1、什么是GTID?

1、全局唯一，一个事务对应一个GTID

2、替代传统的binlog+pos复制；使用master_auto_position=1自动匹配GTID断点进行复制

3、MySQL5.6开始支持

4、在传统的主从复制中，slave端不用开启binlog；但是在GTID主从复制中，必须开启binlog

5、slave端在接受master的binlog时，会校验GTID值

6、为了保证主从数据的一致性，多线程同时执行一个GTID

2、组成

Master_UUID:序列号举例：ceb0ca3d-8366-11e8-ad2b-000c298b7c9a:1-5ceb0ca3d-8366-11e8-ad2b-000c298b7c9a其实就是master的uuid值；1-5是序列号，每次一个事务完成都会自增1，也就是说下一次为1-6。

3、工作原理

1、master更新数据时，会在事务前产生GTID，一同记录到binlog日志中。

2、slave端的i/o 线程将变更的binlog，写入到本地的relay log中。

3、sql线程从relay log中获取GTID，然后对比slave端的binlog是否有记录。

4、如果有记录，说明该GTID的事务已经执行，slave会忽略。

5、如果没有记录，slave就会从relay log中执行该GTID的事务，并记录到binlog。

6、在解析过程中会判断是否有主键，如果没有就用二级索引，如果没有就用全部扫描

master添加配置

```
[mysqld]
basedir = /usr/local/mysql
datadir = /usr/local/mysql/data
port = 12308
socket  = /tmp/mysql.sock
character-set-server = utf8
log-error=/usr/local/mysql/mysqld.log
pid-file=/usr/local/mysql/mysqld.pid
lower_case_table_names=1
event_scheduler = 1

server-id=1
log-bin=/usr/local/mysql/binlog/mysql-bin.log
sql_mode =
"STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_AUTO_CREATE_USER,NO_ENGINE_SUBSTITUTION"
log_bin_trust_function_creators=1

gtid_mode = on #开启GTID模式
enforce_gtid_consistency = on #使用GTID模式复制时，需要开启参数，用来保证数据的一致性强制gtid复制
binlog_format = row #binlog格式为row
log-slave-updates = 1 #决定SLAVE从Master接收到更新且执行是否记录到SLAVE的binlog中
skip_slave_start = 1 #当SLAVE数据库启动的时候，SLAVE不会启动复制
```

```
auto-increment-increment = 2
auto-increment-offset = 1
```

slave添加配置

```
[mysqld]
basedir = /usr/local/mysql
datadir = /usr/local/mysql/data
port = 12308
socket  = /tmp/mysql.sock
character-set-server = utf8
log-error=/usr/local/mysql/mysqld.log
pid-file=/usr/local/mysql/mysqld.pid
lower_case_table_names=1
event_scheduler = 1

slow_query_log=ON
slow_query_log_file=/usr/local/mysql/mysql-slow.log
long_query_time=1

server-id=2
log-bin=/usr/local/mysql/binlog/mysql-bin.log
sql_mode =
"STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_AUTO_CREATE_USER,NO_ENGINE_SUBSTITUTION"
log_bin_trust_function_creators=1

gtid_mode = on #开启GTID模式
enforce_gtid_consistency = on #使用GTID模式复制时，需要开启参数，用来保证数据的一致性强制gtid复制
binlog_format = row #binlog格式为row
log-slave-updates = 1 #决定SLAVE从Master接收到更新且执行是否记录到SLAVE的binlog中
skip_slave_start = 1 #当SLAVE数据库启动的时候，SLAVE不会启动复制
auto-increment-increment = 2
auto-increment-offset = 1
```

server-id 值主从要不一致

master授权配置

```
mysql -uroot -p
mysql>grant replication client,replication slave on *.* to 'rep'@'172.60.51.%' identified by 'qwe123';
mysql> flush privileges;
```

slave配置同步

```
mysql -uroot -p
```

```
mysql> change master to master_host='172.60.51.91',
master_user='rep',master_password='qwe123',master_port=12308,master_auto_position=1;
mysql> start slave;
```

查看slave的状态

mysql> show slave status\G

```
*************************** 1. row ***************************
               Slave_IO_State: Waiting for master to send event
                  Master_Host: 172.60.17.10
                  Master_User: rep
                  Master_Port: 12308
                Connect_Retry: 60
              Master_Log_File: mysql-bin.000005
          Read_Master_Log_Pos: 116849436
               Relay_Log_File: t02-relay-bin.000003
                Relay_Log_Pos: 116849649
        Relay_Master_Log_File: mysql-bin.000005
             Slave_IO_Running: Yes
            Slave_SQL_Running: Yes
              Replicate_Do_DB:
          Replicate_Ignore_DB:
           Replicate_Do_Table:
       Replicate_Ignore_Table:
      Replicate_Wild_Do_Table:
  Replicate_Wild_Ignore_Table:
                   Last_Errno: 0
                   Last_Error:
                 Skip_Counter: 0
          Exec_Master_Log_Pos: 116849436
              Relay_Log_Space: 116849854
              Until_Condition: None
               Until_Log_File:
                Until_Log_Pos: 0
           Master_SSL_Allowed: No
           Master_SSL_CA_File:
           Master_SSL_CA_Path:
              Master_SSL_Cert:
            Master_SSL_Cipher:
               Master_SSL_Key:
        Seconds_Behind_Master: 0
Master_SSL_Verify_Server_Cert: No
                Last_IO_Errno: 0
                Last_IO_Error:
               Last_SQL_Errno: 0
               Last_SQL_Error:
  Replicate_Ignore_Server_Ids:
             Master_Server_Id: 1
                  Master_UUID: cd5d521b-b1c7-11e9-a1dc-000c2980d8b3
             Master_Info_File: /usr/local/mysql/data/master.info
                    SQL_Delay: 0
          SQL_Remaining_Delay: NULL
      Slave_SQL_Running_State: Slave has read all relay log; waiting for more updates
           Master_Retry_Count: 86400
                  Master_Bind:
      Last_IO_Error_Timestamp:
     Last_SQL_Error_Timestamp:
               Master_SSL_Crl:
           Master_SSL_Crlpath:
```

```
        Retrieved_Gtid_Set: cd5d521b-b1c7-11e9-a1dc-000c2980d8b3:1-355
         Executed_Gtid_Set: cd5d521b-b1c7-11e9-a1dc-000c2980d8b3:1-355
             Auto_Position: 1
      Replicate_Rewrite_DB:
              Channel_Name:
        Master_TLS_Version:
1 row in set (0.00 sec)
```

```
                  Master_Host: 172.60.17.10
                  Master_User: rep
                  Master_Port: 12308
                Connect_Retry: 60
              Master_Log_File: mysql-bin.000005
          Read_Master_Log_Pos: 116849436
               Relay_Log_File: t02-relay-bin.000003
                Relay_Log_Pos: 116849649
        Relay_Master_Log_File: mysql-bin.000005
             Slave_IO_Running: Yes
            Slave_SQL_Running: Yes
              Replicate_Do_DB:
```

```
           Master_Server_Id: 1
                Master_UUID: cd5d521b-b1c7-11e9-a1dc-000c2980d8b3
           Master_Info_File: /usr/local/mysql/data/master.info
                  SQL_Delay: 0
        SQL_Remaining_Delay: NULL
    Slave_SQL_Running_State: Slave has read all relay log; waiting for more updates
         Master_Retry_Count: 86400
                Master_Bind:
    Last_IO_Error_Timestamp:
   Last_SQL_Error_Timestamp:
             Master_SSL_Crl:
         Master_SSL_Crlpath:
         Retrieved_Gtid_Set: cd5d521b-b1c7-11e9-a1dc-000c2980d8b3:1-355
          Executed_Gtid_Set: cd5d521b-b1c7-11e9-a1dc-000c2980d8b3:1-355
              Auto_Position: 1
       Replicate_Rewrite_DB:
               Channel_Name:
         Master_TLS_Version:
1 row in set (0.00 sec)
```

Retrieved_Gtid_Set 表示slave从master接受的gtid set，使用 reset slave 命令可以清空此项；

Executed_Gtid_Set 表示slave已执行的gtid set，使用 reset master 命令可以清空此项。

Retrieved_Gtid_Set 和 Executed_Gtid_Set 必须为master 上 gtid set 的子集，否则会报以下错误：

查看master状态

```
mysql> show master status;
+------------------+----------+--------------+------------------+-------------------------------------------+
| File             | Position | Binlog_Do_DB | Binlog_Ignore_DB | Executed_Gtid_Set                         |
+------------------+----------+--------------+------------------+-------------------------------------------+
| mysql-bin.000005 | 116849436 |              |                  | cd5d521b-b1c7-11e9-a1dc-000c2980d8b3:1-355 |
+------------------+----------+--------------+------------------+-------------------------------------------+
1 row in set (0.00 sec)
```

对比slave端，Executed_Gtid_Set的值应该是一样的。

参考URL：https://blog.51cto.com/13434336/2178937

master uuid: cd5d521b-b1c7-11e9-a1dc-000c2980d8b3

slave uuid: 08109ba1-b994-11e9-b0ad-000c290b47ec

其他：

NO.1　从库有数据写入（即从库插入数据）

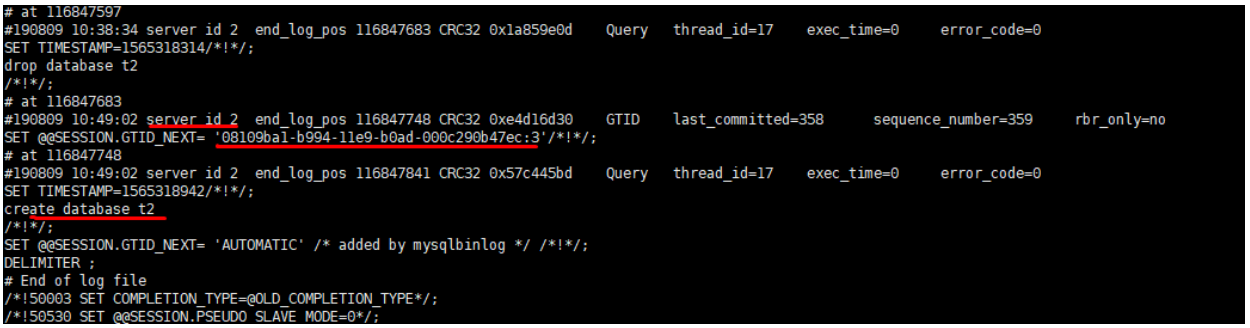
mysql> create database t2;

```
              Until_Log_File:
               Until_Log_Pos: 0
          Master_SSL_Allowed: No
          Master_SSL_CA_File:
          Master_SSL_CA_Path:
             Master_SSL_Cert:
           Master_SSL_Cipher:
              Master_SSL_Key:
       Seconds_Behind_Master: 0
Master_SSL_Verify_Server_Cert: No
               Last_IO_Errno: 0
               Last_IO_Error:
              Last_SQL_Errno: 0
              Last_SQL_Error:
  Replicate_Ignore_Server_Ids:
             Master_Server_Id: 1
                  Master_UUID: cd5d521b-b1c7-11e9-a1dc-000c2980d8b3
             Master_Info_File: /usr/local/mysql/data/master.info
                    SQL_Delay: 0
          SQL_Remaining_Delay: NULL
      Slave_SQL_Running_State: Slave has read all relay log; waiting for more updates
           Master_Retry_Count: 86400
                  Master_Bind:
      Last_IO_Error_Timestamp:
     Last_SQL_Error_Timestamp:
               Master_SSL_Crl:
           Master_SSL_Crlpath:
           Retrieved_Gtid_Set: cd5d521b-b1c7-11e9-a1dc-000c2980d8b3:1-356
            Executed_Gtid_Set: 08109ba1-b994-11e9-b0ad-000c290b47ec:1-3,
cd5d521b-b1c7-11e9-a1dc-000c2980d8b3:1-356
                Auto_Position: 1
         Replicate_Rewrite_DB:
                 Channel_Name:
           Master_TLS_Version:
1 row in set (0.00 sec)
```

可以看到已经执行的事务有来自主库的cd5d521b-b1c7-11e9-a1dc-000c2980d8b3:1-356,

也有从库自己写入的数据：08109ba1-b994-11e9-b0ad-000c290b47ec:1-3。我们可以解析binlog看看

/usr/local/mysql/bin/mysqlbinlog -vv mysql-bin.000001 --include-gtids='08109ba1-b994-11e9-b0ad-000c290b47ec:1-3'



从binlog中可以清楚看到是从库进行了写入。