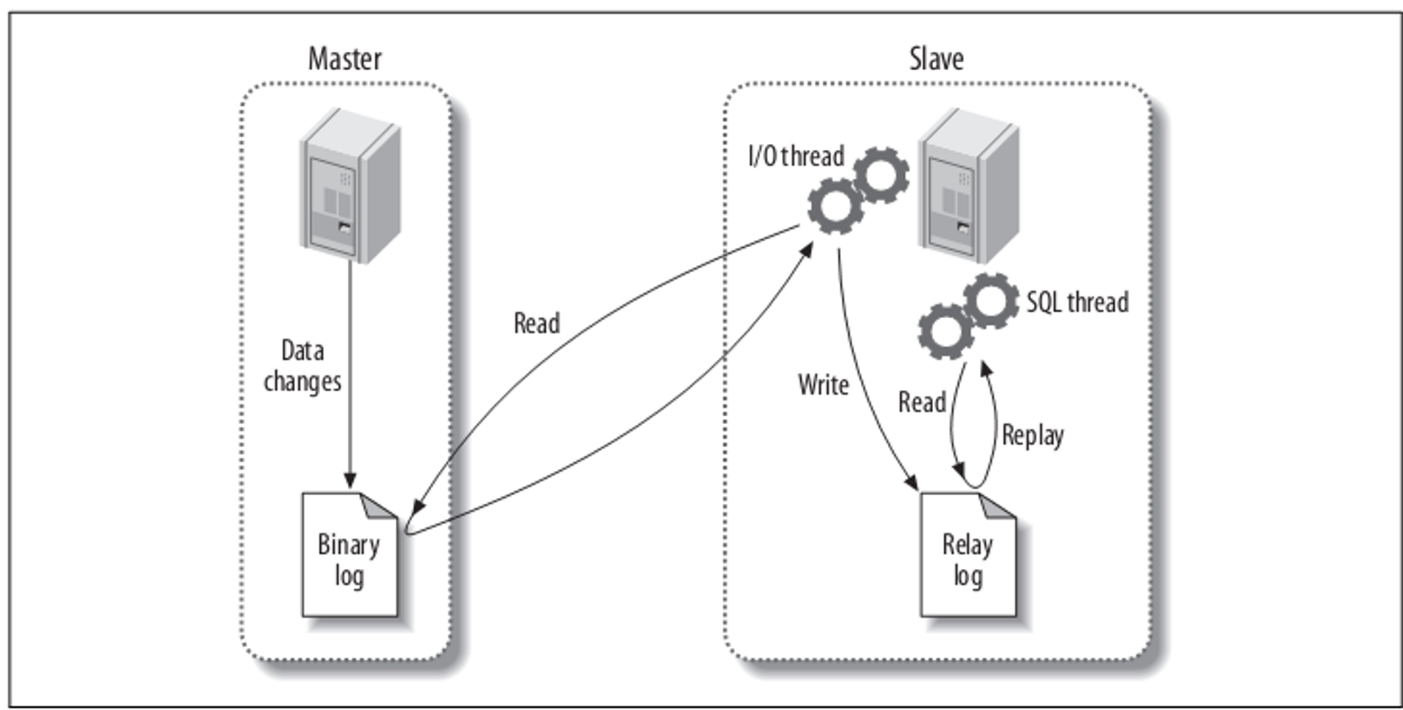
**Replicação de dados MySql 10.5**



**\*\*Criando conta de Replicação\*\***

#Primeiramente deve-se criar uma conta de usuário com permissão de replicação para os servidores

#tanto no master quanto no slave, ou seja,

#o mesmo usuário criado para os servidores em questão. Aqui está como criar a conta de usuário,

#que chamaremos de \*replicacao\*:

#Criar usuario que será usado na replicação

- CREATE USER REPLICACAO IDENTIFIED BY 'SENHA';

#Dar permissão para a replicação acontecer

- GRANT REPLICATION SLAVE ON . TO 'replicacao'@'%';

#Para testar se está funcionando o usuário você pode tentar entrar no mysql com o usuario de replicação

#Foi criada esta conta de usuario no servidor master e no slave. Log em seguida foi dado as permissões

#de replicação (grant replication slave, replication client) para todas as bases de dados.

**\*\*Configurando o Master\*\***

#Depois de criado as contas o próximo passo é ativar algumas configurações no master.

#É preciso ativar o log binário e especificar um ID do servidor. Este passo é feito no aquivo de

#configuração do mysql o “my.cnf” dentro do diretório “/etc”, para isso insira as seguintes linhas no arquivo:

(para debian) nano /etc/mysql/mariadb.conf.d/50-server.cnf

#Foi colocado as linhas abaixo:

-------------------------------------------------------

server-id = 1

user = mysql

pid-file = /run/mysqld/mysqld.pid

socket = /var/run/mysqld/mysqld.sock

basedir = /usr

datadir = /var/lib/mysql

tmpdir = /tmp

lc-messages-dir = /usr/share/mysql

lc-messages = en\_US

skip-external-locking

log\_bin = /var/log/mysql/mysql-bin.log

log\_bin\_index =/var/log/mysql/mysql-bin.log.index

relay\_log = /var/log/mysql/mysql-relay-bin

relay\_log\_index = /var/log/mysql/mysql-relay-bin.index

-------------------------------------------------------

#E comente estas 2 linhas abaixo para que o log que o slave irá ler nunca expire ou

#Tenha algum limite de tamanho.

#expire\_logs\_days = 10

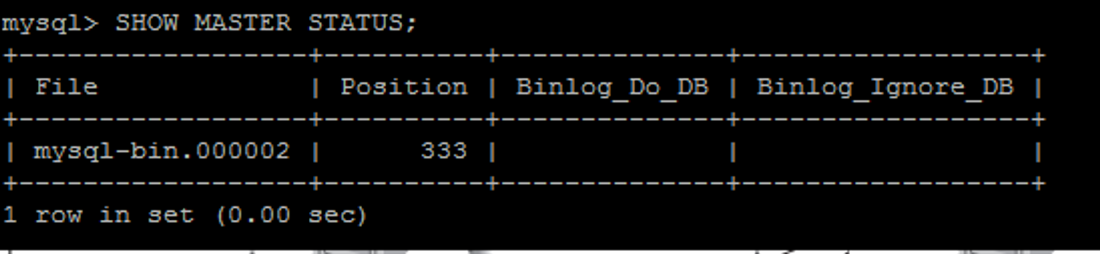
#max\_binlog\_size = 100M

#Depois de realizado essa configuração no master é preciso reiniciar o mysql para o

#servidor ativar as configurações e criar os logs binários.

- systemctl restart mysql

#Para verificar se o arquivo de log binário está criado no master execute o seguinte comando SHOW MASTER STATUS;



**\*\*\*CONFIGURANDO O SLAVE\*\*\***

#Seguindo no processo de configuração da replicação de dados no mysql, o servidor slave precisa ser configurado

#pelo arquivo de configuração do mysql (my.cnf), para isso insira as seguintes linhas no arquivo:

(para debian) nano /etc/mysql/mariadb.conf.d/50-server.cnf

Fazer estas alterações abaixo:

log-bin = mysql-bin

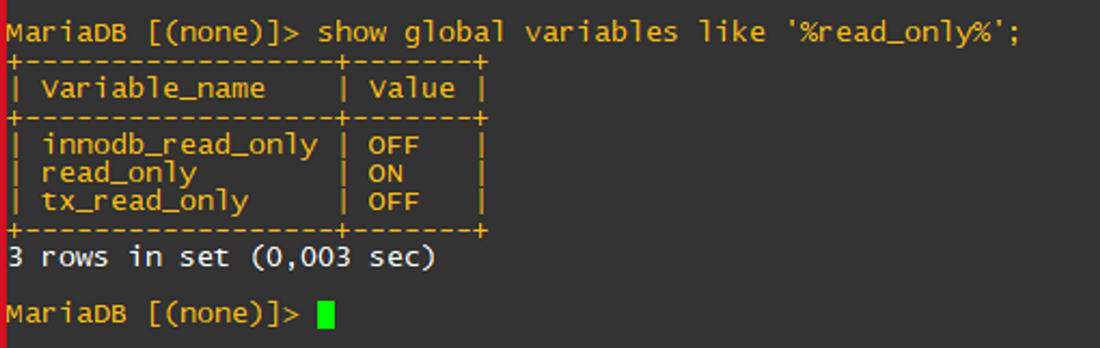
server-id = 2

relay-log = mysql-relay-bin

log-slave-updates = 1

read\_only = 1

- systemctl restart mysql



**\*\*\*PRIMEIRA CONF DO BANCO SLAVE\*\*\***

#Agora precisamos tirar um dump do banco e subir lá no slave para que ele tenha o banco e as tabelas para escrever

os dados que ele recebe pelo bin.log

**\*\*\*TIRANDO DUMP DO BANCO ATUAL\*\*\***

- mysqldump -p -uroot --master\_data --routines --events --single\_transaction --databases zabbix --result-file arquivo\_dump.sql

**\*\*\*TRANSFERIR O DUMP PRO SLAVE\*\*\***

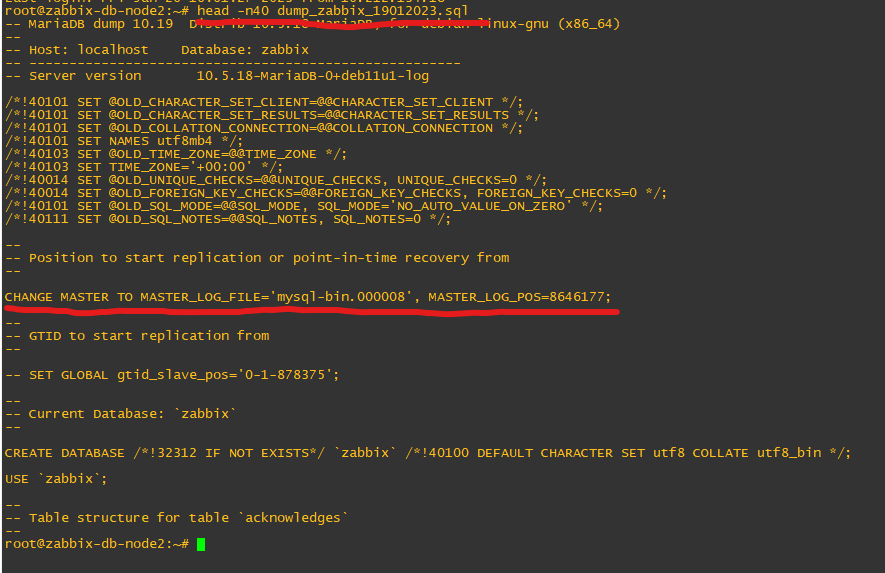
- scp arquivo\_dump.sql root@IP\_SLAVE:/root/

**\*\*\*CONFIGURANDO REPLICA NO SLAVE\*\*\***

#Para podermos fazer a leitura dos bin logs na replicação precisamos saber qual é o MASTER\_LOG\_FILE e MASTER\_LOG\_POS.

#Essas informações ficam no arquivo de dump e podem ser encontradas executando o comando:

- head -n40 arquivo\_dump.sql



#Anote esses dados pois vamos usar e cadastra-los no usuario em seguida

CHANGE MASTER TO MASTER\_LOG\_FILE='mysql-bin.000008', MASTER\_LOG\_POS=8646177;

**\*\*\*SUBINDO O BANCO NO SLAVE\*\*\***

mysql -uroot < arquivo\_dump.sql -p

\*\*\*VALIDAR SE O DUMP FICOU COM O MESMO TAMANHO\*\*\*

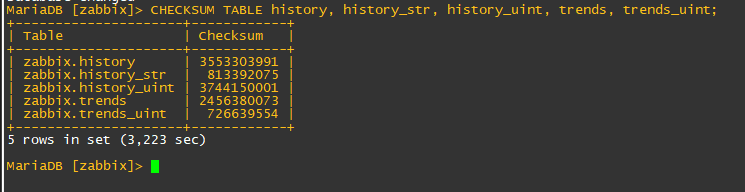
Entre no mysql

Entre na database aqui estou fazendo o teste na zabbix;

Mysql –uroot –p

Use zabbix;

CHECKSUM TABLE history, history\_str, history\_uint, trends, trends\_uint;



**\*\*INICIANDO O SERVIDOR SLAVE\*\***

#Para finalizar a configuração, o próximo passo é dizer ao slave como conectar-se ao master e começar a replicação.

#Aqui é onde devemos colocar as informações de MASTER\_LOG\_FILE e MASTER\_LOG\_POS que pegamos vendo o head do dump no slave

mysql> CHANGE MASTER TO MASTER\_HOST='seu banco',

-> MASTER\_USER='replicacao',

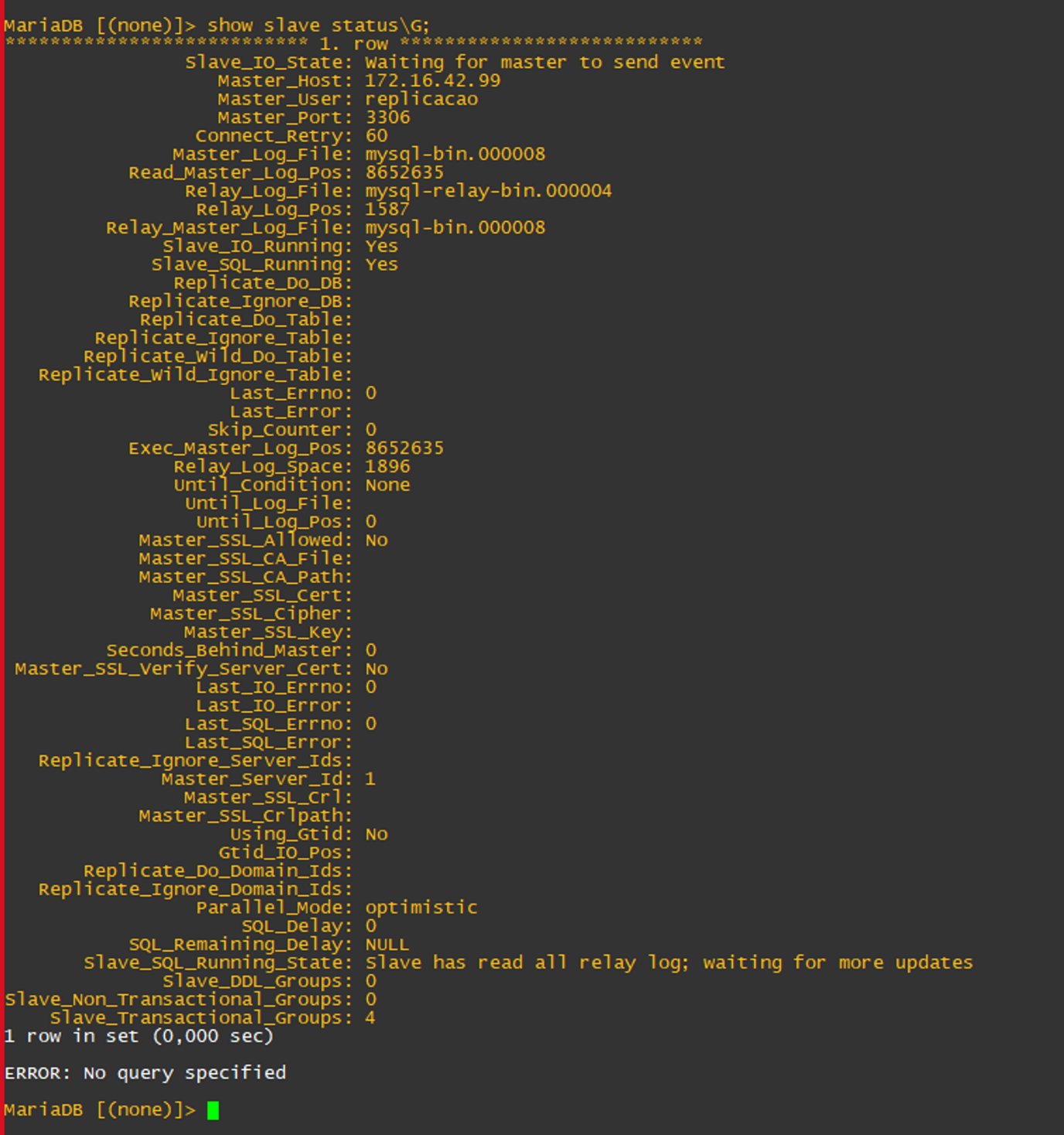
-> MASTER\_PASSWORD='senha',

-> MASTER\_LOG\_FILE='mysql-bin.000001',

-> MASTER\_LOG\_POS=0;

- START SLAVE;

- SHOW SLAVE STATUS\G;



#Pronto, se seu show slave status aparecer igualo de cima, a replicação estará pronta, caso contrario

Outros comandos para troubleshooting:

==================

SHOW BINARY LOGS;

SHOW SLAVE STATUS;

SHOW MASTER STATUS\G

RESET MASTER; ## removes all binary log files that are listed in the index file, leaving only a single, empty binary log file with a numeric suffix of .000001

RESET MASTER TO 1234; ## reset to specific binary log position

PURGE BINARY LOGS BEFORE '2019-10-11 00:20:00';## Numbering is not reset, may be safely used while replication## slaves are running.

FLUSH BINARY LOGS; ## Will reset state of binary logs and restarts numbering

==================