

Live Webinar March 11 | Deploy MariaDB Cloud in Your Microsoft Azure Account. Retain Control. [Register Now](#)



Home MariaDB Platform Server MaxScale Analytics **Galera Cluster** Cor

GALERA MANAGEMENT > INSTALLATION & DEPLOYMENT

Ask

Getting Started with MariaDB Galera Cluster

Complete MariaDB installation guide. Complete setup instructions for Linux, Windows, and macOS with configuration and verification for production use.

The most recent release of [MariaDB 10.11](#) is: [MariaDB 10.11.11](#) Stable (GA) [Download Now](#), [Alternate download from mariadb.org](#)

The current [versions](#) of the Galera wsrep provider library are 26.4.21 for [Galera 4](#). *For convenience, packages containing these libraries are included in the MariaDB [YUM and APT repositories](#).*

Currently, MariaDB Galera Cluster only supports the [InnoDB](#) storage engine (although there is experimental support for [MyISAM](#) and, from [MariaDB 10.6](#), [Aria](#)).

Galera Cluster Support in MariaDB Server

MariaDB Galera Cluster is powered by:

- MariaDB Server.
- The [MySQL-wsrep](#) patch for MySQL Server and MariaDB Server. The patch currently supports only Unix-like operating systems.

- The [Galera wsrep provider library](#) ↗.

The [MySQL-wsrep](#) ↗ patch has been merged into MariaDB Server. This means that the functionality of MariaDB Galera Cluster can be obtained by installing the standard MariaDB Server packages and the [Galera wsrep provider library](#) ↗ package. The following [Galera](#) version corresponds to each MariaDB Server version:

- MariaDB Galera Cluster uses [Galera](#) 4. This means that the [MySQL-wsrep](#) ↗ patch is version 26 and the [Galera wsrep provider library](#) ↗ is version 4.

See [Deciphering Galera Version Numbers](#) ↗ for more information about how to interpret these version numbers.

See [What is MariaDB Galera Cluster: Galera Versions](#) for more information about which specific [Galera](#) version is included in each release of MariaDB Server.

In supported builds, Galera Cluster functionality can be enabled by setting some configuration options that are mentioned below. Galera Cluster functionality is not enabled in a standard MariaDB Server installation unless explicitly enabled with these configuration options.

Prerequisites

Swap Size Requirements

During normal operation, a MariaDB Galera node consumes no more memory than a regular MariaDB server. Additional memory is consumed for the certification index and uncommitted write sets, but normally, this should not be noticeable in a typical application. There is one exception, though:

Writeset caching during state transfer

When a node is receiving a state transfer, it cannot process and apply incoming writesets because it has no state to apply them to yet. Depending on a state transfer mechanism (e.g. [mysqldump](#)) the node that sends the state transfer may not be able to apply writesets as well. Thus, they need to cache those writesets for a catch-up phase. Currently the writesets are cached in memory and, if the system runs out of memory either the state transfer will fail or the cluster will block waiting for the state transfer to end.

To control memory usage for writeset caching, check the [Galera parameters](#):

```
gcs.recv_q_hard_limit , gcs.recv_q_soft_limit , and gcs.max_throttle .
```

Limitations

Before using MariaDB Galera Cluster, we would recommend reading through the [known limitations](#), so you can be sure that it is appropriate for your application.

Installing MariaDB Galera Cluster

To use MariaDB Galera Cluster, there are two primary packages that you need to install:

1. A MariaDB Server version that supports Galera Cluster.
2. The Galera wsrep provider library.

As mentioned in the previous section, Galera Cluster support is actually included in the standard MariaDB Server packages. That means that installing MariaDB Galera Cluster package is the same as installing standard MariaDB Server package in those versions. However, you will also have to install an additional package to obtain the Galera wsrep provider library.

Some [SST](#) methods may also require additional packages to be installed. The [mariadb-backup](#) SST method is generally the best option for large clusters that expect a lot of loads.

Installing MariaDB Galera Cluster with a Package Manager

MariaDB Galera Cluster can be installed via a package manager on Linux. In order to do so, your system needs to be configured to install from one of the MariaDB repositories.

You can configure your package manager to install it from MariaDB Corporation's MariaDB Package Repository by using the [MariaDB Package Repository setup script](#).

You can also configure your package manager to install it from MariaDB Foundation's MariaDB Repository by using the [MariaDB Repository Configuration Tool](#).

Installing MariaDB Galera Cluster with yum/dnf

On RHEL, CentOS, Fedora, and other similar Linux distributions, it is highly recommended to install the relevant [RPM packages](#) from MariaDB's repository using [yum](#) or [dnf](#). Starting with RHEL 8 and Fedora 22, `yum` has been replaced by `dnf`, which is the next major version of `yum`. However, `yum` commands still work on many systems that use `dnf`.

To install MariaDB Galera Cluster with `yum` or `dnf`, follow the instructions at [Installing MariaDB Galera Cluster with yum ↗](#).

Installing MariaDB Galera Cluster with apt-get

On Debian, Ubuntu, and other similar Linux distributions, it is highly recommended to install the relevant [DEB packages ↗](#) from MariaDB's repository using [apt-get ↗](#).

To install MariaDB Galera Cluster with `apt-get`, follow the instructions at [Installing MariaDB Galera Cluster with apt-get ↗](#).

Installing MariaDB Galera Cluster with zypper

On SLES, OpenSUSE, and other similar Linux distributions, it is highly recommended to install the relevant [RPM packages](#) from MariaDB's repository using [zypper](#) ↗.

To install MariaDB Galera Cluster with `zypper`, follow the instructions at [Installing MariaDB Galera Cluster with ZYpp](#) ↗.

Installing MariaDB Galera Cluster with a Binary Tarball

To install MariaDB Galera Cluster with a binary tarball, follow the instructions at [Installing MariaDB Binary Tarballs](#).

To make the location of the `libgalera_smm.so` library in binary tarballs more similar to its location in other packages, the library is now found at `lib/galera/libgalera_smm.so` in the binary tarballs, and there is a symbolic link in the `lib` directory that points to it.

Installing MariaDB Galera Cluster from Source

To install MariaDB Galera Cluster by compiling it from source, you will have to compile both MariaDB Server and the Galera wsrep provider library. For some information on how to do this, see the pages at [Installing Galera From Source](#). The pages at [Compiling MariaDB From Source](#) and Galera Cluster Documentation: Building Galera Cluster for MySQL may also be helpful.

Configuring MariaDB Galera Cluster

A number of options need to be set in order for Galera Cluster to work when using MariaDB. See [Configuring MariaDB Galera Cluster](#) for more information.

Bootstrapping a New Cluster

To first node of a new cluster needs to be bootstrapped by starting [mariadb](#) on that node with the option [--wsrep-new-cluster](#) option. This option tells the node that there is no existing cluster to connect to. The node will create a new UUID to identify the new cluster.

Do not use the [--wsrep-new-cluster](#) option when connecting to an existing cluster. Restarting the node with this option set will cause the node to create new UUID to identify the cluster again, and the node won't reconnect to the old cluster. See the next section about how to reconnect to an existing cluster.

For example, if you were manually starting [mariadb](#) on a node, then you could bootstrap it by executing the following:

```
$ mariadb --wsrep-new-cluster
```

However, keep in mind that most users are not going to be starting [mariadb](#) manually. Instead, most users will use a [service manager](#) to start [mariadb](#). See the following sections on how to bootstrap a node with the most common service managers.

Systemd and Bootstrapping

On operating systems that use [systemd](#), a node can be bootstrapped in the following way:

```
$ galera_new_cluster
```

This wrapper uses [systemd](#) to run [mariadb](#) with the [--wsrep-new-cluster](#) option.

If you are using the [systemd](#) service that supports the [systemd service's method for interacting with multiple MariaDB Server processes](#), then you can bootstrap a specific instance by specifying the instance name as a suffix. For example:

```
$ galera_new_cluster mariadb@node1
```

Systemd support and the `galera_new_cluster` script were added.

SysVinit and Bootstrapping

On operating systems that use [sysVinit](#), a node can be bootstrapped in the following way:

```
$ service mysql bootstrap
```

This runs [mariadb](#) with the `--wsrep-new-cluster` option.

Adding Another Node to a Cluster

Once you have a cluster running and you want to add/reconnect another node to it, you must supply an address of one or more of the existing cluster members in the [wsrep_cluster_address](#) option. For example, if the first node of the cluster has the address `192.168.0.1`, then you could add a second node to the cluster by setting the following option in a server [option group](#) in an [option file](#):

```
[mariadb]
...
wsrep_cluster_address=gcomm://192.168.0.1 # DNS names work as
well, IP is preferred for performance
```

The new node only needs to connect to one of the existing cluster nodes. Once it connects to one of the existing cluster nodes, it will be able to see all of the nodes in the cluster. However, it is generally better to list all nodes of the cluster in [wsrep_cluster_address](#), so that any node can join a cluster by connecting to any of the other cluster nodes, even if one or more of the cluster nodes are down. It is even OK to list a node's own IP address in [wsrep_cluster_address](#), since Galera Cluster is smart enough to ignore it.

Once all members agree on the membership, the cluster's state will be exchanged. If the new node's state is different from that of the cluster, then it will request an IST or [SST](#) to make itself consistent with the other nodes.

Restarting the Cluster

If you shut down all nodes at the same time, then you have effectively terminated the cluster. Of course, the cluster's data still exists, but the running cluster no longer exists. When this happens, you'll need to bootstrap the cluster again.

If the cluster is not bootstrapped and [mariadb](#) on the first node is just started normally, then the node will try to connect to at least one of the nodes listed in the [wsrep_cluster_address](#) option. If no nodes are currently running, then this will fail. Bootstrapping the first node solves this problem.

Determining the Most Advanced Node

In some cases Galera will refuse to bootstrap a node if it detects that it might not be the most advanced node in the cluster. Galera makes this determination if the node was not the last one in the cluster to be shut down or if the node crashed. In those cases, manual intervention is needed.

If you know for sure which node is the most advanced you can edit the `grastate.dat` file in the [datadir](#). You can set `safe_to_bootstrap=1` on the most advanced node.

You can determine which node is the most advanced by checking `grastate.dat` on each node and looking for the node with the highest `seqno`. If the node crashed and `seqno=-1`, then you can find the most advanced node by recovering the `seqno` on each node with the [wsrep_recover](#) option. For example:

```
$ mariadb --wsrep_recover
```

Systemd and Galera Recovery

On operating systems that use [systemd](#), the position of a node can be recovered by running the `galera_recovery` script. For example:


```
$ galera_recovery
```

If you are using the [systemd](#) service that supports the [systemd service's method for interacting with multiple MariaDB Server processes](#), then you can recover the position of a specific instance by specifying the instance name as a suffix. For example:

```
$ galera_recovery mariadb@node1
```

The `galera_recovery` script recovers the position of a node by running [mariabdb](#) with the [wsrep_recover](#) option.

When the `galera_recovery` script runs [mariabdb](#), it does not write to the [error log](#). Instead, it redirects [mariabdb](#) log output to a file named with the format `/tmp/wsrep_recovery.XXXXXX`, where `XXXXXX` is replaced with random characters.

When Galera is enabled, MariaDB's [systemd](#) service automatically runs the `galera_recovery` script prior to starting MariaDB, so that MariaDB starts with the proper Galera position.

Support for [systemd](#) and the `galera_recovery` script were added.

State Snapshot Transfers (SSTs)

In a State Snapshot Transfer (SST), the cluster provisions nodes by transferring a full data copy from one node to another. When a new node joins the cluster, the new node initiates a State Snapshot Transfer to synchronize its data with a node that is already part of the cluster.

See [Introduction to State Snapshot Transfers \(SSTs\)](#) for more information.

Incremental State Transfers (ISTs)

In an Incremental State Transfer (SST), the cluster provisions nodes by transferring a node's missing writesets from one node to another. When a new node joins the cluster, the new node initiates a Incremental State Transfer to synchronize its data with a node that is already part of the cluster.

If a node has only been out of a cluster for a little while, then an IST is generally faster than an SST.

Data at Rest Encryption

MariaDB Galera Cluster supports [Data at Rest Encryption](#). See [SSTs and Data at Rest Encryption](#) for some disclaimers on how SSTs are affected when encryption is configured.

Some data still cannot be encrypted:

- The disk-based [Galera gcache](#) ↗ is not encrypted ([MDEV-8072](#) ↗).

Monitoring

Status Variables

[Galera Cluster's status variables](#) can be queried with the standard [SHOW STATUS](#) ↗ command. For example:

```
SHOW GLOBAL STATUS LIKE 'wsrep_%';
```

Cluster Change Notifications

The cluster nodes can be configured to invoke a command when cluster membership or node status changes. This mechanism can also be used to communicate the event to some external monitoring agent. This is configured by setting [wsrep_notify_cmd](#). See [Galera Cluster documentation: Notification Command](#) ↗ for more information.

See Also

- [What is MariaDB Galera Cluster?](#)
- [About Galera Replication](#)
- [Galera Use Cases](#)
- [Galera Cluster documentation: Notification Command](#)
- [Introducing the “Safe-To-Bootstrap” feature in Galera Cluster](#) ↗
- [Github - galera](#) ↗
- [Github - mysql-wsrep](#) ↗

Footnotes

This page is licensed: CC BY-SA / Gnu FDL

Subscribe to our newsletter!

Previous
Installation & Deployment

Next
Installing MariaDB Galera on IBM Cloud

Last updated 27 days ago

Was this helpful?



Products	Support	Resources	Company
Enterprise Platform	Customer Login	MariaDB Blog	About MariaDB
Community Server	Technical Support	Webinars	Newsroom
Download MariaDB	Remote DBA	Customer Stories	Leadership
Pricing	Professional Services	MariaDB Events	MariaDB Careers
		Documentation	Legal
		Developer Hub	Privacy Policy

© 2026 MariaDB. All rights reserved.