CONFIGURATION TIPS

This chapter contains some advanced configuration tips.

WAN REPLICATION ¶

When running the cluster over WAN, you may frequently experience transient network connectivity failures. To prevent this from partitioning the cluster, you may want to increase the keepalive timeouts.

The following parameters can tolerate 30 second connectivity outages.

```
wsrep_provider_options = "evs.keepalive_period = PT3S;
      evs.suspect_timeout = PT30S;
      evs.inactive_timeout = PT1M;
      evs.install_timeout = PT1M"
```

Note: All `wsrep_provider_options` settings need to be specified on a single line. In case of multiple instances of `wsrep_provider_options`, only the last one is used.

In configuring these parameters, consider the following:

- You want evs.suspect_timeout parameter set as high as possible to help avoid partitions. Given that partitions cause state transfers, which can effect performance.
- You must set the evs.inactive_timeout parameter to a value higher than evs.suspect_timeout.
- You must set the evs.install_timeout parameter to a value higher than the evs.inactive_timeout.

Dealing with WAN Latency

When using Galera Cluster over a WAN, bear in mind that WAN links can have exceptionally high latency. You can correct for this by taking Round-Trip Time (RTT) measurements between cluster nodes and adjust all temporal parameters.

To take RTT measurements, use `ping` on each cluster node to ping the others. For example, if you were to log in to the node at `192.168.1.1`:

```
$ ping -c 3 192.168.1.2
  PING 192.168.1.2 (192.168.1.2) 58(84) bytes of data.
  64 bytes from 192.168.1.2: icmp_seq=1 ttl=64 time=0.736 ms
  64 bytes from 192.168.1.2: icmp_seq=2 ttl=64 time=0.878 ms
  64 bytes from 192.168.1.2: icmp_seq=3 ttl=64 time=12.7 ms

  --- 192.168.1.2 ---
  3 packets transmitted, 3 received, 0% packet loss, time 2002ms
  rtt min/avg/max/mdev = 0.736/4.788/12.752/5.631 ms
```

Take RTT measurements on each node in your cluster and note the highest value among them.

Parameters that relate to periods and timeouts, such as evs.join_retrans_period. They must all use values that exceed the highest RTT measurement in your cluster.

```
wsrep_provider_options="evs.join_retrans_period=PT0.5S"
```

This allows the cluster to compensate for the latency issues of the WAN links between your cluster nodes.

MULTI-MASTER SETUP

A master is a node that can simultaneously process writes from clients.

The more masters you have in the cluster the higher the probability of certification conflicts. This can lead to undesirable rollbacks and performance degradation.

If you find you experience frequent certification conflicts, consider reducing the number of nodes your cluster uses as masters.

SINGLE MASTER SETUP

In the event that your cluster uses only one node as a master, there are certain requirements, such as the slave queue size, that can be relaxed.

To relax flow control, use the settings below:

```
wsrep_provider_options = "gcs.fc_limit = 256;
    gcs.fc_factor = 0.99;
    gcs.fc_master_slave = YES"
```

By reducing the rate of flow control events, these settings may improve replication performance.

Note: You can also use this setting as suboptimal in a multi-master setup.

USING GALERA CLUSTER WITH SELINUX

When you first enable Galera Cluster on a node that runs SELinux, SELinux prohibits all cluster activities. In order to enable replication on the node, you need a policy so that SELinux can recognize cluster activities as legitimate.

To create a policy for Galera Cluster, set SELinux to run in permissive mode. Permissive mode does not block cluster activity, but it does log the actions as warnings. By collecting these warnings, you can iteratively create a policy for Galera Cluster.

Once SELinux no longer registers warnings from Galera Cluster, you can switch it back into enforcing mode. SELinux then uses the new policy to allow the cluster access to the various ports and files it needs.

Note: Almost all Linux distributions ship with a MySQL policy for SELinux. You can use this policy as a starting point for Galera Cluster and extend it, using the above procedure.