Section 17.1.3.4, “Setting Up Replication Using GTIDs”, which contains information about converting existing replication setups to use GTID-based replication.

Prior to MySQL 8.0.16, when the server is running with global transaction identifiers (GTIDs) enabled (`gtid_mode=ON`), do not enable binary logging by `mysql_upgrade` (the `--write-binlog` option). As of MySQL 8.0.16, the server performs the entire MySQL upgrade procedure, but disables binary logging during the upgrade, so there is no issue.

It is not recommended to load a dump file when GTIDs are enabled on the server (`gtid_mode=ON`), if your dump file includes system tables. `mysqldump` issues DML instructions for the system tables which use the non-transactional MyISAM storage engine, and this combination is not permitted when GTIDs are enabled. Also be aware that loading a dump file from a server with GTIDs enabled, into another server with GTIDs enabled, causes different transaction identifiers to be generated.

17.5.4 Troubleshooting Replication

If you have followed the instructions but your replication setup is not working, the first thing to do is *check the error log for messages*. Many users have lost time by not doing this soon enough after encountering problems.

If you cannot tell from the error log what the problem was, try the following techniques:

- Verify that the source has binary logging enabled by issuing a `SHOW MASTER STATUS` statement. Binary logging is enabled by default. If binary logging is enabled, `Position` is nonzero. If binary logging is not enabled, verify that you are not running the source with any settings that disable binary logging, such as the `--skip-log-bin` option.
- Verify that the `server_id` system variable was set at startup on both the source and replica and that the ID value is unique on each server.
- Verify that the replica is running. Use `SHOW REPLICA | SLAVE STATUS` to check whether the `Replica_IO_Running` and `Replica_SQL_Running` values are both `Yes`. If not, verify the options that were used when starting the replica server. For example, the `--skip-slave-start` command line option, or from MySQL 8.0.24, the `skip_slave_start` system variable, prevents the replication threads from starting until you issue a `START REPLICA | SLAVE` statement.
- If the replica is running, check whether it established a connection to the source. Use `SHOW PROCESSLIST`, find the I/O and SQL threads and check their `State` column to see what they display. See Section 17.2.3, “Replication Threads”. If the I/O thread state says `Connecting to master`, check the following:
  - Verify the privileges for the replication user on the source.
  - Check that the host name of the source is correct and that you are using the correct port to connect to the source. The port used for replication is the same as used for client network communication (the default is `3306`). For the host name, ensure that the name resolves to the correct IP address.
  - Check the configuration file to see whether the `skip_networking` system variable has been enabled on the source or replica to disable networking. If so, comment the setting or remove it.
  - If the source has a firewall or IP filtering configuration, ensure that the network port being used for MySQL is not being filtered.
  - Check that you can reach the source by using `ping` or `traceroute`/`tracert` to reach the host.
- If the replica was running previously but has stopped, the reason usually is that some statement that succeeded on the source failed on the replica. This should never happen if you have taken a