KONFIGURASI DATABASE MARIADB PADA MASTER - SLAVE SISTEM TERDISTRIBUSI

OLEH :

NAMA : MUTIAH ANDINI

NIM : 09011182126027

KELAS : SK6A

DOSEN PENGAMPUH : ADI HERMANSYAH, M.T

FAKULTAS ILMU KOMPUTER
SISTEM KOMPUTER
UNIVERSITAS SRIWIJAYA TAHUN AJARAN 2024

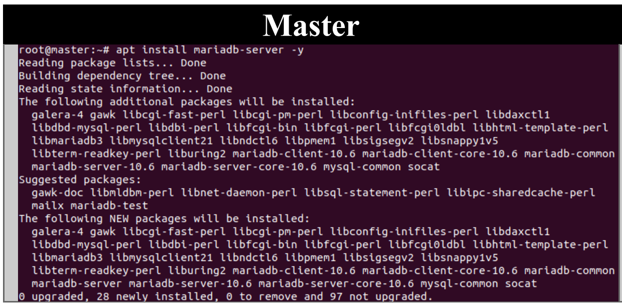
Langkah-langkah :

1. IP

Master	Slave
192.168.65.152	192.168.65.153

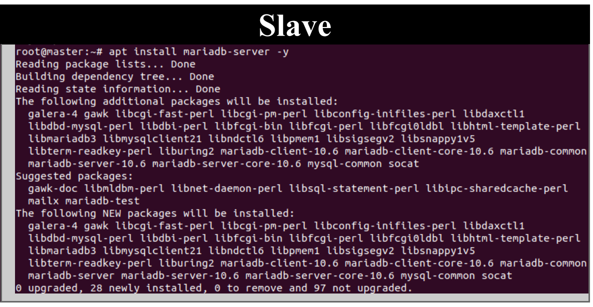
2. Install MariaDB

Master



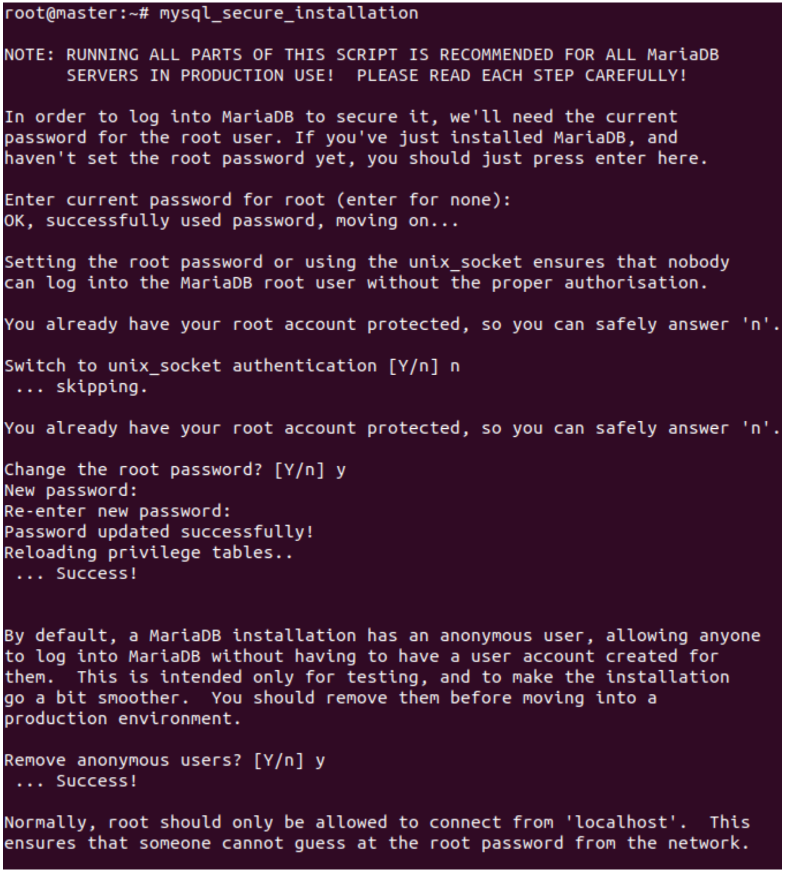
```
root@master:~# apt install mariadb-server -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  galera-4 gawk libcgi-fast-perl libcgi-pm-perl libconfig-inifiles-perl libdaxctl1
  libdbd-mysql-perl libdbi-perl libfcgi-bin libfcgi-perl libfcgi0ldbl libhtml-template-perl
  libmariadb3 libmysqlclient21 libndctl6 libpmem1 libsigsegv2 libsnappy1v5
  libterm-readkey-perl liburing2 mariadb-client-10.6 mariadb-client-core-10.6 mariadb-common
  mariadb-server-10.6 mariadb-server-core-10.6 mysql-common socat
Suggested packages:
  gawk-doc libmldbm-perl libnet-daemon-perl libsql-statement-perl libipc-sharedcache-perl
  mailx mariadb-test
The following NEW packages will be installed:
  galera-4 gawk libcgi-fast-perl libcgi-pm-perl libconfig-inifiles-perl libdaxctl1
  libdbd-mysql-perl libdbi-perl libfcgi-bin libfcgi-perl libfcgi0ldbl libhtml-template-perl
  libmariadb3 libmysqlclient21 libndctl6 libpmem1 libsigsegv2 libsnappy1v5
  libterm-readkey-perl liburing2 mariadb-client-10.6 mariadb-client-core-10.6 mariadb-common
  mariadb-server mariadb-server-10.6 mariadb-server-core-10.6 mysql-common socat
0 upgraded, 28 newly installed, 0 to remove and 97 not upgraded.
```

Slave

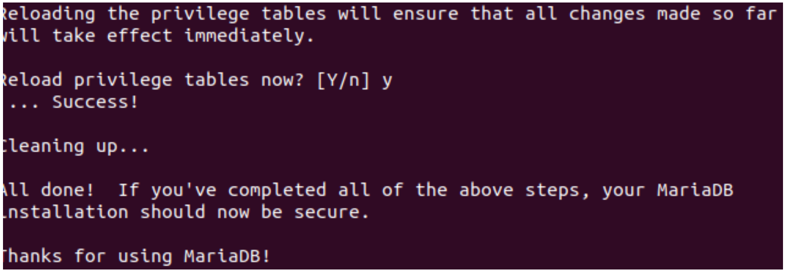


```
root@master:~# apt install mariadb-server -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  galera-4 gawk libcgi-fast-perl libcgi-pm-perl libconfig-inifiles-perl libdaxctl1
  libdbd-mysql-perl libdbi-perl libfcgi-bin libfcgi-perl libfcgi0ldbl libhtml-template-perl
  libmariadb3 libmysqlclient21 libndctl6 libpmem1 libsigsegv2 libsnappy1v5
  libterm-readkey-perl liburing2 mariadb-client-10.6 mariadb-client-core-10.6 mariadb-common
  mariadb-server-10.6 mariadb-server-core-10.6 mysql-common socat
Suggested packages:
  gawk-doc libmldbm-perl libnet-daemon-perl libsql-statement-perl libipc-sharedcache-perl
  mailx mariadb-test
The following NEW packages will be installed:
  galera-4 gawk libcgi-fast-perl libcgi-pm-perl libconfig-inifiles-perl libdaxctl1
  libdbd-mysql-perl libdbi-perl libfcgi-bin libfcgi-perl libfcgi0ldbl libhtml-template-perl
  libmariadb3 libmysqlclient21 libndctl6 libpmem1 libsigsegv2 libsnappy1v5
  libterm-readkey-perl liburing2 mariadb-client-10.6 mariadb-client-core-10.6 mariadb-common
  mariadb-server mariadb-server-10.6 mariadb-server-core-10.6 mysql-common socat
0 upgraded, 28 newly installed, 0 to remove and 97 not upgraded.
```

3. Install mysql MariaDB pada Master-Slave

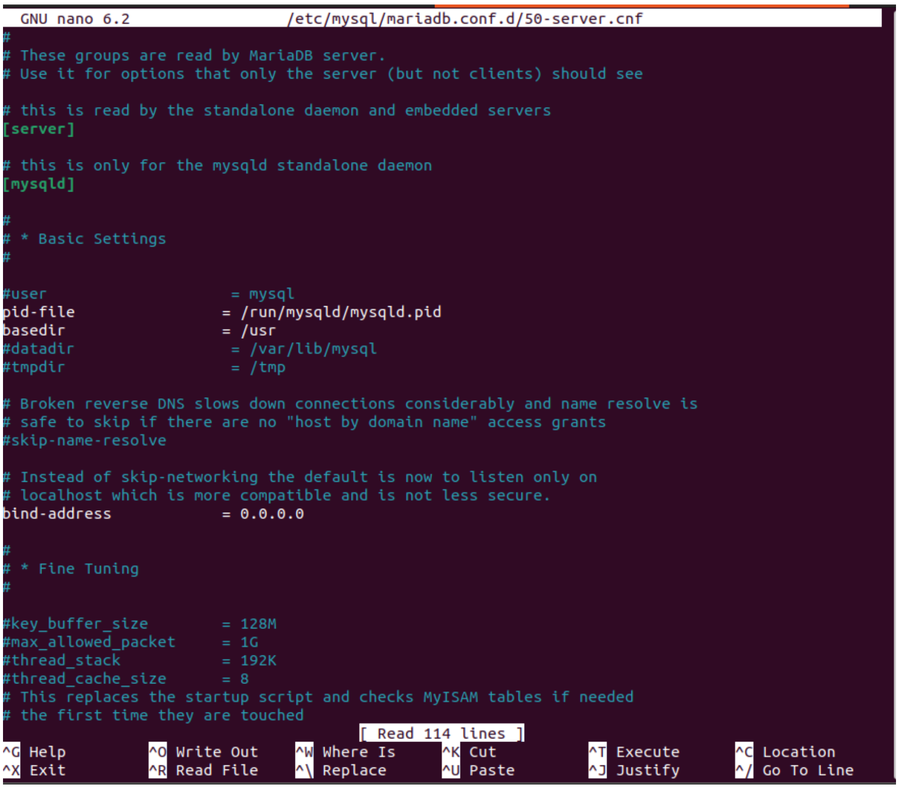


```
root@master:~# mysql_secure_installation

NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB
      SERVERS IN PRODUCTION USE!  PLEASE READ EACH STEP CAREFULLY!

In order to log into MariaDB to secure it, we'll need the current
password for the root user. If you've just installed MariaDB, and
haven't set the root password yet, you should just press enter here.

Enter current password for root (enter for none):
OK, successfully used password, moving on...

Setting the root password or using the unix_socket ensures that nobody
can log into the MariaDB root user without the proper authorisation.

You already have your root account protected, so you can safely answer 'n'.

Switch to unix_socket authentication [Y/n] n
 ... skipping.

You already have your root account protected, so you can safely answer 'n'.

Change the root password? [Y/n] y
New password:
Re-enter new password:
Password updated successfully!
Reloading privilege tables..
 ... Success!


By default, a MariaDB installation has an anonymous user, allowing anyone
to log into MariaDB without having to have a user account created for
them.  This is intended only for testing, and to make the installation
go a bit smoother.  You should remove them before moving into a
production environment.

Remove anonymous users? [Y/n] y
 ... Success!

Normally, root should only be allowed to connect from 'localhost'.  This
ensures that someone cannot guess at the root password from the network.
```



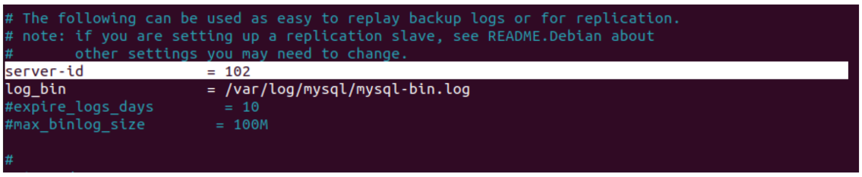
```
eloading the privilege tables will ensure that all changes made so far
ill take effect immediately.

eload privilege tables now? [Y/n] y
... Success!

leaning up...

ll done!  If you've completed all of the above steps, your MariaDB
nstallation should now be secure.

hanks for using MariaDB!
```

4. Konfigurasi MariaDB



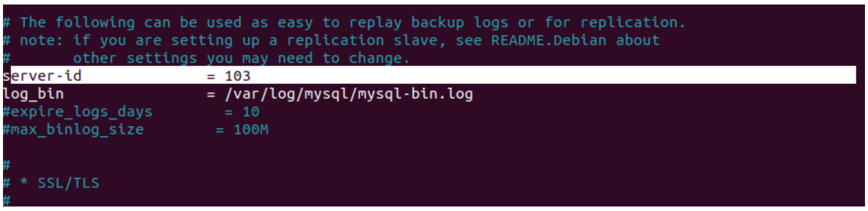
```
  GNU nano 6.2                  /etc/mysql/mariadb.conf.d/50-server.cnf
#
# These groups are read by MariaDB server.
# Use it for options that only the server (but not clients) should see

# this is read by the standalone daemon and embedded servers
[server]

# this is only for the mysqld standalone daemon
[mysqld]

#
# * Basic Settings
#

#user                    = mysql
pid-file                = /run/mysqld/mysqld.pid
basedir                 = /usr
#datadir                 = /var/lib/mysql
#tmpdir                  = /tmp

# Broken reverse DNS slows down connections considerably and name resolve is
# safe to skip if there are no "host by domain name" access grants
#skip-name-resolve

# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
bind-address            = 0.0.0.0

#
# * Fine Tuning
#

#key_buffer_size        = 128M
#max_allowed_packet     = 1G
#thread_stack           = 192K
#thread_cache_size      = 8
# This replaces the startup script and checks MyISAM tables if needed
# the first time they are touched
                                       [ Read 114 lines ]
^G Help          ^O Write Out     ^W Where Is      ^K Cut           ^T Execute       ^C Location
^X Exit          ^R Read File     ^\ Replace       ^U Paste         ^J Justify       ^/ Go To Line
```

- Master



```
# 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               = 102
log_bin                 = /var/log/mysql/mysql-bin.log
#expire_logs_days        = 10
#max_binlog_size        = 100M

#
```

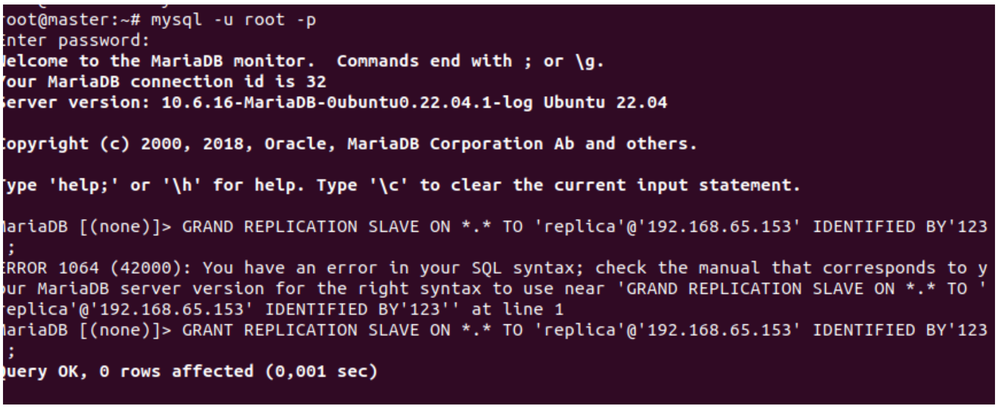
- Slave



```
# 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               = 103
log_bin                 = /var/log/mysql/mysql-bin.log
#expire_logs_days        = 10
#max_binlog_size        = 100M

#
# * SSL/TLS
#
```

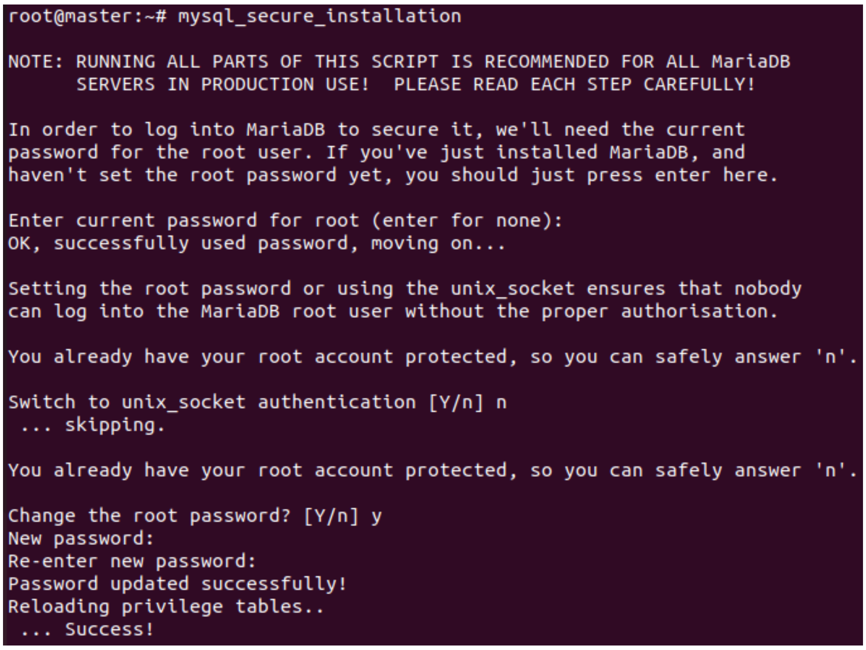
5. Akses ke Database MariaDB

- Master



```
oot@master:~# mysql -u root -p
:nter password:
Jelcome to the MariaDB monitor.  Commands end with ; or \g.
'our MariaDB connection id is 32
erver version: 10.6.16-MariaDB-0ubuntu0.22.04.1-log Ubuntu 22.04

:opyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

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

lariaDB [(none)]> GRAND REPLICATION SLAVE ON *.* TO 'replica'@'192.168.65.153' IDENTIFIED BY'123
;
:RROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to y
ur MariaDB server version for the right syntax to use near 'GRAND REPLICATION SLAVE ON *.* TO '
eplica'@'192.168.65.153' IDENTIFIED BY'123'' at line 1
lariaDB [(none)]> GRANT REPLICATION SLAVE ON *.* TO 'replica'@'192.168.65.153' IDENTIFIED BY'123
;
}uery OK, 0 rows affected (0,001 sec)
```

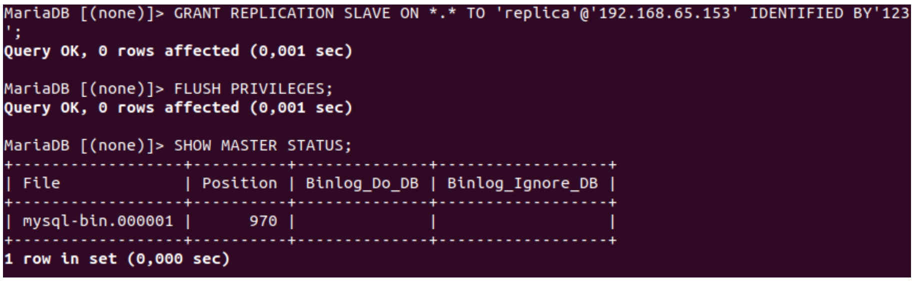
- Slave



```
root@master:~# mysql_secure_installation

NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB
      SERVERS IN PRODUCTION USE!  PLEASE READ EACH STEP CAREFULLY!

In order to log into MariaDB to secure it, we'll need the current
password for the root user. If you've just installed MariaDB, and
haven't set the root password yet, you should just press enter here.

Enter current password for root (enter for none):
OK, successfully used password, moving on...

Setting the root password or using the unix_socket ensures that nobody
can log into the MariaDB root user without the proper authorisation.

You already have your root account protected, so you can safely answer 'n'.

Switch to unix_socket authentication [Y/n] n
 ... skipping.

You already have your root account protected, so you can safely answer 'n'.

Change the root password? [Y/n] y
New password:
Re-enter new password:
Password updated successfully!
Reloading privilege tables..
 ... Success!
```

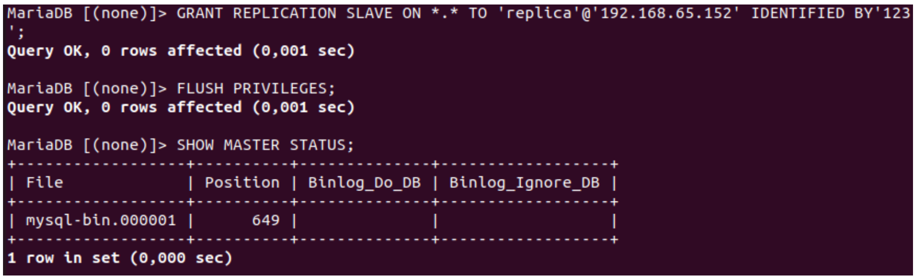
6. memberikan izin kepada pengguna "replica" untuk melakukan replikasi dari master MySQL ke slave MySQL

- Dari Master ke Slave



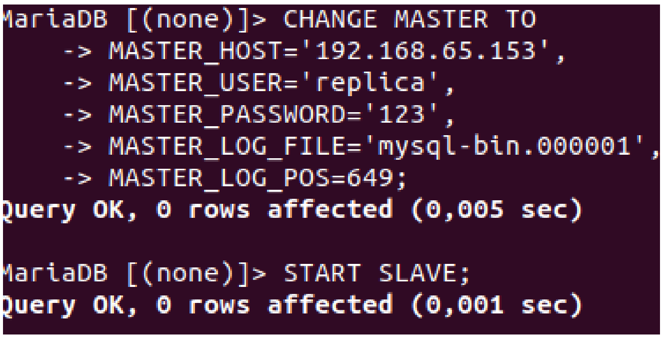
```
MariaDB [(none)]> GRANT REPLICATION SLAVE ON *.* TO 'replica'@'192.168.65.153' IDENTIFIED BY'123
';
Query OK, 0 rows affected (0,001 sec)

MariaDB [(none)]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0,001 sec)

MariaDB [(none)]> SHOW MASTER STATUS;
+------------------+----------+--------------+------------------+
| File             | Position | Binlog_Do_DB | Binlog_Ignore_DB |
+------------------+----------+--------------+------------------+
| mysql-bin.000001 |      970 |              |                  |
+------------------+----------+--------------+------------------+
1 row in set (0,000 sec)
```

- Dari Slave Ke Master



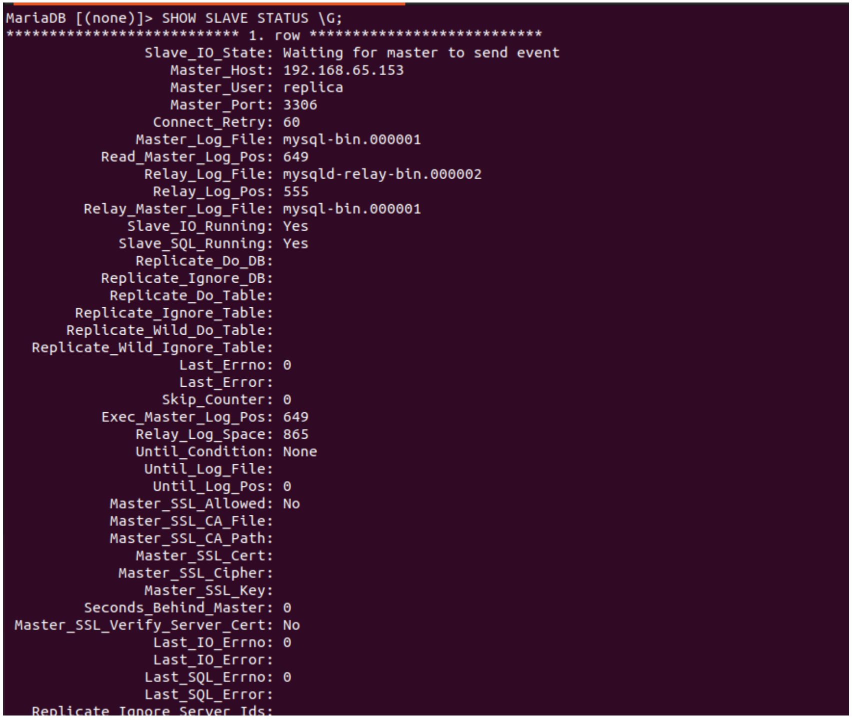
```
MariaDB [(none)]> GRANT REPLICATION SLAVE ON *.* TO 'replica'@'192.168.65.152' IDENTIFIED BY'123
';
Query OK, 0 rows affected (0,001 sec)

MariaDB [(none)]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0,001 sec)

MariaDB [(none)]> SHOW MASTER STATUS;
+------------------+----------+--------------+------------------+
| File             | Position | Binlog_Do_DB | Binlog_Ignore_DB |
+------------------+----------+--------------+------------------+
| mysql-bin.000001 |      649 |              |                  |
+------------------+----------+--------------+------------------+
1 row in set (0,000 sec)
```

7. Konfigurasi Replica Pada Master MariaDB ke Slave



```
MariaDB [(none)]> CHANGE MASTER TO
    -> MASTER_HOST='192.168.65.153',
    -> MASTER_USER='replica',
    -> MASTER_PASSWORD='123',
    -> MASTER_LOG_FILE='mysql-bin.000001',
    -> MASTER_LOG_POS=649;
Query OK, 0 rows affected (0,005 sec)

MariaDB [(none)]> START SLAVE;
Query OK, 0 rows affected (0,001 sec)
```

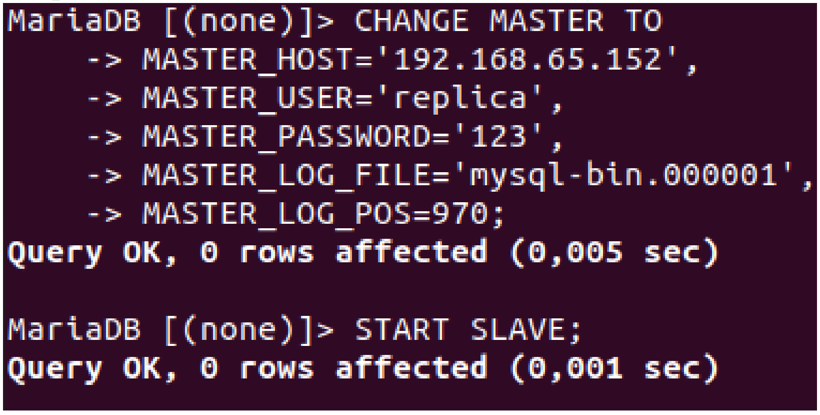
Status dari Konfigurasi Replica:



```
MariaDB [(none)]> SHOW SLAVE STATUS \G;
*************************** 1. row ***************************
                Slave_IO_State: Waiting for master to send event
                   Master_Host: 192.168.65.153
                   Master_User: replica
                   Master_Port: 3306
                 Connect_Retry: 60
               Master_Log_File: mysql-bin.000001
           Read_Master_Log_Pos: 649
                Relay_Log_File: mysqld-relay-bin.000002
                 Relay_Log_Pos: 555
         Relay_Master_Log_File: mysql-bin.000001
              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: 649
               Relay_Log_Space: 865
               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:
```

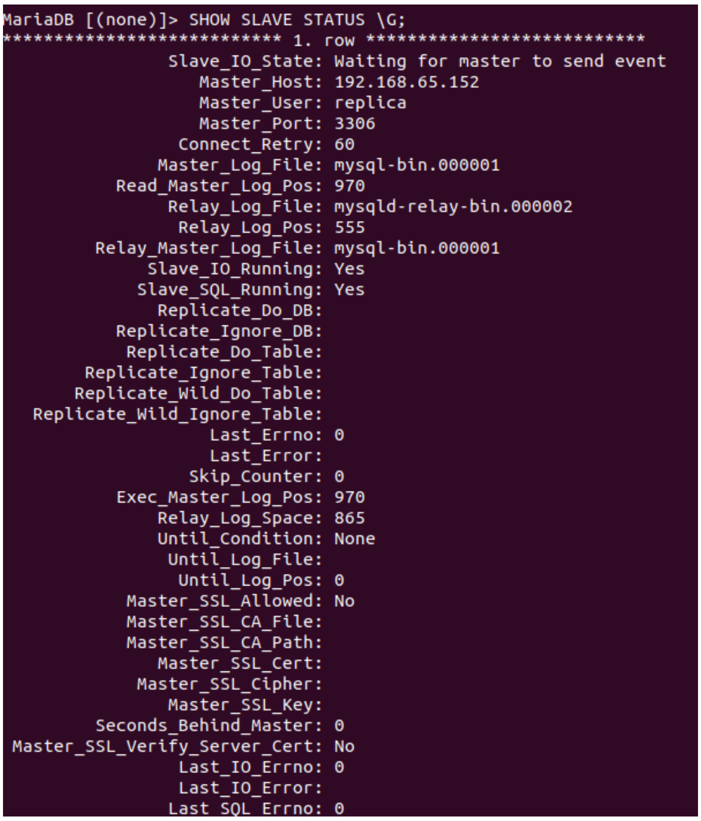
Dengan menggunakan perintah ini, server MariaDB akan dikonfigurasi untuk memulai replikasi dari server master yang berada di host **192.168.65.153**, menggunakan pengguna **replica** dengan kata sandi **123**, dimulai dari posisi tertentu di dalam file log bin **mysql-bin.000001**.

8. Konfigurasi Replica MariaDB Pada Slave ke Master



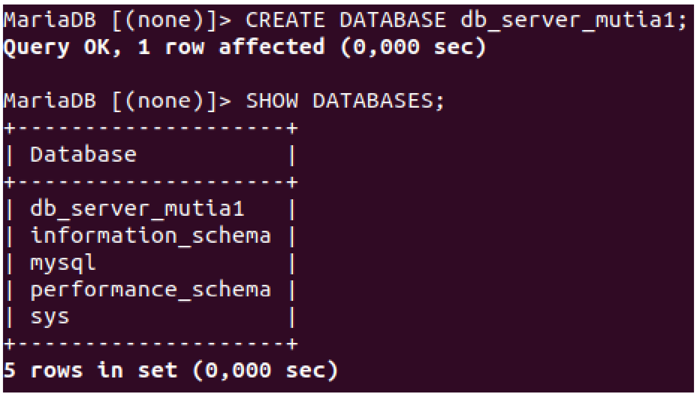
```
MariaDB [(none)]> CHANGE MASTER TO
    -> MASTER_HOST='192.168.65.152',
    -> MASTER_USER='replica',
    -> MASTER_PASSWORD='123',
    -> MASTER_LOG_FILE='mysql-bin.000001',
    -> MASTER_LOG_POS=970;
Query OK, 0 rows affected (0,005 sec)

MariaDB [(none)]> START SLAVE;
Query OK, 0 rows affected (0,001 sec)
```

Status dari Konfigurasi:



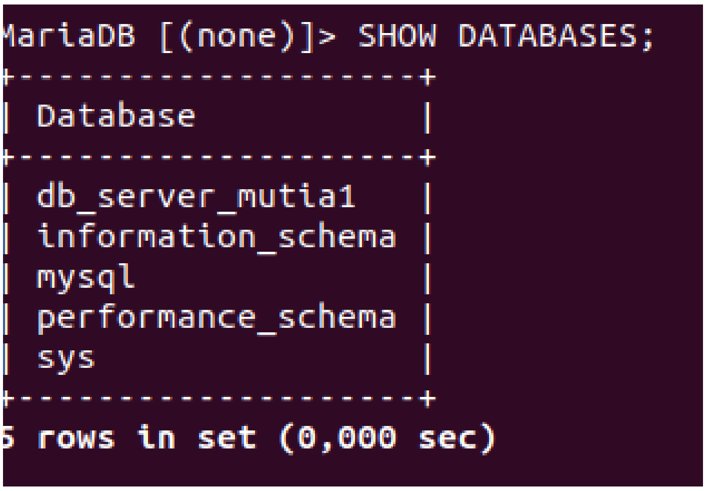
```
ariaDB [(none)]> SHOW SLAVE STATUS \G;
*************************** 1. row ***************************
                Slave_IO_State: Waiting for master to send event
                   Master_Host: 192.168.65.152
                   Master_User: replica
                   Master_Port: 3306
                 Connect_Retry: 60
               Master_Log_File: mysql-bin.000001
           Read_Master_Log_Pos: 970
                Relay_Log_File: mysqld-relay-bin.000002
                 Relay_Log_Pos: 555
         Relay_Master_Log_File: mysql-bin.000001
              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: 970
               Relay_Log_Space: 865
               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
```

9. Membuat Database Pada Master



```
MariaDB [(none)]> CREATE DATABASE db_server_mutia1;
Query OK, 1 row affected (0,000 sec)

MariaDB [(none)]> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| db_server_mutia1   |
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
5 rows in set (0,000 sec)
```

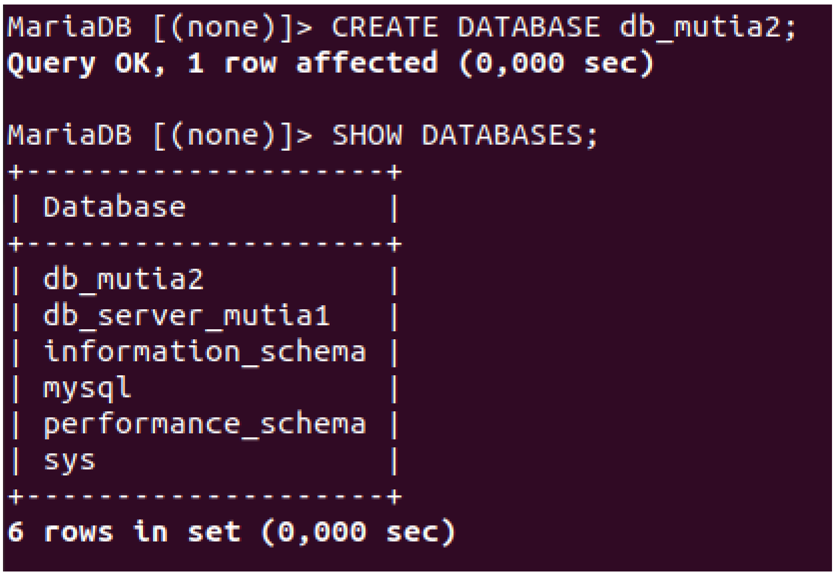
10. Bukti Hasil Replica dari Database Master Pada Slave



```
ariaDB [(none)]> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| db_server_mutia1   |
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
5 rows in set (0,000 sec)
```

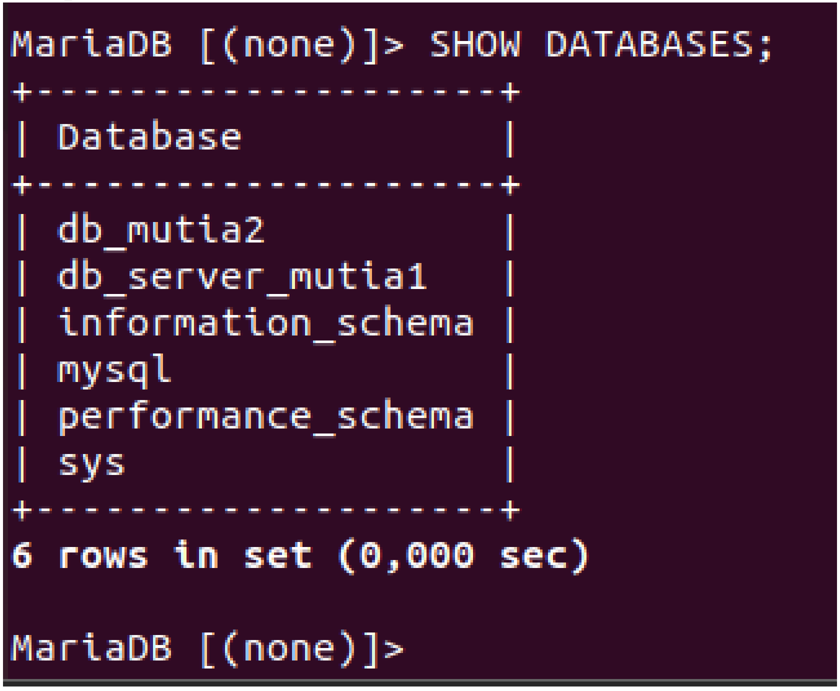
Dari hasil tersebut, konfigurasi replica pada Database MariaDB Master-Slave Berhasil.

11. Membuat Database Pada Slave



```
MariaDB [(none)]> CREATE DATABASE db_mutia2;
Query OK, 1 row affected (0,000 sec)

MariaDB [(none)]> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| db_mutia2          |
| db_server_mutia1   |
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
6 rows in set (0,000 sec)
```

12. Bukti Hasil Replica dari Database Slave Pada Master



```
MariaDB [(none)]> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| db_mutia2          |
| db_server_mutia1   |
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
6 rows in set (0,000 sec)

MariaDB [(none)]>
```

KESIMPULAN :

Konfigurasi database MariaDB pada mode master-slave adalah proses di mana dua server database MariaDB diatur sedemikian rupa sehingga satu server bertindak sebagai master yang menyediakan data yang akan direplikasi, sementara server lainnya bertindak sebagai slave yang menerima dan mengonfirmasi replikasi data dari master. Ini adalah pendekatan umum dalam sistem terdistribusi untuk meningkatkan ketersediaan data, meningkatkan performa, dan memungkinkan pemulihan bencana.

Langkah-langkah yang dijelaskan di kesimpulan adalah proses-proses kunci dalam konfigurasi master-slave MariaDB:

1. **Penetapan Alamat IP**: Setiap server diberikan alamat IP unik, satu untuk master dan satu untuk slave. Ini penting agar server dapat berkomunikasi satu sama lain dalam jaringan.

2. **Instalasi MariaDB**: MariaDB diinstal pada kedua server. Ini melibatkan mengunduh dan menginstal perangkat lunak MariaDB serta mengkonfigurasi pengaturan dasar.
3. **Konfigurasi MariaDB**: Pengaturan MariaDB diubah untuk menyesuaikan peran masing-masing server, termasuk konfigurasi sebagai master atau slave, serta pengaturan lainnya seperti pengaturan jaringan, izin akses, dan pengaturan replikasi.
4. **Akses ke Database**: Pengguna dapat mengakses database dari kedua server untuk melakukan operasi baca dan tulis sesuai dengan izin yang diberikan.
5. **Memberikan Izin untuk Replikasi**: Penggunaan perintah GRANT untuk memberikan izin kepada pengguna yang akan digunakan untuk replikasi data antara master dan slave.
6. **Konfigurasi Replikasi**: Pengaturan khusus diatur pada master dan slave untuk memungkinkan proses replikasi data. Ini melibatkan konfigurasi seperti menentukan server master dan slave, pengguna dan kata sandi untuk autentikasi, serta informasi log bin untuk menentukan titik awal replikasi.
7. **Verifikasi Hasil Replikasi**: Melakukan verifikasi untuk memastikan bahwa data dari master berhasil direplikasi ke slave dan sebaliknya, untuk memastikan bahwa proses replikasi berfungsi dengan benar.

Proses-proses ini membentuk dasar konfigurasi database MariaDB pada mode master-slave, yang memungkinkan sistem terdistribusi untuk memiliki redundansi data dan meningkatkan ketersediaan serta keandalan layanan.