，只需要修改单主模式参数，把 group\_replication\_single\_primary\_mode 参数设置成关闭状态即可，然后按照单主模式的一直执行就可以了。

3个服务器依次执行：

mysql> stop group\_replication;

完成后在依次执行：

mysql> set global group\_replication\_single\_primary\_mode = OFF

然后执行：

#master实例
#需要启动 group\_replication\_bootstrap\_group 引导组，启动后需要关闭，防止脑裂
mysql> set global group\_replication\_bootstrap\_group=ON;
mysql> CHANGE MASTER TO MASTER\_USER='rpl', MASTER\_PASSWORD='123' FOR CHANNEL 'group\_replication\_recovery';
mysql> START GROUP\_REPLICATION;
mysql> set global group\_replication\_bootstrap\_group=Off;
#其他2个实例执行：
mysql> CHANGE MASTER TO MASTER\_USER='rpl', MASTER\_PASSWORD='123' FOR CHANNEL 'group\_replication\_recovery';
mysql> start group\_replication;

**PS：当退出组复制后，无论是 master 还是 slave 都会变成只读模式，这里要特别注意！！！**