# HOW TO CREATE A NEW (OR REPAIR A BROKEN) GTID BASED SLAVE

MySQL 5.6 introduced the new Global Transaction ID ([GTID](http://dev.mysql.com/doc/refman/5.6/en/replication-gtids-concepts.html)) support in replication. Percona XtraBackup from 2.1.0 version, automatically stores the GTID value in the xtrabackup\_binlog\_info when doing the backup of MySQL and Percona Server 5.6 with the GTID mode enabled. This information can be used to create a new (or repair a broken) GTID based slave.

## 分步指南

## TAKE A BACKUP FROM ANY SERVER ON THE REPLICATION ENVIRONMENT, MASTER OR SLAVE

Following command will take a backup to the /data/backups/$TIMESTAMP folder:

$ innobackupex /data/backups/

In the destination folder there will be a file with the name xtrabackup\_binlog\_info. This file will contain both, binary log coordinates and GTID information.

$ cat xtrabackup\_binlog\_info

c777888a-b6df-11e2-a604-080027635ef5:1-4

That information is also printed by innobackupex after backup is taken:

innobackupex: MySQL binlog position: GTID of the last change 'c777888a-b6df-11e2-a604-080027635ef5:1-4'

## PREPARE THE BACKUP

Back will be prepared with the following command:

TheMaster$ innobackupex --apply-log /data/backups/$TIMESTAMP/

You need to select path where your snapshot has been taken, for example /data/backups/2013-05-07\_08-33-33. If everything is ok you should get the same OK message. Now the transaction logs are applied to the data files, and new ones are created: your data files are ready to be used by the MySQL server.

## MOVE THE BACKUP TO THE DESTINATION SERVER

Use **rsync** or **scp** to copy the data to the destination server. If you’re syncing the data directly to already running slave’s data directory it’s advised to stop the MySQL server there.

 TheMaster$ rsync -avprP -e ssh /path/to/backupdir/$TIMESTAMP NewSlave:/path/to/mysql/

After you copy data over, make sure MySQL has proper permissions to access them.

 NewSlave$ chown mysql:mysql /path/to/mysql/datadir

## CONFIGURE AND START REPLICATION

Following command will tell the new slave what was the last GTID executed on the master when backup was taken.

NewSlave > SET GLOBAL gtid\_purged="c777888a-b6df-11e2-a604-080027635ef5:1-4";

NewSlave > CHANGE MASTER TO

MASTER\_HOST="$masterip",

MASTER\_USER="repl",

MASTER\_PASSWORD="$slavepass",

MASTER\_AUTO\_POSITION = 1;

## CHECK THE REPLICATION STATUS

Following command will show the slave status:

 NewSlave > show slave status\G

[..]

Slave\_IO\_Running: Yes

Slave\_SQL\_Running: Yes

[...]

Retrieved\_Gtid\_Set: c777888a-b6df-11e2-a604-080027635ef5:5

Executed\_Gtid\_Set: c777888a-b6df-11e2-a604-080027635ef5:1-5

We can see that the slave has retrieved a new transaction with number 5, so transactions from 1 to 5 are already on this slave.

That’s all, we have created a new slave in our GTID based replication environment.

<https://www.percona.com/doc/percona-xtrabackup/2.2/howtos/recipes_ibkx_gtid.html>

Icon

## 相关的文章

页面:

[数据库服务器部署规范](/pages/viewpage.action?pageId=8094602)

页面:

[MGR集群-大家参考启航的文档，更完善 by 郑靖](/pages/viewpage.action?pageId=8094589)

页面:

[PXC源码GDB调试](/pages/viewpage.action?pageId=8094397)

页面:

[Centos下python2.7.13及pip安装说明](/pages/viewpage.action?pageId=2819228)

页面:

[PXC源码安装](/pages/viewpage.action?pageId=8094393)

Showing first 5 of 22 results