

```
mysql> FLUSH PRIVILEGES;
```

4. If you disabled binary logging, enable it again as soon as you have created the user, by issuing the following statement:

```
mysql> SET SQL_LOG_BIN=1;
```

5. When you have created the replication user, you must supply the user credentials to the server for use with distributed recovery. You can do this by setting the user credentials as the credentials for the `group_replication_recovery` channel, using a `CHANGE REPLICATION SOURCE TO` statement (from MySQL 8.0.23) or `CHANGE MASTER TO` statement (before MySQL 8.0.23). Alternatively, from MySQL 8.0.21, you can specify the user credentials for distributed recovery on the `START GROUP_REPLICATION` statement.
 - User credentials set using `CHANGE REPLICATION SOURCE TO` | `CHANGE MASTER TO` are stored in plain text in the replication metadata repositories on the server. They are applied whenever Group Replication is started, including automatic starts if the `group_replication_start_on_boot` system variable is set to `ON`.
 - User credentials specified on `START GROUP_REPLICATION` are saved in memory only, and are removed by a `STOP GROUP_REPLICATION` statement or server shutdown. You must issue a `START GROUP_REPLICATION` statement to provide the credentials again, so you cannot start Group Replication automatically with these credentials. This method of specifying the user credentials helps to secure the Group Replication servers against unauthorized access.

For more information on the security implications of each method of providing the user credentials, see [Providing Replication User Credentials Securely](#). If you choose to provide the user credentials using a `CHANGE REPLICATION SOURCE TO` | `CHANGE MASTER TO` statement, issue the following statement on the server instance now, replacing `rpl_user` and `password` with the values used when creating the user:

```
mysql> CHANGE MASTER TO MASTER_USER='rpl_user', MASTER_PASSWORD='password' \\\n      FOR CHANNEL 'group_replication_recovery';
```

Or from MySQL 8.0.23:

```
mysql> CHANGE REPLICATION SOURCE TO SOURCE_USER='rpl_user', SOURCE_PASSWORD='password' \\\n      FOR CHANNEL 'group_replication_recovery';
```

18.2.1.4 Launching Group Replication

Once server `s1` has been configured and started, install the Group Replication plugin. If you used `plugin_load_add='group_replication.so'` in the option file then the Group Replication plugin is installed and you can proceed to the next step. In the event that you decide to install the plugin manually, connect to the server and issue the following:

```
INSTALL PLUGIN group_replication SONAME 'group_replication.so';
```



Important

The `mysql.session` user must exist before you can load Group Replication. `mysql.session` was added in MySQL version 8.0.2. If your data dictionary was initialized using an earlier version you must perform the MySQL upgrade procedure (see [Section 2.11, “Upgrading MySQL”](#)). If the upgrade is not run, Group Replication fails to start with the error message `There was an error when trying to access the server with user: mysql.session@localhost`. Make sure the user is present in the server and that `mysql_upgrade` was ran after a server update..