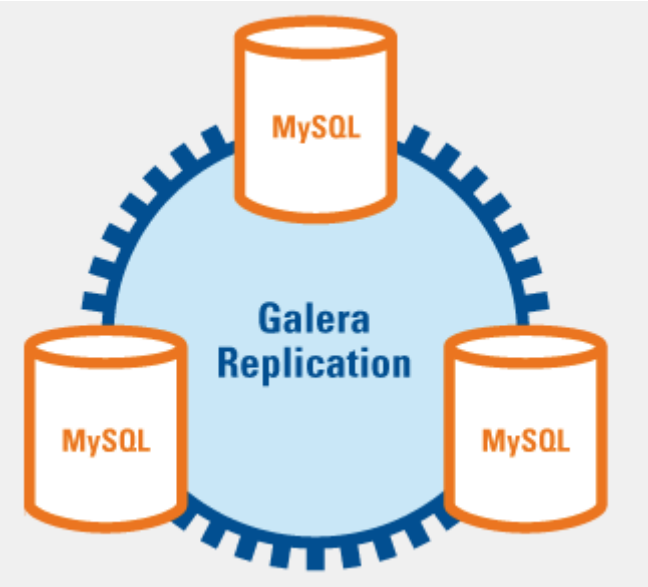
목차

Galera Cluster 구성

DBMS 중에 가장 많이 사용되고 있는 것은 무료로 사용할 수 있는 MySQL이지 않을까 한다. 웹 서비스의 규모가 커지다보니 컴퓨팅 리소스도 많이 필요하게되고 이중화도 필요한 부분이 생기게되었다. 많은 이중화 솔루션 중에 Synchronous Multi-Master Cluster인 Galera Cluster를 소개하고 구성까지 해보려고 한다.



MySQL Cluster CGE의 NDB라는 Storage Engine을 사용하지 않고 MySQL/InnoDB를 지원한다. 따라서, InnoDB에서만 지원된다는 제한사항이 있으니 참고바랍니다.

또한, Galera Cluster에서 권장하는 노드 수는 3node이상입니다. 하지만 이번에 구성은 2node로 진행 할 예정입니다 :)

Galera Cluster의 기능

장점

- 다중 마스터를 지원하므로 Active-Active 구성이 가능하다.
- 동기 복제로 지연이 없으며 데이터 손실이 되지 않는다.
- 장애 조치(Failover)가 없기 때문에 서비스에 연속성을 유지할 수 있다.
- InnoDB 지원
- 모든 노드가 읽기/쓰기하기 때문에 분할하여 사용할 필요가 없다.
- 자동 노드 프로비저닝이 가능하다.

단점

- 모든 노드가 동일한 상태를 유지하기 때문에 노드간 분기 된 데이터를 구성할 수 없다.
- 모든 서버 노드에 동일한 데이터를 유지해야하므로 저장공간이 낭비된다.
- 신규 노드 추가 시 데이터를 복사해야 한다.(부하)

Galera Network Port

- `Standard MariaDB Port(default : 3306)` : MariaDB 기본 포트, State Snapshot Transfers(mysqldump)
- `Galera Peplication Port(default : 4567)` : Galera Cluster replication traffic, UDP & TCP
- `IST Port(default : 4568)` : Incremental State Transfers(증분)
- `SST Port(default : 4444)` : 3306 외의 모든 State Snapshot Transfers

구성 과정 흐름도

1. MariaDB 설치
2. 노드별 설정
3. 최초 노드서버 Clustering 초기화
4. 노드 Cluster에 추가
5. Clustering 확인
6. DB Sync 확인

MariaDB Clustering 구성 환경

- OS : CentOS-7.3-64
- DB : MariaDB 10.2
- Server One IP : 10.41.226.226
- Server Two IP : 10.41.227.100

실 구축이 아닌 Test용으로 구성하기때문에 방화벽은 in/out 모두 오픈하여 진행하였다. 오픈해야하는 Port정보는 Part 1에서 확인이 가능합니다. 또한 `Galera Cluster`는 3개의 `node`를 `Clustering`하는 것을 권장하고 있지만 2 node로도 충분히 구성이 가능하므로 2개의 서버로 진행한다 ^^

Install MariaDB

MariaDB 공식사이트에 들어가면 아주 진철하게 설치하는 방법이 소개되어 있다.

- MariaDB Site : https://mariadb.org/
- MariaDB Repositories Download : https://downloads.mariadb.org/mariadb/repositories/#mirror=iweb

나의 구성환경과 동일하게 선택하면 아래 그림과 같이 경로와 파일정보를 제공해준다.

To generate the entries select an item from each of the boxes below. Once an item is selected in each box, your customized repository configuration will appear below.

1. Choose a Distro

- SLES
- openSUSE
- Arch Linux
- Mageia
- Fedora
- **CentOS**
- RedHat
- Mint
- Ubuntu
- Debian

2. Choose a Release

- CentOS 8 (ppc64le)
- CentOS 8 (x86_64)
- CentOS 7 (ppc64le)
- CentOS 7 (ppc64)
- **CentOS 7 (x86_64)**
- CentOS 6 (x86_64)
- CentOS 6 (x86)

3. Choose a Version

- 10.5 [Beta]
- 10.4 [Stable]
- 10.3 [Old Stable]
- **10.2 [Old Stable]**
- 10.1 [Old Stable]
- 5.5 [Old Stable]

Here is your custom MariaDB YUM repository entry for CentOS. Copy and paste it into a file under /etc/yum.repos.d/ (we suggest naming the file MariaDB.repo or something similar).

```
# MariaDB 10.2 CentOS repository list - created 2020-06-04 00:32 UTC
# http://downloads.mariadb.org/mariadb/repositories/
[mariadb]
name = MariaDB
baseurl = http://yum.mariadb.org/10.2/centos7-amd64
gpgkey=https://yum.mariadb.org/RPM-GPG-KEY-MariaDB
gpgcheck=1
```

After the file is in place, install MariaDB with:

```
sudo yum install MariaDB-server MariaDB-client
```

If you haven't already accepted the MariaDB GPG key, you will be prompted to do so. See "Installing MariaDB with yum" for detailed information.

Please see Installing OQGraph for details on additional install steps needed for that storage engine.

MariaDB 설치 (Server One, Server Two)

```
# Repository를 생성
[root@galera-01 ~]# vi /etc/yum.repo.d/MariaDB.repo


# MariaDB.repo 정보
# MariaDB 10.2 CentOS repository list - created 2020-06-04 00:40 UTC
# http://downloads.mariadb.org/mariadb/repositories/
[mariadb]
name = MariaDB
baseurl = http://yum.mariadb.org/10.2/centos7-amd64
gpgkey=https://yum.mariadb.org/RPM-GPG-KEY-MariaDB
gpgcheck=1


# Install MariaDB
[root@galera-01 ~]# yum install MariaDB-server MariaDB-client
================================================================================
============================
 Package                         Arch          Version
Repository      Size
================================================================================
```

```
============================
Installing:
 MariaDB-client                 x86_64        10.2.32-1.el7.centos
mariadb         11 M
 MariaDB-compat                 x86_64        10.2.32-1.el7.centos
mariadb        2.2 M
     replacing  mariadb-libs.x86_64 1:5.5.52-1.el7
 MariaDB-server                 x86_64        10.2.32-1.el7.centos
mariadb         24 M
Installing for dependencies:
 MariaDB-common                 x86_64        10.2.32-1.el7.centos
mariadb         81 k
 boost-program-options          x86_64        1.53.0-28.el7
base           156 k
  galera                         x86_64        25.3.29-1.rhel7.el7.centos
mariadb        8.2 M
 perl-Compress-Raw-Bzip2        x86_64        2.061-3.el7
base            32 k
 perl-Compress-Raw-Zlib         x86_64        1:2.061-4.el7
base            57 k
 perl-DBI                       x86_64        1.627-4.el7
base           802 k
 perl-IO-Compress               noarch        2.061-2.el7
base           260 k
 perl-Net-Daemon                noarch        0.48-5.el7
base            51 k
 perl-PlRPC                     noarch        0.2020-14.el7
base            36 k
Updating for dependencies:
 openssl                        x86_64        1:1.0.2k-19.el7
base           493 k
 openssl-libs                   x86_64        1:1.0.2k-19.el7
base           1.2 M

Transcription Summary
================================================================================
============================


# MariaDB version 확인
[root@galera-01 ~]# mysql --version
mysql  Ver 15.1 Distrib 10.2.32-MariaDB, for Linux (x86_64) using readline
5.1
```

MariaDB Setup Root (Server One, Server

Two)

MariaDB를 시작하고 Setup을 진행한다. Setting이 완료되면 DB를 Stop해준다.

```
# Start MariaDB
[root@galera-01 ~]# service mysql start
Starting mysql (via systemctl):                            [  OK  ]
[root@galera-01 ~]# mysql_secure_installation


# Setting
Enter current password for root (enter for none):
OK, successfully used password, moving on...

Set root password? [Y/n] y
New password:
Re-enter new password:
Password updated successfully!
Reloading privilege tables..
 ... Success!

 Remove anonymous users? [Y/n] y
 ... Success!

 Disallow root login remotely? [Y/n] n
 ... skipping.

 Remove test database and access to it? [Y/n] y
  - Dropping test database...
 ... Success!
 - Removing privileges on test database...
 ... Success!

 Reload privilege tables now? [Y/n] y
 ... Success!

 Thanks for using MariaDB!


# Stop MariaDB
[root@galera-01 ~]# service mysql stop
Stopping mysql (via systemctl):                            [  OK  ]
```

Galera Cluster 설정

Galera Cluster를 처음으로 구성하는 Node와 추가되는 Node의 설정값에 조금 차이가 있으니 주의해서

설정을 진행한다. /etc/my.cnf.d/server.xml을 수정할 것이고 [galera]부분만 수정을 진행하면 된다.

Node1(Server One) Galera Cluster 설정

```
# Config node1(Server One) galera clutser
[root@galera-01 ~]# vi /etc/my.cnf.d/server.cnf

[galera]
# Mandatory settings
wsrep_on=ON
wsrep_provider=/usr/lib64/galera/libgalera_smm.so    # 초기설치 시 기본경로
wsrep_cluster_address='gcomm://'                     # Galera 초기설정 시 미입
력
wsrep_cluster_name='cluster'                         # Node1/2 동일하게
wsrep_node_address='10.41.226.226'                   # Node1(Server One) IP
wsrep_node_name='galera01'                           # Node1 Name
wsrep_sst_method=rsync
binlog_format=row
default_storage_engine=InnoDB
innodb_autoinc_lock_mode=2
bind-address=0.0.0.0
```

Node2(Server Two) Galera Cluster 설정

```
# Config node2(Server Two) galera clutser
[root@galera-02 ~]# vi /etc/my.cnf.d/server.cnf

[galera]
# Mandatory settings
wsrep_on=ON
wsrep_provider=/usr/lib64/galera/libgalera_smm.so    # 초기설치 시 기본경로
wsrep_cluster_address='gcomm://10.41.226.226'        # Galera Clustering
IP(node1)
wsrep_cluster_name='cluster'                         # Node1/2 동일하게
wsrep_node_address='10.41.227.100'                   # Node2(Server One) IP
wsrep_node_name='galera02'                           # Node2 Name
wsrep_sst_method=rsync
binlog_format=row
default_storage_engine=InnoDB
innodb_autoinc_lock_mode=2
bind-address=0.0.0.0
```

Galera Clustering 작업

초기 설정이 완료되었으니 Node1,2를 Clustering을하여 DB가 잘 연동되었는지 확인해보겠습니다. 확인

후에 추가로 Node1(Server One)의 초기설정을 변경해야하는 중요한 작업이 있으니 꼭 확인해야합니다. 이 부분은 뒤에서 좀 더 자세히 설명하겠습니다.

Node 1 Galera Cluster 초기화

앞서 Node 1,2 모두 초기 설정을 완료했으니 node1에서 cluster를 초기화하는 작업이 먼저 필요합니다.

```
# Galera Cluster 초기화
[root@galera-01 ~]# service mysql start --wsrep-new-cluster
Starting mysql (via systemctl):                            [  OK  ]

# node1(Server One) 상태 확인
[root@galera-01 ~]# ps -ef | grep mysql
mysql     14773  14619  0 13:30 ?        00:00:00 /usr/sbin/mysqld
--basedir=/usr
--datadir=/var/lib/mysql
--plugin-dir=/usr/lib64/mysql/plugin
--user=mysql
--wsrep_on=ON
--wsrep_provider=/usr/lib64/galera/libgalera_smm.so
--log-error=/var/lib/mysql/galera-01.err
--pid-file=/var/lib/mysql/galera-01.pid
--wsrep_start_position=a4fe716e-a609-11ea-95ab-fa16ac4cfe80:0
```

Node 2 Galera Cluster 추가

```
# Galera Cluster 초기화
[root@galera-02 ~]# service mysql start
Starting mysql (via systemctl):                            [  OK  ]

# node2(Server Two) 상태 확인
[root@galera-02 ~]# ps -ef | grep mysql
mysql     16041  15887  0 13:46 ?        00:00:00 /usr/sbin/mysqld
--basedir=/usr
--datadir=/var/lib/mysql
--plugin-dir=/usr/lib64/mysql/plugin
--user=mysql
--wsrep_on=ON
--wsrep_provider=/usr/lib64/galera/libgalera_smm.so
--log-error=/var/lib/mysql/galera-02.err
--pid-file=/var/lib/mysql/galera-02.pid
--wsrep_start_position=00000000-0000-0000-0000-000000000000:-1
```

Clustering 상태 확인

MariaDB에 접속하여 Cluster가 잘 적용되었는지 확인하겠습니다. 또한 아래내용과 같이 검색을하면 각

노드의 값이 동일하게 적용되어 Cluster가 잘 구성된 것을 확인 할 수 있습니다. 'wsrep%'값을 일부만 발취했습니다.

```
# MariaDB 접속
[root@galera-01 ~]# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 12
Server version: 10.2.32-MariaDB MariaDB Server

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input
statement.

MariaDB [(none)]> show status like 'wsrep%';
+------------------------------+----------------------------------------+
| Variable_name                | Value                                  |
+------------------------------+----------------------------------------+
| wsrep_cluster_status         | Primary                                |
| wsrep_connected              | ON                                     |
| wsrep_incoming_addresses     | 10.41.226.226:3306,10.41.227.100:3306 |
| wsrep_local_state_comment    | Synced                                 |
| wsrep_provider_name          | Galera                                 |
| wsrep_ready                  | ON                                     |
+------------------------------+----------------------------------------+
```

DB 생성으로 데이터 sync 확인

node1에서 galeradb database를 생성하여 실시간으로 node2에서도 생성되었는지 확인해보자.

```
# node1에서 database 생성
[root@galera-01 ~]# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 12
Server version: 10.2.32-MariaDB MariaDB Server

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input
statement.

MariaDB [(none)]> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
```

```
| mysql              |
| performance_schema |
+--------------------+

MariaDB [(none)]> create database galeradb;
Query OK, 1 row affected (0.00 sec)

MariaDB [(none)]> show databases;
+--------------------+
| Database           |
+--------------------+
| galeradb           |
| information_schema |
| mysql              |
| performance_schema |
+--------------------+
4 rows in set (0.00 sec)

# node2에 sync되어 생성되었는지 확인
[root@galera-02 ~]# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 13
Server version: 10.2.32-MariaDB MariaDB Server

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input
statement.

MariaDB [(none)]>
MariaDB [(none)]>
MariaDB [(none)]> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
+--------------------+
3 rows in set (0.00 sec)

MariaDB [(none)]> show databases;
+--------------------+
| Database           |
+--------------------+
| galeradb           |
| information_schema |
| mysql              |
| performance_schema |
+--------------------+
```

```
4 rows in set (0.00 sec)
```

중요 완료 후 node1(Server One) 설정 변경

모든 Test가 완료되었습니다. 하지만 아직 마지막으로 중요한 작업이 남아있습니다. 현 상태로 두게되면 DB를 restart했을때 에러가 발생하게 됩니다. 그 이유는 node1에 wsrep_cluster_address='gcomm://'의 값이 아무것도 없기때문에 발생하는 에러로 cluster에 포함된 IP들을 추가해야 합니다.

```
# Config node1(Server One) galera clutser
[root@galera-01 ~]# vi /etc/my.cnf.d/server.cnf

[galera]
# Mandatory settings
wsrep_on=ON
wsrep_provider=/usr/lib64/galera/libgalera_smm.so
wsrep_cluster_address='gcomm://10.41.227.100'        # Galera Cluster IP 추
가
wsrep_cluster_name='cluster'
wsrep_node_address='10.41.226.226'
wsrep_node_name='galera01'
wsrep_sst_method=rsync
binlog_format=row
default_storage_engine=InnoDB
innodb_autoinc_lock_mode=2
bind-address=0.0.0.0
```

출처

- https://syspago.tistory.com/2?category=1126943