Important variables for configuring galera with Percona XtraDB Cluster 5.6

*Submitted by **Walter Heck** on **December 21, 2013***

#MYSQL #GALERA #HIGH AVAILABILITY #CONFIGURATION
#PERCONA XTRADB CLUSTER #XTRABACKUP

In the past few days I have been working on a puppet module for galera. It's not done yet, but a work in progress can be found here. One of the benefits of developing a puppet module is that you need to go through the config files thoroughly to determine which settings do what exactly. While doing that, I thought I'd post a blog outlining the most important ones.

I believe that inter-node communication should preferrably take place on an unmetered private network interface if possible. For this article, we assume each machine has a private and a public interface, like so (Debian-style):

```
root@node1:~# cat /etc/network/interfaces
[..snip..]
# Public interface
auto eth0
iface eth0 inet static
address 5.1.2.3
netmask 255.255.255.248
gateway 5.1.2.254

# Private interface
auto eth1
iface eth1 inet static
address 192.168.100.3
netmask 255.255.255.0

root@node2:~# cat /etc/network/interfaces
[..snip..]
# Public interface
auto eth0
iface eth0 inet static
address 5.1.2.4
netmask 255.255.255.248
gateway 5.1.2.254

# Private interface
auto eth1
iface eth1 inet static
```

```
address 192.168.100.4
netmask 255.255.255.0
```

MySQL options

I won't write too much about the non-galera mysql options you need to configure, just go with something like this:

```
##
## mysqld options _MANDATORY_ for correct opration of the cluster
##
[mysqld]
# (This must be substituted by wsrep_format)
binlog_format=ROW
# Currently only InnoDB storage engine is supported
default-storage-engine=innodb
# to avoid issues with 'bulk mode inserts' using autoinc
innodb_autoinc_lock_mode=2
# This is a must for paralell applying
innodb_locks_unsafe_for_binlog=1
# Query Cache is not supported with wsrep
query_cache_size=0
query_cache_type=0
# Override bind-address
# In some systems bind-address defaults to 127.0.0.1, and with mysqldump SST
# it will have (most likely) disastrous consequences on donor node
bind-address=0.0.0.0
```

The only special variable is bind-address, which you can either set to 0.0.0.0 to make the MySQL server listen on all interfaces, or you can set it to the IP address of eth0 or eth1, whichever your setup makes logical.

wsrep options

I'll list out the most important galera options and tell you what values they should have.

wsrep_provider

This is a value that points to the location of the galera library. For Percona Xtradb Cluster 5.6 on Debian Wheezy (seems to be the same for CentOS 6 at least):

```
# Full path to wsrep provider library or 'none'
wsrep_provider=/usr/lib/libgalera_smm.so
```

wsrep_provider_options

This parameter allows you to fine tune galera. For a full list of possible values see the codership wiki. The only value we need to set if we have multiple network interfaces is base_host, so we can specify it like this:

```
# Provider specific configuration options
wsrep_provider_options="base_host=192.168.100.131"
```

wsrep_cluster_name

This parameter is not really necessary, but especially if you have multiple galera clusters it will be a good idea to name them something logical here. It's recommended to set it though, because if a node connects to a cluster with a different name, the connection will fail.

```
# Logical cluster name. Should be the same for all nodes.
wsrep_cluster_name="galera"
```

wsrep_cluster_address

This is probably one of the most confusing variables. This is because it's supported and/or recommended values have changed over the course of Galera 2 to Galera 3. Add to that that it needs a special empty value in order to bootstrap a cluster and the confusion is complete. What this variable represents is the location of the rest of the galera cluster. This is used by Galera to connect to the cluster and get a copy of the data and set up initial communications.

Bootstrapping a cluster

This is the term used for bringing up the first node of a brand new cluster. For this node, there will obviously not be another node to connect to, so you leave the value empty, like this: wsrep_cluster_address="gcomm://". You can feed that value on the command line to /etc/init.d/mysql if you wish (it'll override the config file value), or you can specify it empty in the config file (not recommended, if you restart MySQL it'll start a new galera cluster).

Pre-Galera 2.2

Before Galera 2.2 multiple addresses in the wsrep_cluster_address variable was not supported, so you would have to choose one of the other nodes and hope that that one was actually up and running when starting MySQL. Then, for a short while there was a wsrep_urls parameter that you could set in the [mysqld_safe] section of my.cnf, but that is no longer recommended (and even deprecated) since Percona XtraDB Cluster 5.5.28 (see the changelog here).

Recommended

Now, it is mostly recommended to list out the addresses of all nodes in the cluster you want to connect to in comma-separated format. That way, the node will connect to each address until it finds a node that is alive. It is also possible to add more configuration parameters here, see the galera wiki for more info.

```
# Group communication system handle
wsrep_cluster_address="gcomm://192.168.100.129"
```

wsrep_node_name

Like wsrep_cluster_name, it's a logical value to be able to recognise your node in the cluster. By default it's the node's hostname.

```
# Human-readable node name (non-unique). Hostname by default.
wsrep_node_name=hancock
```

wsrep_node_address

This parameter is used for identifying the network interface over which communication with the cluster happens. It defaults to the ip of the first interface in the machine, so especially in our case with a private interface it needs to be manually set.

```
# Base replication <address|hostname>[:port] of the node.
# The values supplied will be used as defaults for state transfer receiving,
# listening ports and so on. Default: address of the first network interface.
wsrep_node_address=192.168.100.131
```

wsrep_slave_threads

The number of threads to use for applying slave write sets. By default it's set to 1, but the recommended setting is at least twice the number of CPU cores. If you experience trouble with higher settings, just lower it and see if it improves the situation.

```
# How many threads will process writesets from other nodes
wsrep_slave_threads=1
```

wsrep_notify_cmd

This parameter allows you to specify the name of a script that will be called when this node's status changes. Use it to alert you by mail, or to update an overview of the status of each node in a external location for instance.
An example in the galera wiki shows what's possible. Be aware: this bug currently seems to be present in the script.

```
# Command to call when node status or cluster membership changes.
# Will be passed all or some of the following options:
# --status - new status of this node
# --uuid - UUID of the cluster
# --primary - whether the component is primary or not ("yes"/"no")
# --members - comma-separated list of members
# --index - index of this node in the list
wsrep_notify_cmd=/usr/local/bin/wsrep_notify.sh
```

State Transfer [SST] related options in the [mysqld] section

Galera wil automatically sync a new node's databases with the ones from the cluster when it joins. If possible it will do an Incremental State Transfer [IST], otherwise it will do a

Full State Transfer. There are a number of different parameters that can control what method it uses.

wsrep_sst_method

This is the most important parameter, it determines which method to use. Valid values are 'rsync', 'mysqldump', 'xtrabackup' and 'xtrabackup-v2'. I won't go in-depth in which method to choose, if you're not sure just choose 'xtrabackup-v2'.

```
# State Snapshot Transfer method
wsrep_sst_method=xtrabackup-v2
```

wsrep_sst_receive_address

This is the address that the donor should send the backup to. In our case, the ip address of our private interface. If you're behind NAT for instance, set it to the outside address.

```
# Address which donor should send State Snapshot to.
# Should be the address of THIS node. DON'T SET IT TO DONOR ADDRESS!!!
# (SST method dependent. Defaults to the first IP of the first interface)
wsrep_sst_receive_address=192.168.100.131
```

wsrep_sst_auth

The authentication string to use for the state transfer. For 'xtrabackup-v2' it just uses the format <user>:<password>. The grants needed for xtrabackup are minimum this:

```
CREATE USER 'sst'@'localhost' IDENTIFIED BY 'somepass';
GRANT RELOAD, LOCK TABLES, REPLICATION CLIENT ON *.* TO 'sst'@'localhost';

# SST authentication string. This will be used to send SST to joining nodes.
# Depends on SST method. For mysqldump method it is root:
wsrep_sst_auth=sst_xtrabackup:somepassword
```

wsrep_sst_donor

This setting is optional. You can set it to a comma separated list of the wsrep_node_name's you wish to use as donors. Galera will connect to each one from the first to the last, and take the first available node as a donor. If it can't find any of the nodes in the list available, it tries to autoselect other nodes from the cluster.

```
# Desired SST donor name.
wsrep_sst_donor=node1,node2,node3
```

State Transfer [SST] related options in the [sst] section

Besides the [mysqld] section of the mysql config file, you can also specify some extra options in the [sst] section. Here's some useful ones, provided you chose 'xtrabackup-v2' as the wsrep_sst_method. Check the documentation here for more info.

streamfmt

The streamfmt option specifies which method to use for streaming the backup from xtrabackup. xbstream is recommended over tar (more info here), but it's only available since xtrabackup 2.1.

```
# Xbstream is highly recommended. Refer to http://www.percona.com/doc/percona-xtradb-cluster/5.6/manual/xtrabackup…
# for details and caveats of using tar v/s xbstream for SST.
streamfmt=xbstream
```

progress

If you are working with large databases, it can be a bit hard to see how far along the SST actually is. On the receiving node, MySQL will not start until the full SST is done. The progress parameter can be set to make it a bit more clear how far along the SST is.

```
# If equal to:
# 1 it writes to mysql stderr
# path/to/file writes to that file. If this is a fifo, it needs to exist and
be open on reader end before itself,
# otherwise wsrep_sst_xtrabackup will block indefinitely.
progress=1
```

time

This option is used to show a bit more statistics about your SST so you can see which stage takes how much time.

```
# Enabling it instruments key stages of backup/restore in SST.
time=1
```