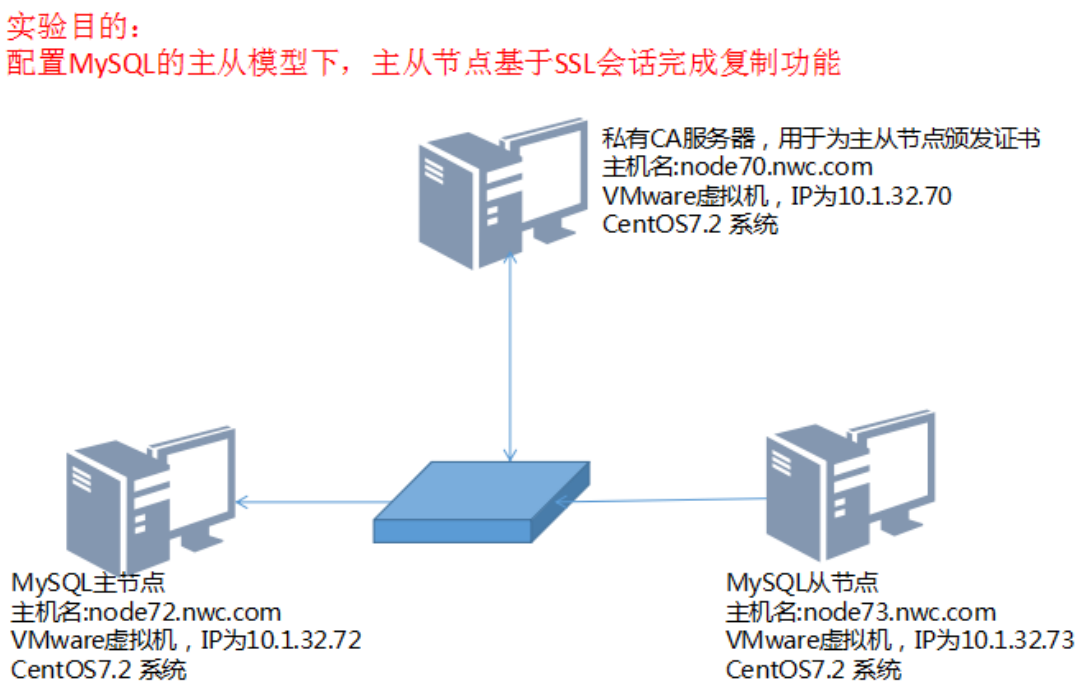
Mariadb数据库复制系列(五)：基于SSL的复制

实验五：基于SSL的主从复制功能的实现

> 在mysql服务器之间复制数据，默认情况下都是基于明文的，在有些场景中，明文传输会造成严重的数据安全隐患，因此，需要对mysql服务器之间的复制时的传输进行加密，传输加密方式可以基于SSL的会话进行

1、实验环境



2、私有CA的搭建

```
[root@node70 ~]# touch /etc/pki/CA/{serial,index.txt}
[root@node70 ~]# echo 01 > /etc/pki/CA/serial
[root@node70 ~]# (umask 077;openssl genrsa -out /etc/pki/CA/private/cakey.pem 2048)
Generating RSA private key, 2048 bit long modulus
..............................+++                    提供私有CA所需的文件，提供证书初始编号，生成CA的私钥
.......................................+++
e is 65537 (0x10001)
[root@node70 ~]#
[root@node70 ~]# openssl req -x509 -new -key /etc/pki/CA/private/cakey.pem -out /etc/pki/CA/cacert.
pem -days 3650
You are about to be asked to enter information that will be incorporated .
into your certificate request.                                          生成CA的自签证书
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [XX]:cn
State or Province Name (full name) []:bj      提供相关信息
Locality Name (eg, city) [Default City]:bj
Organization Name (eg, company) [Default Company Ltd]:nwc
Organizational Unit Name (eg, section) []:nwc
Common Name (eg, your name or your server's hostname) []:ca.nwc.com
Email Address []:
[root@node70 ~]#
```

3、在主节点node72上生成证书签署请求、发送到私有CA服务器

```
[root@node72 ~]# (umask 077;openssl genrsa -out /etc/mysql/ssl/node72.key 1024)
Generating RSA private key, 1024 bit long modulus
.....................................................................++++++
..............++++++          在node72上生成私钥文件，生成证书签署请求，将证书签署请求文件发送到私有CA上
e is 65537 (0x10001)
[root@node72 ~]# openssl req -new -key /etc/mysql/ssl/node72.key -out /etc/mysql/node72.csr -days 3
65
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [XX]:cn
State or Province Name (full name) []:bj
Locality Name (eg, city) [Default City]:bj
Organization Name (eg, company) [Default Company Ltd]:nwc
Organizational Unit Name (eg, section) []:nwc
Common Name (eg, your name or your server's hostname) []:node72.nwc.com
Email Address []:

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
[root@node72 ~]# scp /etc/mysql/node72.csr 10.1.32.70:/tmp
node72.csr                                    100%  635     0.6KB/s   00:00
[root@node72 ~]#
```

4、在从节点node73上生成证书签署请求、发送到私有CA服务器

```
[root@node73 ~]# mkdir -pv /etc/mysql/ssl
mkdir: created directory ‘/etc/mysql’
mkdir: created directory ‘/etc/mysql/ssl’
[root@node73 ~]# (umask 077;openssl genrsa -out /etc/mysql/ssl/node73.key 1024)
Generating RSA private key, 1024 bit long modulus
.....................................................................++++++
.........................++++++        在node73上生成私钥，生成证书签署请求，将签署请求的文件发送给私有CA
e is 65537 (0x10001)
[root@node73 ~]# openssl req -new -key /etc/mysql/ssl/node73.key -out /etc/mysql/ssl/node73.csr -da
ys 365
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [XX]:cn
State or Province Name (full name) []:bj
Locality Name (eg, city) [Default City]:bj
Organization Name (eg, company) [Default Company Ltd]:nwc
Organizational Unit Name (eg, section) []:nwc
Common Name (eg, your name or your server's hostname) []:node73.nwc.com
Email Address []:

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
[root@node73 ~]# scp /etc/mysql/ssl/node73.csr 10.1.32.70:/tmp
node73.csr                                                    100%  635     0.6KB/s   00:00
```

5、私有CA为两个节点颁发证书，将证书发送给两个节点

```
[root@node70 ~]# ls /tmp
node72.csr  node73.csr  yum.log
[root@node70 ~]#
[root@node70 ~]# openssl ca -in /tmp/node72.csr -out /etc/pki/CA/certs/node72.crt -days 365
Using configuration from /etc/pki/tls/openssl.cnf
Check that the request matches the signature        签署node72的证书
Signature ok
Certificate Details:
        Serial Number: 1 (0x1)
        Validity
            Not Before: Nov 20 03:56:23 2016 GMT
            Not After : Nov 20 03:56:23 2017 GMT
        Subject:
            countryName               = cn
            stateOrProvinceName       = bj
            organizationName          = nwc
            organizationalUnitName    = nwc
            commonName                = node72.nwc.com
Certificate is to be certified until Nov 20 03:56:40 2017 GMT (365 days)
Sign the certificate? [y/n]:y


1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
[root@node70 ~]#                          将node72生成的证书文件和CA自己的证书拷贝的node72上
[root@node70 ~]# scp /etc/pki/CA/certs/node72.crt 10.1.32.72:/etc/mysql/ssl/
node72.crt                                                    100% 3650     3.6KB/s   00:00
[root@node70 ~]# scp /etc/pki/CA/cacert.pem 10.1.32.72:/etc/mysql/ssl/
cacert.pem                                                    100% 1277     1.3KB/s   00:00
[root@node70 ~]#
[root@node70 ~]# openssl ca -in /tmp/node73.csr -out /etc/pki/CA/certs/node73.crt -days 365
Using configuration from /etc/pki/tls/openssl.cnf    签署node73的证书
Check that the request matches the signature
Signature ok
Certificate Details:
        Serial Number: 2 (0x2)
        Validity
            Not Before: Nov 20 03:56:40 2016 GMT
            Not After : Nov 20 03:56:40 2017 GMT
        Subject:
            countryName               = cn
            stateOrProvinceName       = bj
            organizationName          = nwc
            organizationalUnitName    = nwc
            commonName                = node73.nwc.com
Certificate is to be certified until Nov 20 03:56:40 2017 GMT (365 days)
Sign the certificate? [y/n]:y


1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
[root@node70 ~]#                                  拷贝生成的node73的证书和CA自己的证书到node73上
[root@node70 ~]# scp /etc/pki/CA/certs/node73.crt 10.1.32.73:/etc/mysql/ssl/
node73.crt                                                    100% 3650     3.6KB/s   00:00
[root@node70 ~]# scp /etc/pki/CA/cacert.pem 10.1.32.73:/etc/mysql/ssl/
cacert.pem                                                    100% 1277     1.3KB/s   00:00
[root@node70 ~]#
```

6、在两个节点上分别修改证书相关文件的权限，让mysql用户拥有读取权限

```
[root@node72 ~]# chown mysql:mysql /etc/mysql/ssl/*
[root@node72 ~]# ll /etc/mysql/ssl/
总用量 12
-rw-r--r-- 1 mysql mysql 1277 11月 20 11:59 cacert.pem
-rw-r--r-- 1 mysql mysql 3650 11月 20 11:59 node72.crt
-rw------- 1 mysql mysql  887 11月 20 11:46 node72.key
[root@node72 ~]#
[root@node73 ~]# chown mysql:mysql /etc/mysql/ssl/*
[root@node73 ~]# ll /etc/mysql/ssl/
total 16
-rw-r--r-- 1 mysql mysql 1277 Nov 20 12:02 cacert.pem
-rw-r--r-- 1 mysql mysql 3650 Nov 20 12:02 node73.crt
-rw-r--r-- 1 mysql mysql  635 Nov 20 11:51 node73.csr
-rw------- 1 mysql mysql  887 Nov 20 11:50 node73.key
[root@node73 ~]#
```

修改私钥、证书、CA证书的权限，让mysql用户具有读取的权限

修改SSL相关的文件的权限，让mysql有读取权限

7、在两个节点上安装mariadb-server

```
[root@node72 ~]# yum install -y mariadb-server
已加载插件：fastestmirror, langpacks
Repodata is over 2 weeks old. Install yum-cron? Or run: yum makecache fast
BASE                                                                | 3.6 kB  00:00:00
EPEL                                                                | 4.3 kB  00:00:00
Determining fastest mirrors
正在解决依赖关系
--> 正在检查事务
---> 软件包 mariadb-server.x86_64.1.5.5.44-2.el7.centos 将被 安装
--> 正在处理依赖关系 mariadb(x86-64) = 1:5.5.44-2.el7.centos，它被软件包 1:mariadb-server-5.5.44-2.
el7.centos.x86_64 需要
--> 正在处理依赖关系 perl-DBI，它被软件包 1:mariadb-server-5.5.44-2.el7.centos.x86_64 需要
--> 正在处理依赖关系 perl-DBD-MySQL，它被软件包 1:mariadb-server-5.5.44-2.el7.centos.x86_64 需要
--> 正在处理依赖关系 perl(Data::Dumper)，它被软件包 1:mariadb-server-5.5.44-2.el7.centos.x86_64 需
要
--> 正在处理依赖关系 perl(DBI)，它被软件包 1:mariadb-server-5.5.44-2.el7.centos.x86_64 需要
--> 正在检查事务
---> 软件包 mariadb.x86_64.1.5.5.44-2.el7.centos 将被 安装
---> 软件包 perl-DBD-MySQL.x86_64.0.4.023-5.el7 将被 安装
[root@node73 ~]# yum install -y mariadb-server
Loaded plugins: fastestmirror
BASE                                                                | 3.6 kB  00:00:00
EPEL                                                                | 4.3 kB  00:00:00
Loading mirror speeds from cached hostfile
Resolving Dependencies
--> Running transaction check
---> Package mariadb-server.x86_64 1:5.5.44-2.el7.centos will be installed
--> Processing Dependency: mariadb(x86-64) = 1:5.5.44-2.el7.centos for package: 1:mariadb-server-5.
5.44-2.el7.centos.x86_64
--> Processing Dependency: perl-DBI for package: 1:mariadb-server-5.5.44-2.el7.centos.x86_64
--> Processing Dependency: perl-DBD-MySQL for package: 1:mariadb-server-5.5.44-2.el7.centos.x86_64
--> Processing Dependency: perl(Data::Dumper) for package: 1:mariadb-server-5.5.44-2.el7.centos.x86
_64
--> Processing Dependency: perl(DBI) for package: 1:mariadb-server-5.5.44-2.el7.centos.x86_64
--> Processing Dependency: libaio.so.1(LIBAIO_0.4)(64bit) for package: 1:mariadb-server-5.5.44-2.el
7.centos.x86_64
```

在node72上安装mariadb-server

在node73上安装mariadb-server

8、配置修改主节点的配置文件，启动服务，让其满足基于SSL会话的主从复制时主节点的相关属性

```
[root@node72 ~]# vim /etc/my.cnf
```
配置node72，也就是主节点上的服务配置文件，让其满足主从结构中主节点的配置

```
[mysqld]
datadir=/var/lib/mysql
socket=/var/lib/mysql/mysql.sock
# Disabling symbolic-links is recommended to prevent assorted security risks
symbolic-links=0
# Settings user and group are ignored when systemd is used.
# If you need to run mysqld under a different user or group,
# customize your systemd unit file for mariadb according to the
# instructions in http://fedoraproject.org/wiki/Systemd
skip_name_resolve=ON
innodb_file_per_table=ON
server_id=1 指明全局唯一的server id
log_bin=node72binlog 启用二进制日志功能，指明对应的二进制日志文件
ssl 启用SSL功能
ssl_ca=/etc/mysql/ssl/cacert.pem 指明信任的CA证书，可以用ssl_capath指明一个路径，该路径下的所有CA都被信任
ssl_cert=/etc/mysql/ssl/node72.crt 指明自身的证书文件
ssl_key=/etc/mysql/ssl/node72.key  指明自身的私钥文件

[mysqld_safe]
log-error=/var/log/mariadb/mariadb.log
pid-file=/var/run/mariadb/mariadb.pid

#
[root@node72 ~]# systemctl start mariadb
[root@node72 ~]# ss -tnl
State       Recv-Q Send-Q    Local Address:Port                 Peer Address:Port
LISTEN      0      50                *:3306                            *:*
LISTEN      0      128               *:22                              *:*
LISTEN      0      100           127.0.0.1:25                          *:*
LISTEN      0      128              :::22                             :::*
LISTEN      0      100              ::1:25                            :::*
[root@node72 ~]# mysql
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 2
Server version: 5.5.44-MariaDB-log MariaDB Server

Copyright (c) 2000, 2015, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> GRANT REPLICATION SLAVE,REPLICATION CLIENT ON *.* TO 'node72user'@'10.1.32.73' ID
ENTIFIED BY '111111' REQUIRE SSL;
Query OK, 0 rows affected (0.00 sec)
```
授权一个具有复制权限的用户，仅允许该用户在node73也就是从节点上登录，且要求其必须基于SSL才能登录

```
MariaDB [(none)]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.02 sec)

MariaDB [(none)]> SHOW MASTER STATUS;
```
查看主节点当前二进制日志文件所处的位置

```
+--------------------+----------+--------------+------------------+
| File               | Position | Binlog_Do_DB | Binlog_Ignore_DB |
+--------------------+----------+--------------+------------------+
| node72binlog.000005 |      508 |              |                  |
+--------------------+----------+--------------+------------------+
1 row in set (0.00 sec)

MariaDB [(none)]>
```

9、在从节点上测试，是否能够基于**ssl**会话的方式与主服务器进行连接

```
[root@node73 ~]# mysql -unode72user -h10.1.32.72 -p111111 --ssl
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 47
Server version: 5.5.44-MariaDB-log MariaDB Server

Copyright (c) 2000, 2015, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> \s
--------------
mysql  Ver 15.1 Distrib 5.5.44-MariaDB, for Linux (x86_64) using readline 5.1

Connection id:          47
Current database:
Current user:           node72user@10.1.32.73
SSL:                    Cipher in use is DHE-RSA-AES256-GCM-SHA384
Current pager:          stdout
Using outfile:          ''
Using delimiter:        ;
Server:                 MariaDB
Server version:         5.5.44-MariaDB-log MariaDB Server
Protocol version:       10
Connection:             10.1.32.72 via TCP/IP
Server characterset:    latin1
Db     characterset:    latin1
Client characterset:    utf8
```

在从节点上验证，是否能够通过ssl绘画方式与主节点建立连接

10、修改从节点的服务器配置文件，让其满足主从结构中从节点的要求

```
[root@node73 ~]# vim /etc/my.cnf

[mysqld]
datadir=/var/lib/mysql
socket=/var/lib/mysql/mysql.sock
# Disabling symbolic-links is recommended to prevent assorted security risks
symbolic-links=0
# Settings user and group are ignored when systemd is used.
# If you need to run mysqld under a different user or group,
# customize your systemd unit file for mariadb according to the
# instructions in http://fedoraproject.org/wiki/Systemd
skip_name_resolve=ON
innodb_file_per_table=ON
server_id=2 设定全局唯一server id
relay_log=node73relaylog 启动中继日志功能
read_only=ON 限定从服务器只读属性

[mysqld_safe]
log-error=/var/log/mariadb/mariadb.log
pid-file=/var/run/mariadb/mariadb.pid
```

配置node73的mysql服务配置文件，让其满足从节点的配置要求

11、定义从节点从主节点复制数据时的属性，让其能够，启动复制线程

```
[root@node73 ~]# systemctl start mariadb
[root@node73 ~]# ss -tnl
State       Recv-Q Send-Q     Local Address:Port                    Peer Address:Port
LISTEN      0      50                     *:3306                               *:*
LISTEN      0      128                    *:22                                 *:*
LISTEN      0      100          127.0.0.1:25                                   *:*
LISTEN      0      128                 :::22                                :::*
LISTEN      0      100                ::1:25                                :::*
[root@node73 ~]# mysql
```

启动从节点的服务

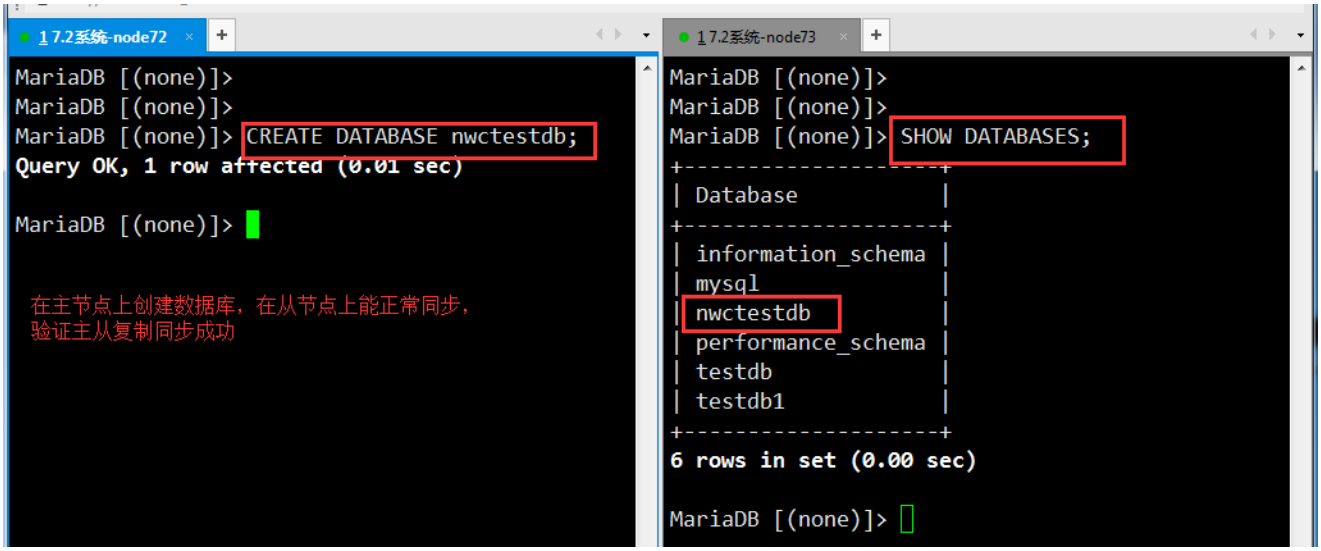
```
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 4
Server version: 5.5.44-MariaDB MariaDB Server

Copyright (c) 2000, 2015, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> CHANGE MASTER TO MASTER_HOST='10.1.32.72',MASTER_USER='node72user',MASTER_PASSWOR
D='111111',MASTER_LOG_FILE='node72binlog.000005',MASTER_LOG_POS=508,MASTER_SSL=1,MASTER_SSL_CA='/et
c/mysql/ssl/cacert.pem',MASTER_SSL_CERT='/etc/mysql/ssl/node73.crt',MASTER_SSL_KEY='/etc/mysql/ssl/
node73.key',MASTER_SSL_VERIFY_SERVER_CERT=0;
Query OK, 0 rows affected (0.03 sec)

MariaDB [(none)]> START SLAVE;
Query OK, 0 rows affected (0.00 sec)

MariaDB [(none)]> SHOW SLAVE STATUS\G
*************************** 1. row ***************************
               Slave_IO_State: Waiting for master to send event
                  Master_Host: 10.1.32.72
                  Master_User: node72user
                  Master_Port: 3306
                Connect_Retry: 60
              Master_Log_File: node72binlog.000005
          Read_Master_Log_Pos: 730
               Relay_Log_File: node73relaylog.000002
                Relay_Log_Pos: 754
        Relay_Master_Log_File: node72binlog.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: 730
              Relay_Log_Space: 1047
              Until_Condition: None
               Until_Log_File:
                Until_Log_Pos: 0
           Master_SSL_Allowed: Yes
           Master_SSL_CA_File: /etc/mysql/ssl/cacert.pem
           Master_SSL_CA_Path:
              Master_SSL_Cert: /etc/mysql/ssl/node73.crt
            Master_SSL_Cipher:
               Master_SSL_Key: /etc/mysql/ssl/node73.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
1 row in set (0.00 sec)
```

设置从服务器连接主服务器进行复制时的相关属性，因为是基于SSL，故要指明SSL使用的证书之类的选项

启动复制线程

查看从节点状态

12、验证基于SSL的主从复制是否配置成功


17.2系统-node72
MariaDB [(none)]>
MariaDB [(none)]>
MariaDB [(none)]> CREATE DATABASE nwctestdb;
Query OK, 1 row affected (0.01 sec)
MariaDB [(none)]>
在主节点上创建数据库，在从节点上能正常同步，
验证主从复制同步成功
17.2系统-node73
MariaDB [(none)]>
MariaDB [(none)]>
MariaDB [(none)]> SHOW DATABASES;
+--------------------+
| Database |
+--------------------+
| information_schema |
| mysql |
| nwctestdb |
| performance_schema |
| testdb |
| testdb1 |
+--------------------+
6 rows in set (0.00 sec)
MariaDB [(none)]>