

Replication cluster on MariaDB 5.5 / ubuntu-server

Mark Schneider

[ms\(at\)it-infrastrukturen\(dot\)org](mailto:ms(at)it-infrastrukturen(dot)org)

2012-05-31



Abstract

Setting of MASTER-SLAVE or MASTER-MASTER replications on MariaDB 5.5 database servers is necessary for higher availability of data and better performance in professional environment.

MariaDB is a database server that offers drop-in replacement functionality for MySQL. MariaDB is built by some of the original authors of MySQL, with assistance from the broader community of Free and open source software developers. In addition to the core functionality of MySQL, MariaDB offers a rich set of feature enhancements including alternate storage engines, server optimizations, and patches.

MariaDB is primarily driven by developers at Monty Program, a company founded by Michael "Monty" Widenius, the original author of MySQL, but this is not the whole story about MariaDB. On the "About MariaDB" page you will find more information about all participants in the MariaDB community, including storage engines XtraDB and PBXT.

This document is designed to be readable by someone with basic Linux/UNIX command-line, ubuntu administration and basic SQL programming skills.

* All trademarks or registered trademarks herein in this document are the property of their respective owners.

Setting of replication cluster on MariaDB 5.5 / ubuntu-server

The below procedure describes steps required to configure replications on MariaDB 5.5 database server running on ubuntu-server 12.04 (LTS). It is also directly applicable for debian squeeze and wheezy and mostly to other linux distributions.

These document and some config files for replication configurations on MariaDB 5.5 are available as .tgz archive at:

<http://rsync.it-infrastrukturen.org/mariadb/ubuntu/mariadb55-ubuntu-replications.tgz>

Two levels of activities are required on each database server box to set up MASTER-SLAVE or MASTER-MASTER replications for MariaDB 5.5:

- shell commands on the operating system prompt
- SQL commands on the MariaDB prompt

This description uses two database server boxes db10 (*MASTER) and db20 (*SLAVE). They run ubuntu-server 12.04 LTS and MariaDB 5.5.24 database server.

* In the MASTER-MASTER configuration each box is a master and slave to the other one.

Short introduction

Replication is basically a three step process which works as follows:

- The master records changes to data in its binary log.
- The slave copies the changes recorded in the master's binlog to its relay log.
- The slave replays the change-set recorded in its relay log, applying these changes to its own data.

A replication account on the master is required to allow the slave to connect to the master and read master's binary log.

Configuration of MASTER-SLAVE replications for MariaDB

MariaDB master-slave replication mechanism allows replication of data from one master database server to one or more slave database servers. Replication is mostly used as a scale-out solution. The load is spreaded among multiple slaves to improve performance. All writes and updates take place on the master server, while reads occur on one or more slaves and/or the master. Such model improves both the write performance on the master (less reads take place there) and the read performance as well (in particular with increasing number of slaves).

After installation of ubuntu-server and MariaDB 5.5 the following steps are necessary:

- modify my.cnf files on db10 and db20 due to MASTER-SLAVE replications requirements
- modify / extend settings of MariaDB on the SQL prompt
- create an example database with one table for testing

Configuration of *MASTER (db10) for MASTER-SLAVE replications

The my.cnf file on the *MASTER should look like in the following listing. After changes a database server restart is required (/etc/init.d/mysql restart).

Listing 1: my.cnf-MASTER-db10

```
# MariaDB database server configuration file.
#
# You can copy this file to one of:
# - "/etc/mysql/my.cnf" to set global options,
# - "~/.my.cnf" to set user-specific options.
#
# One can use all long options that the program supports.
# Run program with --help to get a list of available options and with
# --print-defaults to see which it would actually understand and use.
#
# For explanations see
# http://dev.mysql.com/doc/mysql/en/server-system-variables.html

# This will be passed to all mysql clients
# It has been reported that passwords should be enclosed with ticks/quotes
# especially if they contain "#" chars...
# Remember to edit /etc/mysql/debian.cnf when changing the socket location.
[client]
port                = 3306
socket              = /var/run/mysqld/mysqld.sock

# Here is entries for some specific programs
# The following values assume you have at least 32M ram

# This was formally known as [safe_mysqld]. Both versions are currently parsed.
[mysqld_safe]
socket              = /var/run/mysqld/mysqld.sock
nice                = 0

[mysqld]
#
# * Basic Settings
#
user                = mysql
pid-file            = /var/run/mysqld/mysqld.pid
socket              = /var/run/mysqld/mysqld.sock
port                = 3306
basedir             = /usr
datadir             = /var/lib/mysql
```

```

tmpdir                = /tmp
lc_messages_dir       = /usr/share/mysql
lc_messages           = en_US
skip-external-locking
#
# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
#bind-address          = 127.0.0.1
bind-address          = 192.168.10.50
#
# * Fine Tuning
#
max_connections        = 4000
connect_timeout        = 5
wait_timeout           = 600
max_allowed_packet     = 16M
thread_cache_size      = 128
sort_buffer_size       = 16M
bulk_insert_buffer_size = 16M
tmp_table_size         = 32M
max_heap_table_size    = 32M
#
# * MyISAM
#
# This replaces the startup script and checks MyISAM tables if needed
# the first time they are touched. On error, make copy and try a repair.
myisam_recover         = BACKUP
key_buffer_size        = 128M
#open-files-limit      = 2000
table_cache            = 400
myisam_sort_buffer_size = 512M
concurrent_insert       = 2
read_buffer_size       = 16M
read_rnd_buffer_size   = 8M
#
# * Query Cache Configuration
#
# Cache only tiny result sets, so we can fit more in the query cache.
query_cache_limit       = 128K
query_cache_size        = 64M
# for more write intensive setups, set to DEMAND or OFF
#query_cache_type       = DEMAND
#
# * Logging and Replication
#
# Both location gets rotated by the cronjob.
# Be aware that this log type is a performance killer.
# As of 5.1 you can enable the log at runtime!
#general_log_file        = /var/log/mysql/mysql.log
#general_log             = 1
#
# Error logging goes to syslog due to /etc/mysql/conf.d/mysqld_safe_syslog.cnf.
#
# we do want to know about network errors and such
log_warnings            = 2
#
# Enable the slow query log to see queries with especially long duration
#slow_query_log[={0|1}]
slow_query_log_file     = /var/log/mysql/mariadb-slow.log
long_query_time = 10
#log_slow_rate_limit     = 1000
log_slow_verbosity      = query-plan

```

```

#log-queries-not-using-indexes
#log_slow_admin_statements
#
# The following can be used as easy to replay backup logs or for replication.
# note: if you are setting up a replication slave, see README.Debian about
#       other settings you may need to change.
server-id                = 10
#report_host              = master1
auto_increment_increment = 10
auto_increment_offset    = 1
log_bin                  = /var/log/mysql/mariadb-db10-bin
log_bin_index             = /var/log/mysql/mariadb-db10-bin.index
# not fab for performance, but safer
#sync_binlog              = 1
expire_logs_days         = 10
max_binlog_size           = 100M
# slaves
#relay_log                = /var/log/mysql/relay-bin
#relay_log_index           = /var/log/mysql/relay-bin.index
#relay_log_info_file       = /var/log/mysql/relay-bin.info
#log_slave_updates
#read-only
#
# If applications support it, this stricter sql_mode prevents some
# mistakes like inserting invalid dates etc.
#sql_mode                  = NO_ENGINE_SUBSTITUTION,TRADITIONAL
#
# * InnoDB
#
# InnoDB is enabled by default with a 10MB datafile in /var/lib/mysql/.
# Read the manual for more InnoDB related options. There are many!
default_storage_engine    = InnoDB
# you can't just change log file size, requires special procedure
#innodb_log_file_size     = 50M
innodb_buffer_pool_size   = 2048M
innodb_log_buffer_size    = 8M
innodb_file_per_table     = 1
innodb_open_files         = 400
innodb_io_capacity        = 400
innodb_flush_method       = O_DIRECT
#
# * Security Features
#
# Read the manual, too, if you want chroot!
# chroot = /var/lib/mysql/
#
# For generating SSL certificates I recommend the OpenSSL GUI "tinyca".
#
# ssl-ca=/etc/mysql/cacert.pem
# ssl-cert=/etc/mysql/server-cert.pem
# ssl-key=/etc/mysql/server-key.pem

[mysqldump]
quick
quote-names
max_allowed_packet        = 16M

[mysql]
#no-auto-rehash # faster start of mysql but no tab completion

[isamchk]

```

```
key_buffer                = 16M

#
# * IMPORTANT: Additional settings that can override those from this file!
#   The files must end with '.cnf', otherwise they'll be ignored.
#
!includedir /etc/mysql/conf.d/
```

SQL commands on the MariaDB prompt of *MASTER (db10)

After MariaDB login with "mysql -u root -p" follow the SQL commands below:

Listing 2: SQL commands on the MariaDB prompt

```
% # MariaDB [(none)]>
GRANT REPLICATION SLAVE, REPLICATION CLIENT ON *.* TO
    replicator@'192.168.10.150' IDENTIFIED BY 'somepassword';
flush privileges;
show master status;
```

Listing 3: SQL commands on the MariaDB prompt

```
% # Create an example database with one table for testing
% # -----
create database shop;
use shop

create table customer
(
    customer_id MEDIUMINT UNSIGNED NOT NULL AUTOINCREMENT PRIMARY KEY,
    title VARCHAR(5),
    last_name VARCHAR(32) NOT NULL,
    first_name VARCHAR(32) NOT NULL,
    address_street VARCHAR(32) NOT NULL,
    address_city VARCHAR(32) NOT NULL,
    address_country VARCHAR(24) NOT NULL
);

show tables;

insert into customer
(
    title , last_name , first_name , address_street , address_city , address_country
)
VALUES
(
    "Mr", "Iron", "Mark", "1-st Ironman Avenue 17", "80007 Santa Monica", "USA"
),
(
    "Miss", "Iron", "Murielle", "Ironwoman avenue 7", "10007 Paris", "France"
);

select * from customer;

show master status;
% RESET MASTER;
```

Shell commands on the operating system prompt of *MASTER (db10)

Listing 4: Shell commands on the operating system prompt of primary *MASTER

```
# Dump of databases of the primary replication master
```

```
root@db10:~#  
mysqldump -u root -p --all-databases --master-data=2 > dbdump-all.db  
mysqldump -u root -p shop --master-data=2 > dbdump-shop.db  
  
# * On debian / ubuntu "/etc/mysql/debian.cnf" of the master has to be copied to  
# the replication slave or the password-hash for debian-sys-maint should be set  
# in the mysql.user table on the slave after redumping of master's databases  
# to the local one.
```

Configuration of *SLAVE (db20) for MASTER-SLAVE replications

The my.cnf file on the *SLAVE should look like in the followin listing. After changes a database server restart is required (/etc/init.d/mysql restart).

Listing 5: my.cnf-MASTER-db10

```
# MariaDB database server configuration file.
#
# You can copy this file to one of:
# - "/etc/mysql/my.cnf" to set global options,
# - "~/.my.cnf" to set user-specific options.
#
# One can use all long options that the program supports.
# Run program with --help to get a list of available options and with
# --print-defaults to see which it would actually understand and use.
#
# For explanations see
# http://dev.mysql.com/doc/mysql/en/server-system-variables.html

# This will be passed to all mysql clients
# It has been reported that passwords should be enclosed with ticks/quotes
# especially if they contain "#" chars...
# Remember to edit /etc/mysql/debian.cnf when changing the socket location.
[client]
port                = 3306
socket              = /var/run/mysqld/mysqld.sock

# Here is entries for some specific programs
# The following values assume you have at least 32M ram

# This was formally known as [safe_mysqld]. Both versions are currently parsed.
[mysqld_safe]
socket              = /var/run/mysqld/mysqld.sock
nice                = 0

[mysqld]
#
# * Basic Settings
#
user                = mysql
pid-file            = /var/run/mysqld/mysqld.pid
socket              = /var/run/mysqld/mysqld.sock
port                = 3306
basedir             = /usr
datadir             = /var/lib/mysql
tmpdir              = /tmp
lc_messages_dir     = /usr/share/mysql
lc_messages         = en_US
skip-external-locking
#
# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
#bind-address        = 127.0.0.1
bind-address        = 192.168.10.150
#
# * Fine Tuning
#
max_connections     = 4000
connect_timeout     = 5
wait_timeout        = 600
max_allowed_packet  = 16M
thread_cache_size   = 128
sort_buffer_size    = 16M
```



```

bulk_insert_buffer_size = 16M
tmp_table_size          = 32M
max_heap_table_size     = 32M
#
# * MyISAM
#
# This replaces the startup script and checks MyISAM tables if needed
# the first time they are touched. On error, make copy and try a repair.
myisam-recover           = BACKUP
key_buffer_size          = 128M
#open-files--limit      = 2000
table_cache              = 400
myisam_sort_buffer_size  = 512M
concurrent_insert        = 2
read_buffer_size         = 16M
read_rnd_buffer_size     = 8M
#
# * Query Cache Configuration
#
# Cache only tiny result sets, so we can fit more in the query cache.
query_cache_limit        = 128K
query_cache_size         = 64M
# for more write intensive setups, set to DEMAND or OFF
#query_cache_type        = DEMAND
#
# * Logging and Replication
#
# Both location gets rotated by the cronjob.
# Be aware that this log type is a performance killer.
# As of 5.1 you can enable the log at runtime!
#general_log_file         = /var/log/mysql/mysql.log
#general_log              = 1
#
# Error logging goes to syslog due to /etc/mysql/conf.d/mysqld_safe_syslog.cnf.
#
# we do want to know about network errors and such
log_warnings             = 2
#
# Enable the slow query log to see queries with especially long duration
#slow_query_log[={0|1}]
slow_query_log_file      = /var/log/mysql/mariadb-slow.log
long_query_time = 10
#log_slow_rate_limit     = 1000
log_slow_verbosity       = query_plan

#log-queries--not--using--indexes
#log_slow_admin_statements
#
# The following can be used as easy to replay backup logs or for replication.
# note: if you are setting up a replication slave, see README.Debian about
#      other settings you may need to change.
server-id                = 20
#report_host              = master1
auto_increment_increment = 10
auto_increment_offset    = 2
log_bin                  = /var/log/mysql/mariadb-db20-bin
log_bin_index            = /var/log/mysql/mariadb-db20-bin.index
# not fab for performance, but safer
#sync_binlog              = 1
expire_logs_days         = 10
max_binlog_size           = 100M
# slaves
relay_log                = /var/log/mysql/db20-relay-bin

```

```

relay_log_index = /var/log/mysql/db20-relay-bin.index
relay_log_info_file = /var/log/mysql/db20-relay-bin.info
log_slave_updates
#read_only
#
# If applications support it, this stricter sql_mode prevents some
# mistakes like inserting invalid dates etc.
#sql_mode = NO_ENGINE_SUBSTITUTION,TRADITIONAL
#
# * InnoDB
#
# InnoDB is enabled by default with a 10MB datafile in /var/lib/mysql/.
# Read the manual for more InnoDB related options. There are many!
default_storage_engine = InnoDB
# you can't just change log file size, requires special procedure
#innodb_log_file_size = 50M
innodb_buffer_pool_size = 2048M
innodb_log_buffer_size = 8M
innodb_file_per_table = 1
innodb_open_files = 400
innodb_io_capacity = 400
innodb_flush_method = O_DIRECT
#
# * Security Features
#
# Read the manual, too, if you want chroot!
# chroot = /var/lib/mysql/
#
# For generating SSL certificates I recommend the OpenSSL GUI "tinyca".
#
# ssl-ca=/etc/mysql/cacert.pem
# ssl-cert=/etc/mysql/server-cert.pem
# ssl-key=/etc/mysql/server-key.pem

[mysqldump]
quick
quote-names
max_allowed_packet = 16M

[mysql]
#no-auto-rehash # faster start of mysql but no tab completion

[isamchk]
key_buffer = 16M

#
# * IMPORTANT: Additional settings that can override those from this file!
# The files must end with '.cnf', otherwise they'll be ignored.
#
!includedir /etc/mysql/conf.d/

```

Shell commands on the operating system prompt of *SLAVE (db20)

Listing 6: Shell commands on the operating system prompt of *SLAVE

```

# Dump of databases of the second database server before including the dump from *MASTER
# root@db20:~#
mysqldump -u root -p --all-databases --master-data=2 > dbdump-all-slave-fresh.db
mysql -u root -p shop < dbdump-shop.db

```

```
# * On debian / ubuntu "/etc/mysql/debian.cnf" of the master has to be copied to
# the replication slave or the password-hash for debian-sys-maint should be set
# in the mysql.user table on the slave after redumping of master's databases
# to the local one.
```

SQL commands on the MariaDB prompt of *SLAVE (db20)

After MariaDB login with "mysql -u root -p" follow the SQL commands below:

Listing 7: SQL commands on the MariaDB prompt of *SLAVE

```
% # MariaDB [(none)]>
CHANGE MASTER TO
    master_host='192.168.10.50',
    master_user='replicator',
    master_password='somepassword',
    master_log_file='mariadb-db10-bin.000001',
    master_log_pos=245;

START SLAVE;
SHOW STATUS SLAVE\G;
```

Configuration of MASTER-MASTER replications for MariaDB

MariaDB master-master replication mechanism allows replication of data between two master database servers. Every of them functions as slave for the other one. Replication is mostly used as a scale-out solution. The load is spreaded among both servers to improve performance and data availability. All writes, updates and reads can take place on both servers. Such model improves both the average write performance on each server (less reads take place on each of them) and the read performance as well.

After installation of ubuntu-server and MariaDB 5.5 the following steps are necessary:

- modify my.cnf files on db10 and db20 due to MASTER-MASTER replications requirements
- modify / extend settings of MariaDB on the SQL prompt
- create an example database with one table for testing

Configuration of 1st *MASTER (db10) for MASTER-MASTER replications

The my.cnf file on the primary *MASTER should look like in the followin listing. After changes a database server restart is required (/etc/init.d/mysql restart).

Listing 8: cat my.cnf-MASTER-MASTER-db10

```
# MariaDB database server configuration file.
#
# You can copy this file to one of:
# - "/etc/mysql/my.cnf" to set global options,
# - "~/.my.cnf" to set user-specific options.
#
# One can use all long options that the program supports.
# Run program with --help to get a list of available options and with
# --print-defaults to see which it would actually understand and use.
#
# For explanations see
# http://dev.mysql.com/doc/mysql/en/server-system-variables.html

# This will be passed to all mysql clients
# It has been reported that passwords should be enclosed with ticks/quotes
# especially if they contain "#" chars...
# Remember to edit /etc/mysql/debian.cnf when changing the socket location.
[client]
port                = 3306
socket              = /var/run/mysql/mysql.sock

# Here is entries for some specific programs
# The following values assume you have at least 32M ram

# This was formally known as [safe-mysqld]. Both versions are currently parsed.
[mysqld_safe]
socket              = /var/run/mysql/mysql.sock
nice               = 0

[mysqld]
#
# * Basic Settings
#
user                = mysql
pid-file            = /var/run/mysql/mysql.pid
socket              = /var/run/mysql/mysql.sock
port                = 3306
basedir             = /usr
datadir             = /var/lib/mysql
tmpdir              = /tmp
```

```

lc_messages_dir = /usr/share/mysql
lc_messages     = en_US
skip-external-locking
#
# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
#bind-address    = 127.0.0.1
bind-address     = 192.168.10.50
#
# * Fine Tuning
#
max_connections      = 4000
connect_timeout      = 5
wait_timeout         = 600
max_allowed_packet   = 16M
thread_cache_size    = 128
sort_buffer_size     = 16M
bulk_insert_buffer_size = 16M
tmp_table_size       = 32M
max_heap_table_size  = 32M
#
# * MyISAM
#
# This replaces the startup script and checks MyISAM tables if needed
# the first time they are touched. On error, make copy and try a repair.
myisam_recover       = BACKUP
key_buffer_size      = 128M
#open-files-limit    = 2000
table_cache          = 400
myisam_sort_buffer_size = 512M
concurrent_insert    = 2
read_buffer_size     = 16M
read_rnd_buffer_size = 8M
#
# * Query Cache Configuration
#
# Cache only tiny result sets, so we can fit more in the query cache.
query_cache_limit    = 128K
query_cache_size     = 64M
# for more write intensive setups, set to DEMAND or OFF
#query_cache_type     = DEMAND
#
# * Logging and Replication
#
# Both location gets rotated by the cronjob.
# Be aware that this log type is a performance killer.
# As of 5.1 you can enable the log at runtime!
#general_log_file     = /var/log/mysql/mysql.log
#general_log          = 1
#
# Error logging goes to syslog due to /etc/mysql/conf.d/mysqld_safe_syslog.cnf.
#
# we do want to know about network errors and such
log_warnings         = 2
#
# Enable the slow query log to see queries with especially long duration
#slow_query_log [= {0|1}]
slow_query_log_file   = /var/log/mysql/mariadb-slow.log
long_query_time = 10
#log_slow_rate_limit  = 1000
log_slow_verbosity    = query-plan
#log-queries-not-using-indexes

```

```

#log_slow_admin_statements
#
# The following can be used as easy to replay backup logs or for replication.
# note: if you are setting up a replication slave, see README.Debian about
#       other settings you may need to change.
server-id                = 10
#report_host              = master1
auto_increment_increment = 10
auto_increment_offset    = 1
log_bin                  = /var/log/mysql/mariadb-db10-bin
log_bin_index            = /var/log/mysql/mariadb-db10-bin.index
# not fab for performance, but safer
#sync_binlog              = 1
expire_logs_days         = 10
max_binlog_size          = 100M
# slaves
relay_log                = /var/log/mysql/db10-relay-bin
relay_log_index           = /var/log/mysql/db10-relay-bin.index
relay_log_info_file       = /var/log/mysql/db10-relay-bin.info
log_slave_updates
#read_only
#
# If applications support it, this stricter sql_mode prevents some
# mistakes like inserting invalid dates etc.
#sql_mode                 = NO_ENGINE_SUBSTITUTION,TRADITIONAL
#
# * InnoDB
#
# InnoDB is enabled by default with a 10MB datafile in /var/lib/mysql/.
# Read the manual for more InnoDB related options. There are many!
default_storage_engine   = InnoDB
# you can't just change log file size, requires special procedure
#innodb_log_file_size    = 50M
innodb_buffer_pool_size  = 2048M
innodb_log_buffer_size   = 8M
innodb_file_per_table    = 1
innodb_open_files        = 400
innodb_io_capacity       = 400
innodb_flush_method      = O_DIRECT
#
# * Security Features
#
# Read the manual, too, if you want chroot!
# chroot = /var/lib/mysql/
#
# For generating SSL certificates I recommend the OpenSSL GUI "tinyca".
#
# ssl-ca=/etc/mysql/cacert.pem
# ssl-cert=/etc/mysql/server-cert.pem
# ssl-key=/etc/mysql/server-key.pem

[mysqldump]
quick
quote-names
max_allowed_packet       = 16M

[mysql]
#no-auto-rehash # faster start of mysql but no tab completion

[isamchk]
key_buffer                = 16M

```

```
#
# * IMPORTANT: Additional settings that can override those from this file!
#   The files must end with '.cnf', otherwise they'll be ignored.
#
!includedir /etc/mysql/conf.d/
```

Shell commands on the operating system prompt of primary *MASTER (db10)

Listing 9: Shell commands on the operating system prompt of primary *MASTER

```
# Dump of databases of the primary database server before including the dump from *MASTER
# root@db20:~#
mysqldump -u root -p --all-databases --master-data=2 > dbdump-all-2nd-master-fresh.db
mysql -u root -p shop < dbdump-shop.db

# * On debian / ubuntu "/etc/mysql/debian.cnf" of the master has to be copied to
#   the replication slave or the password-hash for debian-sys-maint should be set
#   in the mysql.user table on the slave after redumping of master's databases
#   to the local one.
```

SQL commands on the MariaDB prompt of primary *MASTER (db10)

After MariaDB login with "mysql -u root -p" follow the SQL commands below:

Listing 10: SQL commands on the MariaDB prompt of *SLAVE

```
% # MariaDB [(none)]>

% # MASTER part of MASTER-MASTER configuration
% # -----
GRANT REPLICATION SLAVE, REPLICATION CLIENT ON *.* TO
    replicator@'192.168.10.150' IDENTIFIED BY 'somepassword';
FLUSH PRIVILEGES;
SHOW MASTER STATUS;

% # SLAVE part of MASTER-MASTER configuration
% # -----
CHANGE MASTER TO
    master_host='192.168.10.150',
    master_user='replicator',
    master_password='somepassword',
    master_log_file='mariadb-db20-bin.000001',
    master_log_pos=245;

START SLAVE;
SHOW STATUS SLAVE\G;
```

Configuration of 2nd *MASTER (db20) for MASTER-MASTER replications

The my.cnf file on the second *MASTER should look like in the followin listing. After changes a database server restart is required (/etc/init.d/mysql restart).

Listing 11: cat my.cnf-MASTER-MASTER-db20

```
# MariaDB database server configuration file.
#
# You can copy this file to one of:
# - "/etc/mysql/my.cnf" to set global options,
# - "~/.my.cnf" to set user-specific options.
#
# One can use all long options that the program supports.
# Run program with --help to get a list of available options and with
# --print-defaults to see which it would actually understand and use.
#
# For explanations see
# http://dev.mysql.com/doc/mysql/en/server-system-variables.html

# This will be passed to all mysql clients
# It has been reported that passwords should be enclosed with ticks/quotes
# especially if they contain "#" chars...
# Remember to edit /etc/mysql/debian.cnf when changing the socket location.
[client]
port                = 3306
socket              = /var/run/mysqld/mysqld.sock

# Here is entries for some specific programs
# The following values assume you have at least 32M ram

# This was formally known as [safe_mysqld]. Both versions are currently parsed.
[mysqld_safe]
socket              = /var/run/mysqld/mysqld.sock
nice                = 0

[mysqld]
#
# * Basic Settings
#
user                = mysql
pid-file            = /var/run/mysqld/mysqld.pid
socket              = /var/run/mysqld/mysqld.sock
port                = 3306
basedir             = /usr
datadir             = /var/lib/mysql
tmpdir              = /tmp
lc_messages_dir     = /usr/share/mysql
lc_messages         = en_US
skip-external-locking
#
# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
#bind-address        = 127.0.0.1
bind-address        = 192.168.10.150
#
# * Fine Tuning
#
max_connections     = 4000
connect_timeout     = 5
wait_timeout        = 600
max_allowed_packet  = 16M
thread_cache_size   = 128
sort_buffer_size    = 16M
```



```

bulk_insert_buffer_size = 16M
tmp_table_size          = 32M
max_heap_table_size     = 32M
#
# * MyISAM
#
# This replaces the startup script and checks MyISAM tables if needed
# the first time they are touched. On error, make copy and try a repair.
myisam-recover           = BACKUP
key_buffer_size          = 128M
#open-files--limit      = 2000
table_cache              = 400
myisam_sort_buffer_size  = 512M
concurrent_insert        = 2
read_buffer_size         = 16M
read_rnd_buffer_size     = 8M
#
# * Query Cache Configuration
#
# Cache only tiny result sets, so we can fit more in the query cache.
query_cache_limit        = 128K
query_cache_size         = 64M
# for more write intensive setups, set to DEMAND or OFF
#query_cache_type        = DEMAND
#
# * Logging and Replication
#
# Both location gets rotated by the cronjob.
# Be aware that this log type is a performance killer.
# As of 5.1 you can enable the log at runtime!
#general_log_file        = /var/log/mysql/mysql.log
#general_log              = 1
#
# Error logging goes to syslog due to /etc/mysql/conf.d/mysqld_safe_syslog.cnf.
#
# we do want to know about network errors and such
log_warnings             = 2
#
# Enable the slow query log to see queries with especially long duration
#slow_query_log[={0|1}]
slow_query_log_file      = /var/log/mysql/mariadb-slow.log
long_query_time = 10
#log_slow_rate_limit     = 1000
log_slow_verbosity       = query_plan

#log-queries--not--using--indexes
#log_slow_admin_statements
#
# The following can be used as easy to replay backup logs or for replication.
# note: if you are setting up a replication slave, see README.Debian about
#      other settings you may need to change.
server-id                = 20
#report_host              = master1
auto_increment_increment = 10
auto_increment_offset    = 2
log_bin                  = /var/log/mysql/mariadb-db20-bin
log_bin_index            = /var/log/mysql/mariadb-db20-bin.index
# not fab for performance, but safer
#sync_binlog              = 1
expire_logs_days         = 10
max_binlog_size           = 100M
# slaves
relay_log                = /var/log/mysql/db20-relay-bin

```

```

relay_log_index          = /var/log/mysql/db20-relay-bin.index
relay_log_info_file      = /var/log/mysql/db20-relay-bin.info
log_slave_updates
#read_only
#
# If applications support it, this stricter sql_mode prevents some
# mistakes like inserting invalid dates etc.
#sql_mode                = NO_ENGINE_SUBSTITUTION,TRADITIONAL
#
# * InnoDB
#
# InnoDB is enabled by default with a 10MB datafile in /var/lib/mysql/.
# Read the manual for more InnoDB related options. There are many!
default_storage_engine   = InnoDB
# you can't just change log file size, requires special procedure
#innodb_log_file_size    = 50M
innodb_buffer_pool_size  = 2048M
innodb_log_buffer_size   = 8M
innodb_file_per_table    = 1
innodb_open_files        = 400
innodb_io_capacity       = 400
innodb_flush_method      = O_DIRECT
#
# * Security Features
#
# Read the manual, too, if you want chroot!
# chroot = /var/lib/mysql/
#
# For generating SSL certificates I recommend the OpenSSL GUI "tinyca".
#
# ssl-ca=/etc/mysql/cacert.pem
# ssl-cert=/etc/mysql/server-cert.pem
# ssl-key=/etc/mysql/server-key.pem

[mysqldump]
quick
quote-names
max_allowed_packet       = 16M

[mysql]
#no-auto-rehash # faster start of mysql but no tab completion

[isamchk]
key_buffer               = 16M

#
# * IMPORTANT: Additional settings that can override those from this file!
# The files must end with '.cnf', otherwise they'll be ignored.
#
!includedir /etc/mysql/conf.d/

```

Shell commands on the operating system prompt of second *MASTER (db20)

Listing 12: Shell commands on the operating system prompt of second *MASTER

```

# Dump of databases of the second database server before including the dump from *MASTER
# root@db20:~#
mysqldump -u root -p --all-databases --master-data=2 > dbdump-all-2nd-master-fresh.db
mysql -u root -p shop < dbdump-shop.db

```

```
# * On debian / ubuntu "/etc/mysql/debian.cnf" of the master has to be copied to
# the replication slave or the password-hash for debian-sys-maint should be set
# in the mysql.user table on the slave after redumping of master's databases
# to the local one.
```

SQL commands on the MariaDB prompt of second *MASTER (db20)

After MariaDB login with "mysql -u root -p" follow the SQL commands below:

Listing 13: SQL commands on the MariaDB prompt of *SLAVE

```
% # MariaDB [(none)]>

% # MASTER part of MASTER-MASTER configuration
% # -----
GRANT REPLICATION SLAVE, REPLICATION CLIENT ON *.* TO
    replicator@'192.168.10.50' IDENTIFIED BY 'somepassword';
FLUSH PRIVILEGES;
SHOW MASTER STATUS;

% # SLAVE part of MASTER-MASTER configuration
% # -----
CHANGE MASTER TO
    master_host='192.168.10.50',
    master_user='replicator',
    master_password='somepassword',
    master_log_file='mariadb-db10-bin.000001',
    master_log_pos=245;

START SLAVE;
SHOW STATUS SLAVE\G;

% # Some test statements to check if changes are replicated
% # to the other database server (db10)
SELECT * FROM shop.customer;

INSERT INTO shop.customeshop.customer
(
    title, last_name, first_name, address_street, address_city, address_country
)
VALUES
(
    "Mrs", "Shuli", "Guo", "Mao street 57", "10000 Beijing", "China"
)

SELECT * FROM shop.customer;

DELETE FROM shop.customer WHERE customer_id = 1;
```

APPENDIX

diff of my.cnf config files for MASTER-SLAVE replications

Listing 14: diff for MASTER-SLAVE replications

```
# diff -aur my.cnf-ORIG my.cnf-MASTER-db10 | grep "^[+-]"
--- my.cnf-ORIG 2012-06-02 00:14:15.010071060 +0200
+++ my.cnf-MASTER-db10 2012-06-02 00:22:37.400562369 +0200
-bind-address          = 127.0.0.1
+bind-address          = 192.168.10.50
-max_connections       = 100
+max_connections       = 4000
-sort_buffer_size      = 4M
+sort_buffer_size      = 16M
-read_buffer_size      = 2M
-read_rnd_buffer_size  = 1M
+read_buffer_size      = 16M
+read_rnd_buffer_size  = 8M
+server-id             = 10
-log_bin               = /var/log/mysql/mariadb-bin
-log_bin_index         = /var/log/mysql/mariadb-bin.index
+auto_increment_increment = 10
+auto_increment_offset = 1
+log_bin               = /var/log/mysql/mariadb-db10-bin
+log_bin_index         = /var/log/mysql/mariadb-db10-bin.index
-innodb_buffer_pool_size = 256M
+innodb_buffer_pool_size = 2048M

# diff -aur my.cnf-ORIG my.cnf-SLAVE-db20 | grep "^[+-]"
--- my.cnf-ORIG 2012-06-02 00:14:15.010071060 +0200
+++ my.cnf-SLAVE-db20 2012-06-02 00:22:49.792623664 +0200
-bind-address          = 127.0.0.1
+bind-address          = 192.168.10.150
-max_connections       = 100
+max_connections       = 4000
-sort_buffer_size      = 4M
+sort_buffer_size      = 16M
-read_buffer_size      = 2M
-read_rnd_buffer_size  = 1M
+read_buffer_size      = 16M
+read_rnd_buffer_size  = 8M
+server-id             = 20
-log_bin               = /var/log/mysql/mariadb-bin
-log_bin_index         = /var/log/mysql/mariadb-bin.index
+auto_increment_increment = 10
+auto_increment_offset = 2
+log_bin               = /var/log/mysql/mariadb-db20-bin
+log_bin_index         = /var/log/mysql/mariadb-db20-bin.index
+relay_log             = /var/log/mysql/db20-relay-bin
+relay_log_index       = /var/log/mysql/db20-relay-bin.index
+relay_log_info_file   = /var/log/mysql/db20-relay-bin.info
+log_slave_updates
-innodb_buffer_pool_size = 256M
+innodb_buffer_pool_size = 2048M
```

diff of my.cnf config files for MASTER-MASTER replications

Listing 15: diff for MASTER-MASTER replications

```
# diff -aur my.cnf-ORIG my.cnf-MASTER-MASTER-db10 | grep "^[+-][^#]"
--- my.cnf-ORIG 2012-06-02 00:14:15.010071060 +0200
+++ my.cnf-MASTER-MASTER-db10 2012-06-02 00:13:15.573776254 +0200
-bind-address = 127.0.0.1
+bind-address = 192.168.10.50
-max_connections = 100
+max_connections = 4000
-sort_buffer_size = 4M
+sort_buffer_size = 16M
-read_buffer_size = 2M
-read_rnd_buffer_size = 1M
+read_buffer_size = 16M
+read_rnd_buffer_size = 8M
+server-id = 10
-log_bin = /var/log/mysql/mariadb-bin
-log_bin_index = /var/log/mysql/mariadb-bin.index
+auto_increment_increment = 10
+auto_increment_offset = 1
+log_bin = /var/log/mysql/mariadb-db10-bin
+log_bin_index = /var/log/mysql/mariadb-db10-bin.index
+relay_log = /var/log/mysql/db10-relay-bin
+relay_log_index = /var/log/mysql/db10-relay-bin.index
+relay_log_info_file = /var/log/mysql/db10-relay-bin.info
+log_slave_updates
-innodb_buffer_pool_size = 256M
+innodb_buffer_pool_size = 2048M

# diff -aur my.cnf-ORIG my.cnf-MASTER-MASTER-db20 | grep "^[+-]"
--- my.cnf-ORIG 2012-06-02 00:14:15.010071060 +0200
+++ my.cnf-MASTER-MASTER-db20 2012-06-02 00:13:27.361834633 +0200
-bind-address = 127.0.0.1
+bind-address = 192.168.10.150
-max_connections = 100
+max_connections = 4000
-sort_buffer_size = 4M
+sort_buffer_size = 16M
-read_buffer_size = 2M
-read_rnd_buffer_size = 1M
+read_buffer_size = 16M
+read_rnd_buffer_size = 8M
+server-id = 20
-log_bin = /var/log/mysql/mariadb-bin
-log_bin_index = /var/log/mysql/mariadb-bin.index
+auto_increment_increment = 10
+auto_increment_offset = 2
+log_bin = /var/log/mysql/mariadb-db20-bin
+log_bin_index = /var/log/mysql/mariadb-db20-bin.index
+relay_log = /var/log/mysql/db20-relay-bin
+relay_log_index = /var/log/mysql/db20-relay-bin.index
+relay_log_info_file = /var/log/mysql/db20-relay-bin.info
+log_slave_updates
-innodb_buffer_pool_size = 256M
+innodb_buffer_pool_size = 2048M
```