How to setup MariaDB Galera Cluster 10.0 on CentOS

Tables Of Contents

注意：

- 定義每台server爲 cluster node；
- 以下操作可能需要sudo或root權限；
- 若沒有特別說明，則操作須在所有cluster node上都執行；
- 爲防止`split brain`的發生，cluster需至少(≥3)個node；
- 第一台cluster node的配置、cluster啓動與之後的cluster node不同；

假令三台server的ip分別是

- 192.168.1.1
- 192.168.1.2
- 192.168.1.3

Preparing The Server

`依據來源`：PREPARING THE SERVER

在安裝`MariaDB Galera Cluster 10.0`之前，須先在 **每個** cluster node上進行如下操作。

1. Disabling SELinux for mysqld
2. Firewall Configuration
3. Removing Postfix

Disabling SELinux for mysqld

```
setenforce 0
或
echo 0 > /selinux/enforce
```

Firewall Configuration

允許端口號`3306`、`4444`、`4567`、`4568`通信

Removing Postfix

在`Red Hat Enterprise Linux`和`Fedora`，如果yum安裝`Galera Cluster`時不先移除`postfix`，可能會報錯。

```
yum remove postfix
```

Add MariaDB Repository

在 **每個** cluster node路徑`/etc/yum.repos.d/`下創建MariaDB倉庫`MariaDB.repo`。根據Server System Info，在Setting up MariaDB Repositories中依次選擇`Distro` -> `Release` -> `Version`，將生成的倉庫信息填入`MariaDB.repo`。

註：因MariaDB Galera Cluster當前Stable release版本號分別是`10.0.21`和`5.5.46`，故`Version`應選擇`10.0`或`5.5`，此處選擇`10.0`。

`/etc/yum.repos.d/MariaDB.repo`建好後，通過yum命令安裝MariaDB Galera Cluster，如果Server上已經安裝有MySQL或MariaDB，請先用`yum remove`命令刪除。

操作命令

```
yum remove [package_name] #刪除已經存在的MySQL或MariaDB
yum clean all
yum make cache
```

Install MariaDB Galera Cluster 10.0

Install The socat Package

爲了成功安裝`MariaDB Galera Cluster 10.0`，需先安裝`socat`包，可通過倉庫`EPEL`安裝。

*CentOS6*最小化安裝默認沒有安裝該包，*CentOS7*最小化安裝中已集成該包。——`沒有驗證過`

```
yum install epel-release
yum install socat
```

Install MariaDB and Galera

`依據來源`

1. Installing MariaDB Galera Cluster with YUM
2. INSTALLING GALERA CLUSTER

如果cluster node已經安裝有`MariaDB-server`，則需要移除，已有的數據庫不受影響，但最好先備份數據再操作。

```
yum remove MariaDB-server
```

安裝`MariaDB Galera Cluster`

```
yum install MariaDB-Galera-server MariaDB-client
galera
```

Setup MariaDB Security

MariaDB初始化設置，先啓動`mysql`服務，在執行`mysql_secure_installation`

1. 啓動`mysql`服務

```
#通用
/etc/init.d/mysql start

#CentOS6
service mysql start

#CentOS7
systemctl start mysql
```

2. 設置`mysql`服務開機啓動

```
#CentOS6
chkconfig mysql on

#CentOS7
systemctl enable mysql
```

3.執行`mysql_secure_installation`

該操作用以提高MariaDB安全性。
MariaDB文檔 mysql_secure_installation

```
mysql_secure_installation
```

可在此操作爲root用戶創建密碼，並確定是否允許遠程登錄。

Create MariaDB Galera Cluster users

創建可以訪問數據庫的用戶帳號，該帳號用於數據庫node之間在State Snapshot Transfer(SST)下彼此進行認證。

以創建用戶`cluster`，密碼`cluster12345`爲例，在`每個` cluster node上進行如下操作。

登錄數據庫

```
mysql -uroot -p
```

登入數據庫後，依次執行

```
DELETE FROM mysql.user WHERE user='';
GRANT ALL ON *.* TO 'cluster'@'%' IDENTIFIED BY
'cluster12345';
FLUSH PRIVILEGES;
exit
```

注：

1. `%`代表任意主機，可設置爲某一具體的Host IP；
2. `ALL`代表除`GRANT OPTION`以外的所有privileges，可按實際需求設置。

Create Galera Cluster config

1. 關閉所有cluster node的`mysql`服務

```
#通用
/etc/init.d/mysql stop

#CentOS6
service mysql stop

#CentOS7
systemctl start stop
```

2. 配置cluster node
   在 **每個** cluster node的路徑`/etc/my.cnf.d/`下新建文件`server.cnf`，進行參數配置，放置在`server.cnf`中的option`[mariadb]`或`[mysqld]`或`[mariadb-10.0]`下

```
vim /etc/my.cnf.d/server.cnf
```

基本參數

```
# a path to Galera library
wsrep_provider

# Cluster connection URL containing the IPs of other
nodes in the cluster
wsrep_cluster_address

# method used for the state snapshot transfer,4
kinds of value,there is rsync,
mysqldump,xtrabackup,xtrabackup-v2,default is rsync
wsrep_sst_method

# used to set up the unique node name
wsrep_node_name

# In o rder for Galera to work correctly binlog
format should be ROW
binlog_format=ROW

# MyISAM storage engine has only experimental
support
default_storage_engine=InnoDB

# This changes how InnoDB autoincrement locks are
managed
innodb_autoinc_lock_mode=2
```

額外參數

`Authentication for SST method`

```
wsrep_sst_auth=user:password #此處的user和password就是
上文Create MariaDB Galera Cluster users中創建的用戶
cluster，密碼cluster12345
```

For MariaDB Galera cluster, `query_cache_size` sholud be disabled

> Limited support for Query Cache has been implemented. Query cache cannot still be fully enabled during the startup.
>
> To enable query cache, mysqld should be started with `query_cache_type`=1 and `query_cache_size=0` and then `query_cache_size` should be changed to desired value during runtime.

```
query_cache_size=0
query_cache_type=0
```

參見 Query Cache

參數配置，之後的cluster node與 1st cluster node相比，3個參數不一樣

```
wsrep_cluster_address
wsrep_node_address
wsrep_node_name
```

> Although note that cluster membership is not defined by `wsrep_cluster_address` setting, it is defined by the nodes that join the cluster with the proper cluster name configured Variable `wsrep_cluster_name` is used for that, if not explicitly set it will default to `my_wsrep_cluster`.
>
> Hence, **variable `wsrep_cluster_address` does not need to be identical on all nodes**, it's just a best practice because on restart the node will try all other nodes in that list and look for any that are currently up and running the cluster.

1st Server

```
query_cache_size=0
binlog_format=ROW
default_storage_engine=innodb
innodb_autoinc_lock_mode=2

wsrep_provider=/usr/lib/galera/libgalera_smm.so
wsrep_cluster_address="gcomm://192.168.1.1,192.168.1
.2,192.168.1.3"
wsrep_cluster_name='cluster_sample'
wsrep_node_address='192.168.1.1'
wsrep_node_name='node1'
wsrep_sst_method=rsync
wsrep_sst_auth=cluster:cluster12345
```

2nd Server

```
query_cache_size=0
binlog_format=ROW
default_storage_engine=innodb
innodb_autoinc_lock_mode=2

wsrep_provider=/usr/lib/galera/libgalera_smm.so
wsrep_cluster_address="gcomm://192.168.1.1,192.168.1
.2,192.168.1.3"
wsrep_cluster_name='cluster_sample'
wsrep_node_address='192.168.1.2'
wsrep_node_name='node2'
wsrep_sst_method=rsync
wsrep_sst_auth=cluster:cluster12345
```

3rd Server

```
query_cache_size=0
binlog_format=ROW
default_storage_engine=innodb
innodb_autoinc_lock_mode=2

wsrep_provider=/usr/lib/galera/libgalera_smm.so
wsrep_cluster_address="gcomm://192.168.1.1,192.168.1
.2,192.168.1.3"
wsrep_cluster_name='cluster_sample'
wsrep_node_address='192.168.1.3'
wsrep_node_name='node3'
wsrep_sst_method=rsync
wsrep_sst_auth=cluster:cluster12345
```

Initialize The First Cluster Node

參見 Bootstrapping a new cluster

使用選項`--wsrep-new-cluster`啓動1st Server的mysql服務，這樣cluster的primary node就被初始化設置。

```
/etc/init.d/mysql start --wsrep-new-cluster
```

可用如下命令查看cluster狀態

```
mysql -uroot -p -e "show status like 'wsrep%'"
```

Add The Other Cluster Nodes

在其它cluster node上順序執行

```
#通用
/etc/init.d/mysql start

#CentOS6
service mysql start

#CentOS7
systemctl start mysql
```

server一台一台順序啓動

可用如下命令查看cluster狀態

```
mysql -uroot -p -e "show status like 'wsrep%'"
```

當

> wsrep_local_state_comment = Synced
> wsrep_connected = ON
> wsrep_ready = ON
> wsrep_cluster_size = “節點數”
> wsrep_incoming_addresses = “node ip地址列表，逗號間隔”

可以確定MariaDB Galera Cluster搭建成功，可以進行讀寫測試，看是否同步。

Related Website

1. MariaDB
2. Percona

3. Galera Cluster

Reference Blog

1. **Getting Started with MariaDB Galera Cluster**
2. **Percona XtraDB Cluster Release 5.6.26-25.12 Operations Manual**
3. **GALERA CLUSTER DOCUMENTATION**
4. How to Setup MariaDB Galera Cluster 10.0 on CentOS/RedHat & Fedora
5. How To Setup MariaDB Galera Cluster 10.0 On CentOS
6. How to setup MariaDB Galera Cluster 10.0 on CentOS

Writer：馬雪東

Note Time：2015.11.13 16:43