TP 1 Prise en main

- Démarrage de MySQL

Démarrage de MySQL

Le démarrage se fait via simple.sh

Faire un show engines:

```
SHOW ENGINES \G;
*************************** 1. row ***************************
      Engine: FEDERATED
     Support: NO
     Comment: Federated MySQL storage engine
Transactions: NULL
          XA: NULL
  Savepoints: NULL
*************************** 2. row ***************************
      Engine: MEMORY
     Support: YES
     Comment: Hash based, stored in memory, useful for temporary tables
Transactions: NO
          XA: NO
  Savepoints: NO
*************************** 3. row ***************************
      Engine: InnoDB
     Support: DEFAULT
     Comment: Supports transactions, row-level locking, and foreign keys
Transactions: YES
          XA: YES
  Savepoints: YES
*************************** 4. row ***************************
      Engine: PERFORMANCE_SCHEMA
     Support: YES
     Comment: Performance Schema
Transactions: NO
          XA: NO
  Savepoints: NO
*************************** 5. row ***************************
      Engine: MyISAM
     Support: YES
```

```
     Comment: MyISAM storage engine
Transactions: NO
          XA: NO
  Savepoints: NO
*************************** 6. row ***************************
      Engine: MRG_MYISAM
     Support: YES
     Comment: Collection of identical MyISAM tables
Transactions: NO
          XA: NO
  Savepoints: NO
*************************** 7. row ***************************
      Engine: BLACKHOLE
     Support: YES
     Comment: /dev/null storage engine (anything you write to it disappears)
Transactions: NO
          XA: NO
  Savepoints: NO
*************************** 8. row ***************************
      Engine: CSV
     Support: YES
     Comment: CSV storage engine
Transactions: NO
          XA: NO
  Savepoints: NO
*************************** 9. row ***************************
      Engine: ARCHIVE
     Support: YES
     Comment: Archive storage engine
Transactions: NO
          XA: NO
  Savepoints: NO
9 rows in set (0.00 sec)
```

Faire un show tables afin de voir les détails d'une tables:

```
mysql> SHOW table status \G;
*************************** 1. row ***************************
           Name: titi
         Engine: InnoDB
```

```
        Version: 10
     Row_format: Dynamic
           Rows: 0
 Avg_row_length: 0
    Data_length: 16384
Max_data_length: 0
   Index_length: 0
      Data_free: 0
 Auto_increment: NULL
    Create_time: 2020-06-16 21:03:06
    Update_time: NULL
     Check_time: NULL
      Collation: utf8mb4_0900_ai_ci
       Checksum: NULL
 Create_options:
        Comment:
1 row in set (0.00 sec)
```

Faire un

```
CREATE TABLE t2 (i INT NOT NULL) ENGINE = CSV;
Query OK, 0 rows affected (0.05 sec)
```

Et vérifier que le fichier est bien la:

```
/home/pilou/Formation/simpleinit/data/t1
```

Rajoutons une ligne et remarquons qu'il faut bien flusher les tables si on veut que cela fasse qq chose:

```
mysql> select * from t2;
+---+
| i |
+---+
| 4 |
+---+
1 row in set (0.00 sec)

mysql> flush tables;
```

```
Query OK, 0 rows affected (0.06 sec)

mysql> select * from t2;
+---+
| i |
+---+
| 4 |
| 5 |
+---+
2 rows in set (0.01 sec)
```

Le moteur de stockage peut être changer dynamiquement par défaut:

```
SET default_storage_engine=MYISAM;
```

On vérifie que MYISAM n'est pas transactionnel:

```
mysql> begin;
Query OK, 0 rows affected (0.00 sec)

mysql> insert into testisam values(3);
Query OK, 1 row affected (0.01 sec)

mysql> rollback;
Query OK, 0 rows affected, 1 warning (0.00 sec)

mysql> select * from testisam;
+------+
| i    |
+------+
|    3 |
+------+
1 row in set (0.00 sec)
```

Il est possible de modifier le storage engine de la table

```
mysql> ALTER TABLE testisam ENGINE = InnoDB;
Query OK, 1 row affected (0.07 sec)
Records: 1  Duplicates: 0  Warnings: 0
```

mysql tuner

```
[--] Status: +ARCHIVE +BLACKHOLE +CSV -FEDERATED +InnoDB +MEMORY +MRG_MYISAM +MyISAM
+PERFORMANCE_SCHEMA
>default-storage-engine=InnoDB
>disabled_storage_engines="MyISAM,BLACKHOLE,FEDERATED,ARCHIVE,CSV"

[--] Data in CSV tables: 0B (Tables: 1)
[--] Data in InnoDB tables: 49.0K (Tables: 4)
[OK] Total fragmented tables: 0

-------- Analysis Performance Metrics
----------------------------------------------------------------
[--] innodb_stats_on_metadata: OFF
[OK] No stat updates during querying INFORMATION_SCHEMA.

-------- Security Recommendations
----------------------------------------------------------------
[--] Skipped due to unsupported feature for MySQL 8

-------- CVE Security Recommendations
----------------------------------------------------------------
[OK] NO SECURITY CVE FOUND FOR YOUR VERSION

-------- Performance Metrics
----------------------------------------------------------------------
[--] Up for: 54s (22 q [0.407 qps], 15 conn, TX: 56K, RX: 1K)
[--] Reads / Writes: 100% / 0%
[--] Binary logging is enabled (GTID MODE: OFF)
[--] Physical Memory     : 2.9G
[--] Max MySQL memory    : 9.8G
[--] Other process memory: 0B
[--] Total buffers: 168.0M global + 65.1M per thread (151 max threads)
151 est  max_connections. 168M read_buffer_size + sort_buffer_size + join_buffer_size

[--] P_S Max memory usage: 72B
[--] Galera GCache Max memory usage: 0B
[OK] Maximum reached memory usage: 298.3M (9.97% of installed RAM)
[!!] Maximum possible memory usage: 9.8G (334.35% of installed RAM)
[!!] Overall possible memory usage with other process exceeded memory
```

```
[OK] Slow queries: 0% (0/22)
[OK] Highest usage of available connections: 1% (2/151)
[OK] Aborted connections: 0.00%  (0/15)
[!!] name resolution is active : a reverse name resolution is made for each new connection and can reduce performance
[--] Query cache have been removed in MySQL 8
[OK] Sorts requiring temporary tables: 0% (0 temp sorts / 2 sorts)
[!!] Joins performed without indexes: 1
[OK] Temporary tables created on disk: 0% (0 on disk / 5 total)
[OK] Thread cache hit rate: 86% (2 created / 15 connections)
[OK] Table cache hit rate: 64% (145 open / 224 opened)
[OK] table_definition_cache(2000) is upper than number of tables(315)
[OK] Open file limit used: 0% (2/8K)
[OK] Table locks acquired immediately: 100% (4 immediate / 4 locks)
[OK] Binlog cache memory access: 0% (0 Memory / 0 Total)

-------- Performance schema
------------------------------------------------------------------------
[--] Memory used by P_S: 72B
[--] Sys schema is installed.

-------- ThreadPool Metrics
------------------------------------------------------------------------
[--] ThreadPool stat is disabled.

-------- MyISAM Metrics
------------------------------------------------------------------------
[--] MyISAM Metrics are disabled on last MySQL versions.

-------- InnoDB Metrics
------------------------------------------------------------------------
[--] InnoDB is enabled.
[--] InnoDB Thread Concurrency: 0
[OK] InnoDB File per table is activated
[OK] InnoDB buffer pool / data size: 128.0M/49.0K
[!!] Ratio InnoDB log file size / InnoDB Buffer pool size (75 %): 48.0M * 2/128.0M should be equal to 25%
[OK] InnoDB buffer pool instances: 1
[--] Number of InnoDB Buffer Pool Chunk : 1 for 1 Buffer Pool Instance(s)
[OK] Innodb_buffer_pool_size aligned with Innodb_buffer_pool_chunk_size &
```

```
Innodb_buffer_pool_instances
[OK] InnoDB Read buffer efficiency: 94.02% (13442 hits/ 14297 total)
[OK] InnoDB Write log efficiency: 97.67% (630 hits/ 645 total)
[OK] InnoDB log waits: 0.00% (0 waits / 15 writes)

-------- AriaDB Metrics
-----------------------------------------------------------------------------
[--] AriaDB is disabled.

-------- TokuDB Metrics
-----------------------------------------------------------------------------
[--] TokuDB is disabled.

-------- XtraDB Metrics
-----------------------------------------------------------------------------
[--] XtraDB is disabled.

-------- Galera Metrics
-----------------------------------------------------------------------------
[--] Galera is disabled.

-------- Replication Metrics
-------------------------------------------------------------------------
[--] Galera Synchronous replication: NO
[--] No replication slave(s) for this server.
[--] Binlog format: ROW
[--] XA support enabled: ON
[--] Semi synchronous replication Master: Not Activated
[--] Semi synchronous replication Slave: Not Activated
[--] This is a standalone server

-------- Recommendations
-----------------------------------------------------------------------------
General recommendations:
    Control warning line(s) into /home/pilou/Formation/simpleinit/mysqld.log file
    Control error line(s) into /home/pilou/Formation/simpleinit/mysqld.log file
    MySQL was started within the last 24 hours - recommendations may be inaccurate
    Reduce your overall MySQL memory footprint for system stability
    Dedicate this server to your database for highest performance.
    Configure your accounts with ip or subnets only, then update your configuration with skip-
```

```
name-resolve=1
    We will suggest raising the 'join_buffer_size' until JOINs not using indexes are found.
              See https://dev.mysql.com/doc/internals/en/join-buffer-size.html
              (specially the conclusions at the bottom of the page).
    Before changing innodb_log_file_size and/or innodb_log_files_in_group read this:
https://bit.ly/2TcGgtU
Variables to adjust:
  *** MySQL's maximum memory usage is dangerously high ***
  *** Add RAM before increasing MySQL buffer variables ***
    join_buffer_size (> 256.0K, or always use indexes with JOINs)
    innodb_log_file_size should be (=16M) if possible, so InnoDB total log files size equals
to 25% of buffer pool size.
```

https://wiki.deimos.fr/MysqlTuner_:_Optimiser_votre_serveur_MySQL.html#Patch_mysqltuner

https://www.mysqlcalculator.com/

https://www.percona.com/blog/2016/05/03/best-practices-for-configuring-optimal-mysql-memory-usage/

My.ini sample

```
# MySQL Server Instance Configuration File
# ----------------------------------------------------------------------
# Generated by the MySQL Server Instance Configuration Wizard
#
#
# Installation Instructions
# ----------------------------------------------------------------------
#
# On Linux you can copy this file to /etc/my.cnf to set global options,
# mysql-data-dir/my.cnf to set server-specific options
# (@localstatedir@ for this installation) or to
# ~/.my.cnf to set user-specific options.
#
# On Windows you should keep this file in the installation directory
# of your server (e.g. C:\Program Files\MySQL\MySQL Server X.Y). To
# make sure the server reads the config file use the startup option
# "--defaults-file".
```

```
#
# To run run the server from the command line, execute this in a
# command line shell, e.g.
# mysqld --defaults-file="C:\Program Files\MySQL\MySQL Server X.Y\my.ini"
#
# To install the server as a Windows service manually, execute this in a
# command line shell, e.g.
# mysqld --install MySQLXY --defaults-file="C:\Program Files\MySQL\MySQL Server X.Y\my.ini"
#
# And then execute this in a command line shell to start the server, e.g.
# net start MySQLXY
#
#
# Guildlines for editing this file
# ------------------------------------------------------------------
#
# In this file, you can use all long options that the program supports.
# If you want to know the options a program supports, start the program
# with the "--help" option.
#
# More detailed information about the individual options can also be
# found in the manual.
#
#
# CLIENT SECTION
# ------------------------------------------------------------------
#
# The following options will be read by MySQL client applications.
# Note that only client applications shipped by MySQL are guaranteed
# to read this section. If you want your own MySQL client program to
# honor these values, you need to specify it as an option during the
# MySQL client library initialization.
#
[client]

port=3306

[mysql]

default-character-set=latin1
```

```
# SERVER SECTION
# ------------------------------------------------------------------------
#
# The following options will be read by the MySQL Server. Make sure that
# you have installed the server correctly (see above) so it reads this
# file.
#
[mysqld]

# The TCP/IP Port the MySQL Server will listen on
port=3306


#Path to installation directory. All paths are usually resolved relative to this.
basedir="C:/Program Files/MySQL/MySQL Server 5.1/"

#Path to the database root
datadir="C:/ProgramData/MySQL/MySQL Server 5.1/Data/"

# The default character set that will be used when a new schema or table is
# created and no character set is defined
character-set-server=latin1

# The default storage engine that will be used when create new tables when
default-storage-engine=INNODB

# Set the SQL mode to strict
sql-mode="STRICT_TRANS_TABLES,NO_AUTO_CREATE_USER,NO_ENGINE_SUBSTITUTION"

# The maximum amount of concurrent sessions the MySQL server will
# allow. One of these connections will be reserved for a user with
# SUPER privileges to allow the administrator to login even if the
# connection limit has been reached.
max_connections=100

# Query cache is used to cache SELECT results and later return them
# without actual executing the same query once again. Having the query
# cache enabled may result in significant speed improvements, if your
```

```
# have a lot of identical queries and rarely changing tables. See the
# "Qcache_lowmem_prunes" status variable to check if the current value
# is high enough for your load.
# Note: In case your tables change very often or if your queries are
# textually different every time, the query cache may result in a
# slowdown instead of a performance improvement.
query_cache_size=0

# The number of open tables for all threads. Increasing this value
# increases the number of file descriptors that mysqld requires.
# Therefore you have to make sure to set the amount of open files
# allowed to at least 4096 in the variable "open-files-limit" in
# section [mysqld_safe]
table_cache=256

# Maximum size for internal (in-memory) temporary tables. If a table
# grows larger than this value, it is automatically converted to disk
# based table This limitation is for a single table. There can be many
# of them.
tmp_table_size=205M


# How many threads we should keep in a cache for reuse. When a client
# disconnects, the client's threads are put in the cache if there aren't
# more than thread_cache_size threads from before.  This greatly reduces
# the amount of thread creations needed if you have a lot of new
# connections. (Normally this doesn't give a notable performance
# improvement if you have a good thread implementation.)
thread_cache_size=8

#*** MyISAM Specific options

# The maximum size of the temporary file MySQL is allowed to use while
# recreating the index (during REPAIR, ALTER TABLE or LOAD DATA INFILE.
# If the file-size would be bigger than this, the index will be created
# through the key cache (which is slower).
myisam_max_sort_file_size=100G

# If the temporary file used for fast index creation would be bigger
# than using the key cache by the amount specified here, then prefer the
```

```
# key cache method.  This is mainly used to force long character keys in
# large tables to use the slower key cache method to create the index.
myisam_sort_buffer_size=410M

# Size of the Key Buffer, used to cache index blocks for MyISAM tables.
# Do not set it larger than 30% of your available memory, as some memory
# is also required by the OS to cache rows. Even if you're not using
# MyISAM tables, you should still set it to 8-64M as it will also be
# used for internal temporary disk tables.
key_buffer_size=354M

# Size of the buffer used for doing full table scans of MyISAM tables.
# Allocated per thread, if a full scan is needed.
read_buffer_size=64K
read_rnd_buffer_size=256K

# This buffer is allocated when MySQL needs to rebuild the index in
# REPAIR, OPTIMZE, ALTER table statements as well as in LOAD DATA INFILE
# into an empty table. It is allocated per thread so be careful with
# large settings.
sort_buffer_size=256K


#*** INNODB Specific options ***


# Use this option if you have a MySQL server with InnoDB support enabled
# but you do not plan to use it.  This will save memory and disk space
# and speed up some things.
#skip-innodb

# Additional memory pool that is used by InnoDB to store metadata
# information.  If InnoDB requires more memory for this purpose it will
# start to allocate it from the OS.  As this is fast enough on most
# recent operating systems, you normally do not need to change this
# value. SHOW INNODB STATUS will display the current amount used.
innodb_additional_mem_pool_size=15M

# If set to 1, InnoDB will flush (fsync) the transaction logs to the
# disk at each commit, which offers full ACID behavior. If you are
```

```
# willing to compromise this safety, and you are running small
# transactions, you may set this to 0 or 2 to reduce disk I/O to the
# logs. Value 0 means that the log is only written to the log file and
# the log file flushed to disk approximately once per second. Value 2
# means the log is written to the log file at each commit, but the log
# file is only flushed to disk approximately once per second.
innodb_flush_log_at_trx_commit=1

# The size of the buffer InnoDB uses for buffering log data. As soon as
# it is full, InnoDB will have to flush it to disk. As it is flushed
# once per second anyway, it does not make sense to have it very large
# (even with long transactions).
innodb_log_buffer_size=7M

# InnoDB, unlike MyISAM, uses a buffer pool to cache both indexes and
# row data. The bigger you set this the less disk I/O is needed to
# access data in tables. On a dedicated database server you may set this
# parameter up to 80% of the machine physical memory size. Do not set it
# too large, though, because competition of the physical memory may
# cause paging in the operating system.  Note that on 32bit systems you
# might be limited to 2-3.5G of user level memory per process, so do not
# set it too high.
innodb_buffer_pool_size=686M

# Size of each log file in a log group. You should set the combined size
# of log files to about 25%-100% of your buffer pool size to avoid
# unneeded buffer pool flush activity on log file overwrite. However,
# note that a larger logfile size will increase the time needed for the
# recovery process.
innodb_log_file_size=343M

# Number of threads allowed inside the InnoDB kernel. The optimal value
# depends highly on the application, hardware as well as the OS
# scheduler properties. A too high value may lead to thread thrashing.
innodb_thread_concurrency=10
```

My.ini 2

```
#
# FromDual configuration file template for MySQL, Galera Cluster, MariaDB and Percona Server
# Location: %MYCNF%
# This template is intended to work with MySQL 5.7 and newer and MariaDB 10.0 and newer
# Get most recent updated from here:
# https://www.fromdual.com/mysql-configuration-file-sample
#

[client]

port                           = %PORT%                                 # default 3306
socket                         = %SOCKET%                               # Use mysqld.sock on
Ubuntu, conflicts with AppArmor otherwise


[mysql]

no_auto_rehash
max_allowed_packet             = 16M
prompt                         = '\u@\h [\d]> '                         # 'user@host [schema]> '
default_character_set          = utf8                                   # Possibly this setting
is correct for most recent Linux systems


[mysqldump]

max_allowed_packet             = 16M


[mysqld_safe]                                                           # Becomes sooner or later
obsolete with systemd

open_files_limit               = 8192                                   # You possibly have to
adapt your O/S settings as well
user                           = mysql
log-error                      = %INSTANCEDIR%/log/%UNAME%_%INSTANCE%_error.log   # Adjust
AppArmor configuration: /etc/apparmor.d/local/usr.sbin.mysqld


[mysqld]
```

```
# Connection and Thread variables

port                            = %PORT%                                # default 3306
socket                          = %SOCKET%                              # Use mysqld.sock on
Ubuntu, conflicts with AppArmor otherwise
basedir                         = %BASEDIR%
datadir                         = %DATADIR%
# tmpdir                          = '%INSTANCEDIR%/tmp'
# innodb_tmpdir                   = '%INSTANCEDIR%/tmp'                 # MySQL 5.7

max_allowed_packet              = 16M
default_storage_engine          = InnoDB
# explicit_defaults_for_timestamp = 1                                   # MySQL 5.6, test
carefully! This can have an impact on application.
# disable_partition_engine_check  = true                                # Since MySQL 5.7.17 to
5.7.20. To get rid of nasty message in error log

# character_set_server            = utf8mb4                             # For modern
applications, default in MySQL 8.0
# collation_server                = utf8mb4_general_ci


max_connections                 = 151                                 # Values < 1000 are
typically good
max_user_connections            = 145                                 # Limit one specific
user/application
thread_cache_size               = 151                                 # Up to max_connections
makes sense


# Query Cache (does not exist in MySQL 8.0 any more!)

# query_cache_type                = 1                                   # Set to 0 to avoid
global QC Mutex
# query_cache_size                = 32M                                 # Avoid too big (>
128M) QC because of QC clean-up lock!


# Session variables
```

```
sort_buffer_size              = 2M                                    # Could be too big for
many small sorts
tmp_table_size                = 32M                                   # Make sure your
temporary results do NOT contain BLOB/TEXT attributes

read_buffer_size              = 128k                                  # Resist to change this
parameter if you do not know what you are doing
read_rnd_buffer_size          = 256k                                  # Resist to change this
parameter if you do not know what you are doing
join_buffer_size              = 128k                                  # Resist to change this
parameter if you do not know what you are doing



# Other buffers and caches

table_definition_cache        = 1400                                  # As big as many tables
you have
table_open_cache              = 2000                                  # connections x
tables/connection (~2)
table_open_cache_instances    = 16                                    # New default in 5.7



# MySQL error log

log_error                     = %INSTANCEDIR%/log/%UNAME%_%INSTANCE%_error.log   # Adjust
AppArmor configuration: /etc/apparmor.d/local/usr.sbin.mysqld
# log_timestamps                = SYSTEM                                  # MySQL 5.7, equivalent
to old behaviour
log_warnings                  = 2                                     # MySQL 5.6, equivalent
to log_error_verbosity = 3
# log_error_verbosity           = 3                                     # MySQL 5.7, equivalent
to log_warnings = 2, MariaDB does NOT support this!
innodb_print_all_deadlocks    = 1
# wsrep_log_conflicts           = 1                                     # for Galera only!



# Slow Query Log

slow_query_log_file           = %INSTANCEDIR%/log/%UNAME%_%INSTANCE%_slow.log   # Adjust
AppArmor configuration: /etc/apparmor.d/local/usr.sbin.mysqld
slow_query_log                = 0
```

```
log_queries_not_using_indexes  = 0                                  # Interesting on
developer systems!
long_query_time                = 0.5
min_examined_row_limit         = 100


# General Query Log

general_log_file               = %INSTANCEDIR%/log/%UNAME%_%INSTANCE%_general.log   # Adjust
AppArmor configuration: /etc/apparmor.d/local/usr.sbin.mysqld
general_log                    = 0


# Performance Schema

# performance_schema             = ON                                 # for MariaDB 10
releases
performance_schema_consumer_events_statements_history_long = ON      # MySQL 5.6/MariaDB 10
and newer


# Binary logging and Replication

server_id                      = %SERVERID%                           # Must be set on MySQL
5.7 and newer if binary log is enabled!
log_bin                        = %INSTANCEDIR%/binlog/%UNAME%_%INSTANCE%_binlog            #
Locate outside of datadir, adjust AppArmor configuration:
/etc/apparmor.d/local/usr.sbin.mysqld
# master_verify_checksum         = ON                                 # MySQL 5.6
binlog_cache_size              = 1M
binlog_stmt_cache_size         = 1M
max_binlog_size                = 128M                                 # Make bigger for high
traffic to reduce number of files
sync_binlog                    = 1                                    # Set to 0 or higher to
increase write performance
expire_logs_days               = 5                                    # We will survive easter
holidays
binlog_format                  = ROW                                  # Use MIXED if you want
to experience some troubles
# binlog_row_image               = MINIMAL                            # Since 5.6
# auto_increment_increment       = 2                                  # For Master/Master
```

```
set-ups use 2 for both nodes
# auto_increment_offset          = 1                                     # For Master/Master
set-ups use 1 and 2


# Slave variables

log_slave_updates              = 1                                   # Use if Slave is used
for Backup and PiTR
read_only                      = 0                                   # Set to 1 to prevent
writes on Slave
# super_read_only                = 0                                   # Set to 1 to prevent
writes on Slave for users with SUPER privilege. Since 5.7, not in MariaDB
# skip_slave_start               = 1                                   # To avoid start of
Slave thread
# relay_log                      = %UNAME%_%INSTANCE%_relay-bin
# relay_log_info_repository      = table                               # MySQL 5.6
# master_info_repository         = table                               # MySQL 5.6
# slave_load_tmpdir              = '%INSTANCEDIR%/tmp'


# Crash-safe replication Master

# binlog_checksum                = CRC32                               # default
# sync_binlog                    = 1                                   # default since 5.7.6,
but slow!
# innodb_support_xa              = 1                                   # default, depracted
since 5.7.10


# Crash-safe replication Slave

# master_info_repository         = TABLE
# relay_log_info_repository      = TABLE
# relay_log_recovery             = 1
# sync_relay_log_info            = 1
# relay_log_purge                = 1                                   # default
# slave_sql_verify_checksum      = 1                                   # default


# GTID replication
```

```
# gtid_mode                         = ON                                    # Master and Slave
# enforce_gtid_consistency          = 1                                     # Master and Slave

# log_bin                           = %INSTANCEDIR%/binlog/%UNAME%_%INSTANCE%_binlog   # In 5.6
also on Slave
# log_slave_updates                 = 1                                     # In 5.6 also on
Slave


# Security variables

# local_infile                    = 0                                   # If you are security
aware
# secure_auth                     = 1                                   # If you are security
aware
# sql_mode                        =
TRADITIONAL,ONLY_FULL_GROUP_BY,NO_ENGINE_SUBSTITUTION,STRICT_ALL_TABLES,NO_ZERO_IN_DATE,NO_ZER
O_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_AUTO_CREATE_USER   # Be careful changing this afterwards
# skip_name_resolve               = 0                                   # Set to 1 if you do
not trust your DNS or experience problems
# secure_file_priv                = '%INSTANCEDIR%/tmp'                   # chmod 750, adjust
AppArmor configuration: /etc/apparmor.d/local/usr.sbin.mysqld


# MyISAM variables

key_buffer_size                 = 8M                                  # Set to 25 - 33 % of RAM
if you still use MyISAM
myisam_recover_options          = 'BACKUP,FORCE'
# disabled_storage_engines        = 'MyISAM,MEMORY'                       # MySQL 5.7, do NOT
during/before mysql_upgrade, good for Galera!


# MEMORY variables

max_heap_table_size             = 64M                                 # Should be greater or
equal to tmp_table_size


# InnoDB variables
```

```
innodb_strict_mode             = ON
# innodb_file_format_check       = 1                                 # Desupported in MySQL
8.0
# innodb_file_format             = Barracuda                         # For dynamic and
compressed InnoDB tables, default in 5.7
innodb_buffer_pool_size        = 128M                              # Go up to 80% of your
available RAM
innodb_buffer_pool_instances   = 8                                 # Bigger if huge InnoDB
Buffer Pool or high concurrency

innodb_file_per_table          = 1                                 # Is the recommended way
nowadays
# innodb_flush_method            = O_DIRECT                        # O_DIRECT is sometimes
better for direct attached storage
# innodb_write_io_threads        = 8                               # If you have a strong
I/O system or SSD
# innodb_read_io_threads         = 8                               # If you have a strong
I/O system or SSD
# innodb_io_capacity             = 1000                            # If you have a strong
I/O system or SSD

innodb_flush_log_at_trx_commit = 2                                 # 1 for durability, 0 or
2 for performance
innodb_log_buffer_size         = 8M                                # Bigger if
innodb_flush_log_at_trx_commit = 0
innodb_log_file_size           = 256M                              # Bigger means more write
throughput but longer recovery time


                                                                   # Since MariaDB 10.0 and
MySQL 5.6
innodb_monitor_enable = all                                        # Overhead < 1% according
to PeterZ/Percona



# Galera specific MySQL parameter

# default_storage_engine         = InnoDB                            # Galera only works with
InnoDB
# innodb_flush_log_at_trx_commit = 2                                 # Durability is achieved
by committing to the Group
```

```
# innodb_autoinc_lock_mode       = 2                                # For parallel applying
# binlog_format                  = row                              # Galera only works with
RBR
# query_cache_type               = 0                                # Use QC with Galera only
in a Master/Slave set-up
# query_cache_size               = 0



# WSREP parameter

# wsrep_on                       = on                                 # Only MariaDB >= 10.1
# wsrep_provider                 = /usr/lib/galera/libgalera_smm.so    # Location of Galera
Plugin on Ubuntu ?
# wsrep_provider                 = /usr/lib64/galera-3/libgalera_smm.so   # Location of Galera
Plugin on CentOS 7
# wsrep_provider                 = none                               # Start mysqld without
Galera
# wsrep_provider_options         = 'gcache.size = 1G'                 # Depends on you
workload, WS kept for IST

# wsrep_cluster_name             = "My cool Galera Cluster"           # Same Cluster name for
all nodes
# wsrep_cluster_address          = "gcomm://192.168.0.2,192.168.0.3"   # Start other nodes
like this

# wsrep_node_name                = "Node A"                           # Unique node name
# wsrep_node_address             = 192.168.0.1                        # Our address where
replication is done
# wsrep_node_incoming_address    = 10.0.0.1                           # Our external
interface where application comes from
# wsrep_sync_wait                = 1                                  # If you need realy
full-synchronous replication (Galera 3.6 and newer)
# wsrep_slave_threads            = 16                                 # 4 - 8 per core, not
more than wsrep_cert_deps_distance

# wsrep_sst_method               = rsync                              # SST method (initial
full sync): mysqldump, rsync, rsync_wan, xtrabackup-v2
# wsrep_sst_auth                 = sst:secret                         # Username/password for
sst user
# wsrep_sst_receive_address      = 192.168.2.1                        # Our address where to
receive SST
```

```
# Group Replication parameter

# default_storage_engine         = InnoDB                                # Group Replication
only works with InnoDB
# server_id                      = %SERVERID%                            # Should be different
on all 3 nodes
# log_bin                        = %INSTANCEDIR%/binlog/%UNAME%_%INSTANCE%_binlog   # Locate
outside of datadir, adjust AppArmor configuration: /etc/apparmor.d/local/usr.sbin.mysqld
# binlog_format                  = ROW
# binlog_checksum                = NONE                                  # not default!
# gtid_mode                      = ON
# enforce_gtid_consistency       = ON
# master_info_repository         = TABLE
# relay_log_info_repository      = TABLE
# log_slave_updates              = ON

# slave_parallel_workers         = <n>                                   # 1-2/core, max. 10
# slave_preserve_commit_order    = ON
# slave_parallel_type            = LOGICAL_CLOCK

# transaction_write_set_extraction            = XXHASH64

# loose-group_replication_group_name          = "$(uuidgen)"            # Must be the same on
all nodes
# loose-group_replication_start_on_boot       = OFF
# loose-group_replication_local_address       = "192.168.0.1"
# loose-group_replication_group_seeds         = "192.168.0.1,192.168.0.2,192.168.0.3"    # All
nodes of Cluster
# loose-group_replication_bootstrap_group     = OFF
# loose-group_replication_single_primary_mode = FALSE                   # = multi-primary
```