

# mysql 기반의 리플리케이션 및 mmm 설치

## MYSQL 설치

### 1. 설치환경 구성1. Mysql사용자정보 추가

```
[root@localhost ~]# groupadd -g 1002 mysql
[root@localhost ~]# useradd -M mysql -u 1002 -g 1002 -d /svc/mysql/
```

#### 1. 추가 패키지 설치

```
[root@localhost ~]# yum install ncurses* openssl-devel* \
gdbm-devel* curl* gmp* libxml2* libtermcap* cmake* libc-client-devel\
libcurl-devel libxslt -y
```

#### 2. 바이너리&환경설정 설치패키지&환경설정 다운로드

```
[root@localhost ~]# tar -zxf mysql-5.6.34-linux-glibc2.5-x86_64.tar.gz
[root@localhost ~]# mv mysql-5.6.34-linux-glibc2.5-x86_64 /svc/mysql
[root@localhost ~]# mv -f my.cnf /etc/my.cnf
[root@localhost ~]# cp /svc/mysql/
[root@localhost ~]# ./scripts/mysql_install_db --user=mysql --datadir=/svc/db_data/
[root@localhost ~]# chown -R root: /svc/mysql/
[root@localhost ~]# chown -R mysql: /svc/db_data/
[root@localhost ~]# cp support-files/mysql.server /etc/init.d/mysqld
[root@localhost ~]# chmod +x /etc/init.d/mysqld
```

#### 3. 서비스 등록 및 실행

```
[root@localhost ~]# /etc/init.d/mysqld start
Starting mysqld: [ OK ]
```

### 2. MYSQL 이중화 구성1. 서버 ID설정 (Active/Standby 모두 설정)

```
[root@localhost ~]# vi /etc/my.cnf
...
server-id = 1 (Active는 1, Standby는 2로 설정)
```

#### 1. 리플리케이션 권한 설정서버 리플리케이션 설정 (Active/Standby 모두 설정)

```
mysql> grant replication slave on . to 'repl'@'상대방DB_IP' identified by 'repl';
mysql> flush privileges;
Query OK, 0 rows affected (0.00 sec)
```

#### 2. 액티브 서버에서 수행

```
mysql> show master status \G;

1. row *****
File: binlog.000004
Position: 722
Binlog_Do_DB:
Binlog_Ignore_DB:
Executed_Gtid_Set:
1 row in set (0.00 sec)

ERROR:
No query specified
```

#### 3. 스탠바이서버에서 수행

```
mysql> change master to master_host='상대방DB_IP', master_user='repl', master_password='repl', master_log_file='binlog.000004',
master_log_pos=722;
Query OK, 0 rows affected, 2 warnings (0.02 sec)
```

#### 4. 리플리케이션 연동 (각 DB서버에서 수행)

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

#### 5. Standby 시작 및 상태 확인 (Active/Standby 모두 설정)

```
mysql> show slave status \G;  
***** 1. row *****  
  
Slave_IO_State: Waiting for master to send event  
Master_Host: 상대방DB_IP  
Master_User: repl  
Master_Port: 3306  
Connect_Retry: 60  
Master_Log_File: binlog.000004  
Read_Master_Log_Pos: 722  
Relay_Log_File: relay-bin.000007  
Relay_Log_Pos: 280  
Relay_Master_Log_File: binlog.000005  
Slave_IO_Running: Yes  
Slave_SQL_Running: Yes  
Replicate_Do_DB:  
Replicate_Ignore_DB:  
Replicate_Do_Table:  
Replicate_Ignore_Table:  
Replicate_Wild_Do_Table:  
Replicate_Wild_Ignore_Table:  
Last_Errno: 0  
Last_Error:  
Skip_Counter: 0  
Exec_Master_Log_Pos: 120  
Relay_Log_Space: 607  
Until_Condition: None  
Until_Log_File:  
Until_Log_Pos: 0  
Master_SSL_Allowed: No  
Master_SSL_CA_File:  
Master_SSL_CA_Path:  
Master_SSL_Cert:  
Master_SSL_Cipher:  
Master_SSL_Key:  
Seconds_Behind_Master: 0  
Master_SSL_Verify_Server_Cert: No  
Last_IO_Errno: 0  
Last_IO_Error:  
Last_SQL_Errno: 0  
Last_SQL_Error:  
Replicate_Ignore_Server_Ids:  
Master_Server_Id: 1  
Master_UUID: 2667fd34-0483-11e7-b59d-525400a0f91d  
Master_Info_File: /svc/db_data/master.info  
SQL_Delay: 0  
SQL_Remaining_Delay: NULL  
Slave_SQL_Running_State: Slave has read all relay log; waiting for the slave I/O thread to update it  
Master_Retry_Count: 86400  
Master_Bind:  
Last_IO_Error_Timestamp:  
Last_SQL_Error_Timestamp:  
Master_SSL_Crl:  
Master_SSL_Crlpath:  
Retrieved_Gtid_Set:  
Executed_Gtid_Set:  
Auto_Position: 0  
1 row in set (0.00 sec)  
  
ERROR:  
No query specified
```

# Mysql-MMM 설치하기

- 1. Mysql-MMM구성도
- 2. MMM(Multi-Master Replication Manager for MySQL)은 리플리케이션으로 DB데이터를 이중화했을때, 각 DBMS별로 상태 확인하여 ReadWrite, Readonly VIP를 관리하기 위한 오픈소스
- 3. 서버별 설치 Component

대상서버	Component	역 할
svcdb/dashboard_db	Mysql, mmm-agent	Mysql, 데이터 동기화, mmm 상태확인
db monitor	mmm-monitor	각 Agent별 Healthcheck, VIP 관리

- 4. MMM접근 설정 (DB서버)

```
mysql> grant process,super,replication client on . to 'agent'@'자신IP' identified by 'agent';
Query OK, 0 rows affected (0.00 sec)

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

- 5. MMM-agent 설치 (DB서버)모니터 접근 허용하기 (DB서버)

```
mysql> GRANT REPLICATION CLIENT ON . TO 'monitor'@'MONITOR서버IP' IDENTIFIED BY 'monitor';
1 row in set (0.00 sec)

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

## MMM agent 설치

- 1. 설치에 필요한 패키지
  - 1. mysql-mmm-2.2.1-2.el6.noarch.rpm
  - 2. mysql-mmm-agent-2.2.1-2.el6.noarch.rpm
  - 3. mysql-mmm-tools-2.2.1-2.el6.noarch.rpmOS환경에 따라 종속된 패키지가 필요할 수 있음. 종속된 패키지는 perl관련 패키지가 설치되어 있어야 함.
- 2. mmm-agent설치

```
[root@localhost ~]# yum install -y mysql-mmm-2.2.1-2.el6.noarch.rpm
[root@localhost ~]# yum install -y mysql-mmm-agent-2.2.1-2.el6.noarch.rpm
[root@localhost ~]# yum install -y mysql-mmm-tools-2.2.1-2.el6.noarch.rpm
```

- 3. mmm-common.conf 설정

```
$> vi /etc/mysql-mmm/mmm_common.conf
active_master_role writer

<host default>
cluster_interface bond0 #서버NIC 인터페이스 기재
pid_path /var/run/mysql-mmm/mmm_agentd.pid
bin_path /usr/libexec/mysql-mmm/
replication_user repl #리플리케이션 사용자 입력
replication_password repl #리플리케이션 사용자 패스워드
agent_user agent #mmm을 통해 DBMS로 접근할 사용자 입력
agent_password agentpass #패스워드 입력
</host>

<host db1>
ip Mater 서버IP
mode master
peer db2
</host>

<host db2>
ip Standby 서버IP
mode master
peer db1
</host>
```

```
<role writer>
hosts db1, db2
ips DB VIP
mode exclusive
</role>
```

#### 4. mmm - mmm\_agent.conf 설정

```
$> vi /etc/mysql-mmm/mmm_agent.conf
include mmm_common.conf

# The 'this' variable refers to this server. Proper operation requires
# that 'this' server (db1 by default), as well as all other servers, have the
# proper IP addresses set in mmm_common.conf.

this db1 # Active는 db1, Standby는 db2로 기재
```

#### 5. mmm-agent 서비스 실행 후 프로세스 상태 확인

```
[root@localhost ~]# /etc/init.d/mysql-mmm-agent start
Starting mmm_agentd: [ OK ]

[root@localhost ~]# ps -ef | grep mmm
root 8052 8031 0 10:12 pts/0 00:00:00 grep mmm
root 18251 1 0 2016 ? 00:00:00 mmm_agentd
root 18253 18251 0 2016 ? 04:03:18 mmm_agentd
```

## MMM-monitor 설치 (Statistics#2 서버)

#### 1. 설치에 필요한 패키지

1. mysql-mmm-2.2.1-2.el6.noarch.rpm
2. mysql-mmm-monitor-2.2.1-2.el6.noarch.rpm
3. mysql-mmm-tools-2.2.1-2.el6.noarch.rpm

OS환경에 따라 종속된 패키지가 필요할 수 있음. 종속된 패키지는 perl관련 패키지가 설치되어 있어야 함.

#### 2. svcd용 MMM 설정

```
$> vi /etc/mysql-mmm/mmm_common_svcd.conf

active_master_role writer

<host default>
cluster_interface bond0 □ 서버NIC 인터페이스 기재
pid_path /var/run/mysql-mmm/mmm_agentd_svcd.pid
bin_path /usr/libexec/mysql-mmm/
replication_user repl □ 리플리케이션 사용자 입력
replication_password repl □ 리플리케이션 사용자 패스워드
agent_user agent □ mmm을 통해 DBMS로 접근할 사용자 입력
agent_password agentpass □ 패스워드 입력
</host>

<host db1>
ip svcd Master 서버IP
mode master
peer db2
</host>

<host db2>
ip svcd Standby 서버IP
mode master
peer db1
</host>

<role writer>
hosts db1, db2
ips svcd VIP
mode exclusive
</role>
```

```
$> vi /etc/mysql-mmm/mmm_mon_svcd.db.conf

include mmm_common_svcd.db.conf

<monitor>
ip 127.0.0.1
pid_path /var/run/mysql-mmm/mmm_mond_svcd.db.pid
bin_path /usr/libexec/mysql-mmm
status_path /var/lib/mysql-mmm/mmm_mond_svcd.db.status
ping_ips svcd.db RIP □ IP는 콤마(,)로 구분
auto_set_online 60
port 9991
</monitor>

<host default>
monitor_user mmm_monitor □ mmm 접근 사용자
monitor_password monitorpw □ mmm 패스워드정보
</host>
debug 0

$> vi /etc/mysql-mmm/mmm_mon_log_svcd.db.conf
log4perl.logger = INFO, MMMLog

log4perl.appender.MMMLog = Log::Log4perl::Appender::File
log4perl.appender.MMMLog.Threshold = INFO
log4perl.appender.MMMLog.filename = /var/log/mysql-mmm/mmm_mond_svcd.db.log
log4perl.appender.MMMLog.recreate = 1
log4perl.appender.MMMLog.layout = PatternLayout
log4perl.appender.MMMLog.layout.ConversionPattern = %d %5p %m%n
```

### 3. 서비스 등록

#### 1. init.d스크립트 복사

```
[root@localhost ~]# cp /etc/init.d/mysql-mmm-monitor /etc/init.d/mysql-mmm-svcd.db
```

#### 2. 스크립트 수정

```
$> vi /etc/init.d/mysql-mmm-svcd.db
#!/bin/sh
#
# mysql-mmm-monitor This shell script takes care of starting and stopping
# the mmm monitoring daemon.
#
# chkconfig: - 64 36
# description: MMM Monitor.
# processname: mmm_mond
# config: /etc/mysql-mmm/mmm_mon.conf
# pidfile: /var/run/mysql-mmm/mmm_mond.pid

# Source function library and defaults file.
. /etc/rc.d/init.d/functions
. /etc/default/mysql-mmm-monitor

# Cluster name (it can be empty for default cases)
CLUSTER='svcd.db'
LOCKFILE='/var/lock/subsys/mysql-mmm-monitor'
prog='MMM Monitor Daemon'

if [ "$CLUSTER" != "" ]; then
    MMMD_MON_BIN="/usr/sbin/mmm_mond @$CLUSTER"
    MMMD_MON_PIDFILE="/var/run/mysql-mmm/mmm_mond_$CLUSTER.pid"
else
    MMMD_MON_BIN="/usr/sbin/mmm_mond"
    MMMD_MON_PIDFILE="/var/run/mysql-mmm/mmm_mond.pid"
```

#### 4. dashboard\_db MMM설정

```
$> vi /etc/mysql-mmm/mmm_common_dashboard_db.conf
active_master_role writer

<host default>
cluster_interface bond0 □ 서버NIC 인터페이스 기재
pid_path /var/run/mysql-mmm/mmm_agentd-cssatdb.pid
bin_path /usr/libexec/mysql-mmm/
replication_user repl □ 리플리케이션 사용자 입력
replication_password repl □ 리플리케이션 사용자 패스워드
agent_user agent □ mmm을 통해 DBMS로 접근할 사용자 입력
agent_password agentpass □ 패스워드 입력
</host>

<host db1>
ip dashboard_db Mater 서버IP
mode master
peer db2
</host>

<host db2>
ip dashboard_db Standby 서버IP
mode master
peer db1
</host>

<role writer>
hosts db1, db2
ips dashboard_db VIP
mode exclusive
</role>

$> vi /etc/mysql-mmm/mmm_mon_dashboard_db.conf
include mmm_common_dashboard_db.conf

<monitor>
ip 127.0.0.1
pid_path /var/run/mysql-mmm/mmm_mond_dashboard_db.pid
bin_path /usr/libexec/mysql-mmm
status_path /var/lib/mysql-mmm/mmm_mond_dashboard_db.status
ping_ips dashboard_db RIP입력 □ IP는 콤마(,)로 구분
auto_set_online 60
port 9990
</monitor>

<host default>
monitor_user mmm_monitor □ mmm 접근 사용자
monitor_password monitorpw □ mmm 패스워드정보
</host>

debug 0

$> vi /etc/mysql-mmm/mmm_mon_log_dashboard_db.conf
log4perl.logger = INFO, MMMLog

log4perl.appender.MMMLog = Log::Log4perl::Appender::File
log4perl.appender.MMMLog.Threshold = INFO
log4perl.appender.MMMLog.filename = /var/log/mysql-mmm/mmm_mond_dashboard_db.log
log4perl.appender.MMMLog.recreate = 1
log4perl.appender.MMMLog.layout = PatternLayout
log4perl.appender.MMMLog.layout.ConversionPattern = %d %5p %m%n
```

#### 1. 서비스 등록

## 2. init.d스크립트 복사

```
[root@localhost ~]# cp /etc/init.d/mysql-mmm-monitor /etc/init.d/mysql-mmm-dashboard_db
```

## 5. init.d스크립트 내용변경

```
%> vi /etc/init.d/mysql-mmm-dashboard_db

#!/bin/sh

mysql-mmm-monitor This shell script takes care of starting and stopping

the mmm monitoring daemon.

chkconfig: - 64 36

description: MMM Monitor.

processname: mmm_mond

config: /etc/mysql-mmm/mmm_mon.conf

pidfile: /var/run/mysql-mmm/mmm_mond.pid

Source function library and defaults file.
. /etc/rc.d/init.d/functions
. /etc/default/mysql-mmm-monitor

Cluster name (it can be empty for default cases)
CLUSTER='dashboard_db'
LOCKFILE="/var/lock/subsys/mysql-mmm-monitor"
prog='MMM Monitor Daemon'

if [ "$CLUSTER" != "" ]; then
MMMD_MON_BIN="/usr/sbin/mmm_mond @$CLUSTER"
MMMD_MON_PIDFILE="/var/run/mysql-mmm/mmm_mond_${CLUSTER}.pid"
else
MMMD_MON_BIN="/usr/sbin/mmm_mond"
MMMD_MON_PIDFILE="/var/run/mysql-mmm/mmm_mond.pid"
fi
```

## 6. Centos7 환경에서MMM-monitor등록

## 7. systemctl 서비스 생성 및 등록

```
[root@localhost ~]# cd /usr/lib/systemd/system
[root@localhost system]# cp mysql-mmm-monitor.service mysql-mmm-monitor_svcdb.service
[root@localhost system]# cp mysql-mmm-monitor.service mysql-mmm-monitor_dashboard_db.service
[root@localhost system]# systemctl enable mysql-mmm-monitor_svcdb.service
[root@localhost system]# systemctl enable mysql-mmm-monitor_dashboard_db.service

$> vi /usr/lib/systemd/system/ mysql-mmm-monitor_svcdb.service

[Unit]
Description=MySQL MMM Monitor
After=network.target mysqld.service

[Service]
Type=forking
PIDFile=/var/run/mmm_mond_svcdb.pid
ExecStart=/etc/init.d/mysql-mmm-monitor-svcdb start
ExecStop=/etc/init.d/mysql-mmm-monitor-svcdb stop
```

```
PrivateTmp=true
```

```
[Install]
```

```
WantedBy=multi-user.target
```

```
$> vi /usr/lib/systemd/system/ mysql-mmm-monitor_dashboard_db.service
```

```
[Unit]
```

```
Description=MySQL MMM Monitor
```

```
After=network.target mysqld.service
```

```
[Service]
```

```
Type=forking
```

```
PIDFile=/var/run/mmm_mond_svcd.db.pid
```

```
ExecStart=/etc/init.d/mysql-mmm-monitor-dashboard_db start
```

```
ExecStop=/etc/init.d/mysql-mmm-monitor-dashboard_db stop
```

```
PrivateTmp=true
```

```
[Install]
```

```
WantedBy=multi-user.target
```

## 8. 서비스 실행

### 1. MMM-monitor 서비스 실행 (monitor서버에서 실행)

```
[root@localhost ~]# /etc/init.d/mysql-mmm-svcd.db start
Starting mmm_mond: [ OK ]
[root@localhost ~]# /etc/init.d/mysql-mmm-dashboard_db start
Starting mmm_mond: [ OK ]
```

### 2. Mysql 프로세스 실행 상태 확인서비스 상태 확인 (db서버에서 실행)

```
[root@localhost ~]# ps -ef | grep mysql
root 6714 1 0 Jan08 ? 00:00:00 /bin/sh /usr/local/mysql/bin/mysqld_safe --datadir=/home/mysqldata --pid-file=/home/mysqldata/localhost.pid
mysql 7856 6714 0 Jan08 ? 00:26:50 /usr/local/mysql/bin/mysqld --basedir=/usr/local/mysql --datadir=/home/mysqldata --plugin-
dir=/usr/local/mysql/lib/plugin --user=mysql --log-error=/home/mysqldata/error.log --open-files-limit=8192 --pid-
file=/home/mysqldata/localhost.pid --socket=/tmp/mysql.sock --port=3306
```

### 3. Mysql 쿼리 처리여부 확인

```
[root@localhost ~]# mysql -u root -p'root패스워드' -e'select now()'
2017-03-12 19:06:14
```

### 4. Mmm 프로세스상태 확인

```
[root@localhost ~]# ps -ef | grep mmm
root 10832 1 0 Mar07 ? 00:00:00 mmm_mond-dashboard_db
root 10834 10832 0 Mar07 ? 00:04:27 mmm_mond-dashboard_db
root 10843 10834 0 Mar07 ? 00:01:49 perl /usr/libexec/mysql-mmm/monitor/checker @dashboard_db ping_ip
root 10846 10834 0 Mar07 ? 00:01:03 perl /usr/libexec/mysql-mmm/monitor/checker @dashboard_db mysql
root 10848 10834 0 Mar07 ? 00:00:35 perl /usr/libexec/mysql-mmm/monitor/checker @dashboard_db ping
root 10850 10834 0 Mar07 ? 00:01:29 perl /usr/libexec/mysql-mmm/monitor/checker @dashboard_db rep_backlog
root 10852 10834 0 Mar07 ? 00:01:28 perl /usr/libexec/mysql-mmm/monitor/checker @dashboard_db rep_threads
root 10960 1 0 Mar07 ? 00:00:00 mmm_mond-svcd.db
root 10962 10960 0 Mar07 ? 00:04:23 mmm_mond-svcd.db
root 10969 10962 0 Mar07 ? 00:01:47 perl /usr/libexec/mysql-mmm/monitor/checker @svcd.db ping_ip
root 10972 10962 0 Mar07 ? 00:01:03 perl /usr/libexec/mysql-mmm/monitor/checker @svcd.db mysql
root 10974 10962 0 Mar07 ? 00:00:35 perl /usr/libexec/mysql-mmm/monitor/checker @svcd.db ping
root 10976 10962 0 Mar07 ? 00:01:27 perl /usr/libexec/mysql-mmm/monitor/checker @svcd.db rep_backlog
root 10978 10962 0 Mar07 ? 00:01:27 perl /usr/libexec/mysql-mmm/monitor/checker @svcd.db rep_threads
root 29560 27996 0 21:40 pts/3 00:00:00 grep mmm
```

### 5. MMM 이중화 상태 확인 (dashboard\_db)

```
[root@localhost ~]# mmm_control @dashboard_db show
db1(192.168.67.230) master/ONLINE. Roles: writer(192.168.67.229)
db2(192.168.67.231) master/ONLINE. Roles:
```

### 6. MMM 이중화 상태 확인 (svcd.db)

```
[root@localhost ~]# mmm_control @svcd.db show
```



```
db1(192.168.67.224) master/ONLINE. Roles: writer(192.168.67.223)
db2(192.168.67.225) master/ONLINE. Roles:
```

## 7. DBMS별 상태 확인 (svcdcb)

```
[root@localhost ~]# mmm_control @svcdcb checks all
db2 ping [last change: 2017/03/09 11:03:42] OK
db2 mysql [last change: 2017/03/09 11:03:42] OK
db2 rep_threads [last change: 2017/03/09 11:03:42] OK
db2 rep_backlog [last change: 2017/03/09 06:02:35] OK: Backlog is null
db1 ping [last change: 2017/03/09 11:03:42] OK
db1 mysql [last change: 2017/03/09 11:03:42] OK
db1 rep_threads [last change: 2017/03/09 11:03:42] OK
db1 rep_backlog [last change: 2017/03/09 11:03:42] OK: Backlog is null
```

## 8. DBMS별 상태 확인 (dashboard\_db)

```
[root@localhost ~]# mmm_control @dashboard_db checks all
db2 ping [last change: 2017/03/09 10:58:37] OK
db2 mysql [last change: 2017/03/09 10:58:37] OK
db2 rep_threads [last change: 2017/03/09 10:58:37] OK
db2 rep_backlog [last change: 2017/03/09 10:58:37] OK: Backlog is null
db1 ping [last change: 2017/03/09 10:58:37] OK
db1 mysql [last change: 2017/03/09 10:58:37] OK
db1 rep_threads [last change: 2017/03/09 10:58:37] OK
db1 rep_backlog [last change: 2017/03/09 10:58:37] OK: Backlog is null
```

# FAQ

## 1. 설치 과정의 오류

현상	rpm설치시 아래내용으로 메시지 출력후 설치 실패 [root@localhost ~]# rpm -Uvh 123.rpm warning: 123.rpm: Header V3 RSA/SHA256 Signature, key ID 0608b895: NOKEY error: Failed dependencies: perl(Log::Log4perl) is needed by mysql-mmm-2.2.1-2.el6.noarch
원인	MMM 설치시 종속된 패키지 설치 실패
확인사항	perl-Proc-Daemon, perl-Proc-ProcessTable, perl-Log, perl-Mail 패키지 설치여부 확인
조치방안	관련된 전체 패키지 내려받은 후 설치작업 수행

## 2. MMM-monitor 서비스 구동 실패현상

현상	설치 및 설정완료후 서비스 구동시 구동불가 [root@localhost ~]# /etc/init.d/mysql-mmm-dashboard_db-monitor start Starting MMM Monitor Daemon: Configuration file /etc/mysql-mmm/mmm_mon_dashboard_db.conf is world readable! at /usr/share/perl5/vendor_perl/MMM/Common/Config.pm line 132 [실패]
원인	Mmm config의 권한 설정의 오류
확인사항	/etc/mysql-mmm/ 경로에 있는 config파일의 퍼미션 확인
조치방안	전체 퍼미션을 640으로 설정 수행 Chmod 640 /etc/mysql-mmm/*.conf

## 3. VIP할당 실패

현상	Mmm 서비스 구동 후 IP할당불가
원인	monitor에서 mysql로 접근하는 권한이 오류가 있어 mysql상태체크 불가
확인사항	/var/log/mysql-mmm/mmm-agent.log에서 Can't connect to MySQL sever 문자열 존재여부 확인
조치방안	Mysql 접근권한 설정

4. MMM 리소스 확인시 중복출력

현상	mmm_config를 통해 DBMS별 상태 확인 불가
원인	Group별 생성되는 포트를 생성하지 않는 경우 발생
확인 사항	/etc/mysql-mmm/mmm_mon_svldb.conf, /etc/mysql-mmm/mmm_mon_svldb.conf 파일안에 port항목이 기재되어 있고, 각각 다른 포트를 사용하는지 확인
조치 방안	dbms별 포트 설정

5. Centos7에서 mmm\_monitor cluster 구성실패

현상	Systemctl을 이용한 mysql-monitor 서비스 작동 실패
원인	Systemctl을 이용한 mmm-monitor실행은 cluster모드를 지원하지 않음
확인사항	Systemctl 구문안에 ExecStart / ExecStop 구문의 명령어 확인 /etc/init.d/mysql-mmm-monitor- 파일이 있는지 확인
조치방안	설정파일 및 서비스 구동 명령어 확인