NSD DBA2 DAY03

1 案例1：准备MHA集群环境

1.1 问题

- 准备6台虚拟机，并按照本节规划配置好IP参数
- 在这些虚拟机之间实现SSH免密登录
- 在相应节点上安装好MHA相关的软件包

1.2 方案

使用6台RHEL 7虚拟机，如图-1所示。准备集群环境，安装依赖包，授权用户，配置ssh密钥对认证登陆，所有节点之间互相以root秘钥对认证登录，管理主机以root密钥对认证登录所有数据节点主机，配置mha集群。

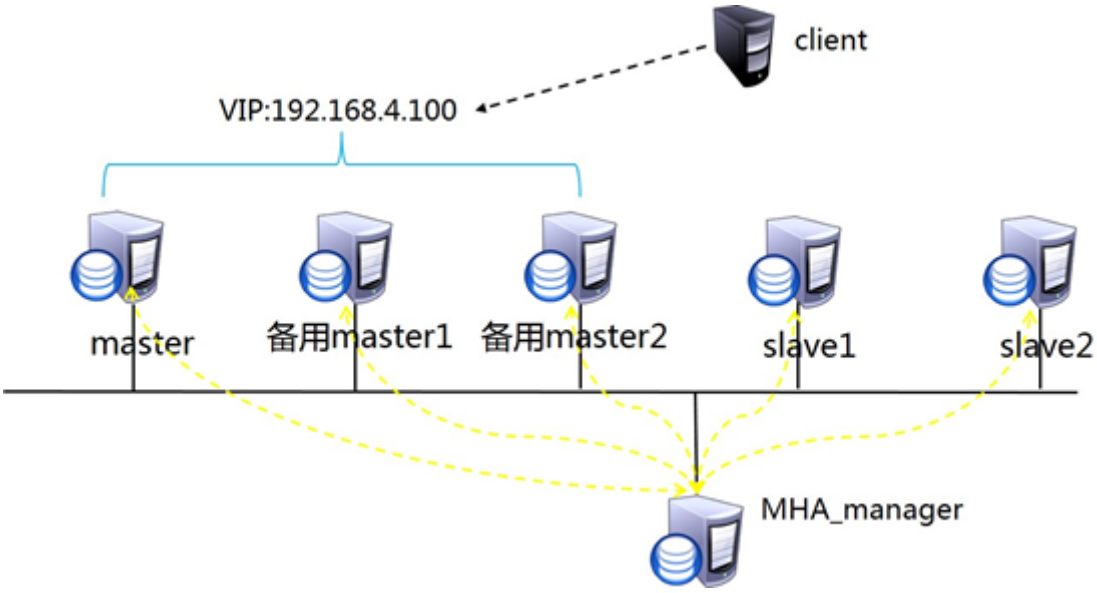

图 - 1

IP规划，如图-2所示：

Top

角色	IP地址	主机名
Master主节点服务器	192.168.4.51	master51
备用1主节点服务器	192.168.4.52	master52
备用2主节点服务器	192.168.4.53	master53
第1台 slave服务器	192.168.4.54	slave54
第2台 slave服务器	192.168.4.55	slave55
MHA_manager服务器	192.168.4.56	mgm56
VIP地址	192.168.4.100	

图-2

1.3

1.4 步骤

实现此案例需要按照如下步骤进行。

步骤一： 准备集群环境

1）修改主机名，配置IP（其余几台请按照图-2修改IP和主机名，这里以master51为例）

```
[root@localhost ~]# echo master51 > /etc/hostname
[root@localhost ~]# nmcli connection modify eth0 ipv4.method manual ipv4.addresses 1
[root@localhost ~]# nmcli connection up eth0
```

2在所有主机上安装Perl依赖包（51-56操作）

```
[root@master51 ~]# cd mysql/mha-soft-student/
[root@master51 ~]# yum -y install perl-*.rpm
```

3）在所有数据库服务器上安装mha-node包（51-55操作）

```
[root@master51 mha-soft-student]# yum -y install perl-DBD-mysql perl-DBI
 [root@master51 mha-soft-student]# rpm -ivh mha4mysql-node-0.56-0.el6.noarch.rpm
Preparing...                ########################################### [100%]
Updating / installing...                                                Top
   1:mha4mysql-node-0.56-0.el6      ########################################### [100%]
```

4）在管理主机上安装mha_node 和 mha-manager包（56操作）

```
[root@mgm56 mha-soft-student]# yum -y  install perl-DBD-mysql  perl-DBI
[root@mgm56 mha-soft-student]# rpm -ivh  mha4mysql-node-0.56-0.el6.noarch.rpm
Preparing...                ########################################### [100%]
Updating / installing...
   1:mha4mysql-node-0.56-0.el6        ########################################### [100%]
[root@mgm56 mha-soft-student]# yum -y  install perl-ExtUtils-*  perl-CPAN-*
[root@mgm56 mha-soft-student]# tar -zxf mha4mysql-manager-0.56.tar.gz
[root@mgm56 mha-soft-student]# cd mha4mysql-manager-0.56/
[root@mgm56 mha4mysql-manager-0.56]# perl Makefile.PL
*** Module::AutoInstall version 1.03
*** Checking for Perl dependencies...
[Core Features]
- DBI                ...loaded. (1.627)
- DBD::mysql         ...loaded. (4.023)
- Time::HiRes        ...loaded. (1.9725)
- Config::Tiny       ...loaded. (2.14)
- Log::Dispatch      ...loaded. (2.41)
- Parallel::ForkManager ...loaded. (1.18)
- MHA::NodeConst     ...loaded. (0.56)
*** Module::AutoInstall configuration finished.   //配置完成
Checking if your kit is complete...
Looks good
Writing Makefile for mha4mysql::manager
Writing MYMETA.yml and MYMETA.json
[root@mgm56 mha4mysql-manager-0.56]# make
[root@mgm56 mha4mysql-manager-0.56]# make install
```

步骤二： 配置ssh密钥对认证登陆

1）所有节点之间可以互相以ssh密钥对方式认证登陆以（以51为例）

```
[root@master51 mha-soft-student]# ssh-keygen
[root@master51 mha-soft-student]# ssh-copy-id 192.168.4.52
//除了传给52外，53，54，55也要传，52-55主机也是一样的
```

6）配置56主机 无密码ssh登录所有数据节点主机

Top

```
[root@mgm56 mha4mysql-manager-0.56]# ssh-keygen
[root@mgm56 mha4mysql-manager-0.56]# ssh-copy-id 192.168.4.51
//除传给51外，还要传给52-55
```

2 案例2：配置MHA集群环境

2.1 问题

- 配置主节点 master51
- 配置两个备用主节点 master52、master53
- 配置两个从节点 slave54、slave55
- 配置管理节点 mgm56

1.

2.2 步骤

实现此案例需要按照如下步骤进行。

步骤一：配置mha集群环境

1）安装数据库（51-55同样操作，以51为例）

```
[root@master51 ~]# cd /root/mysql
[root@master51 mysql]# tar -xf mysql-5.7.17.tar
[root@master51 mysql]# yum -y install perl-JSON
[root@master51 mysql]# rpm -Uvh mysql-community-*.rpm
[root@master51 mysql]# rpm -qa | grep -i mysql
```

2）master51 数据库服务器配置文件

```
[root@master51 mysql]# vim /etc/my.cnf
plugin-load="rpl_semi_sync_master=semisync_master.so;rpl_semi_sync_slave=semisync_
rpl-semi-sync-master-enabled=1
rpl-semi-sync-slave-enabled=1
server_id=51
log-bin=master51
binlog-format="mixed"

[root@master51 mysql]# systemctl restart mysqld

[root@master51 mysql]# mysql -u root -p123456
```

Top

```
12.
13.  mysql> set global relay_log_purge=off; //不自动删除本机的中继日志文件
14.  Query OK, 0 rows affected (0.00 sec)
15.
16.  mysql> grant replication slave on *.* to repluser@"%" identified by "123456";
17.  //添加主从同步授权用户
18.  Query OK, 0 rows affected, 1 warning (10.01 sec)
19.
20.  mysql> show master status;
21.  +----------------+----------+--------------+------------------+-------------------
22.  | File           | Position | Binlog_Do_DB | Binlog_Ignore_DB | Executed_Gtid_Set |
23.  +----------------+----------+--------------+------------------+-------------------
24.  | master51.000003 |      441 |              |                  |                   |
25.  +----------------+----------+--------------+------------------+-------------------
26.  1 row in set (0.00 sec)
```

3）master52数据库服务器配置文件

```
01.  [root@master52 mysql]# vim /etc/my.cnf
02.  plugin-load="rpl_semi_sync_master=semisync_master.so;rpl_semi_sync_slave=semisync_s
03.  rpl-semi-sync-master-enabled=1
04.  rpl-semi-sync-slave-enabled=1
05.  server_id=52
06.  log-bin=master52
07.  binlog-format="mixed"
08.
09.  [root@master52 mysql]# systemctl restart mysqld
10.  [root@master52 mysql]# mysql -u root -p123456
11.  mysql> set global relay_log_purge=off;
12.  mysql> change master to
13.      -> master_host="192.168.4.51",
14.      -> master_user="repluser",
15.      -> master_password="123456",
16.      -> master_log_file="master51.000003",
17.      -> master_log_pos=441;
18.  Query OK, 0 rows affected, 2 warnings (0.01 sec)
19.  mysql> start slave;
20.  Query OK, 0 rows affected (0.01 sec)
21.  mysql> show slave status\G;
```

Top

```
22.  ...
23.               Slave_IO_Running: Yes
24.              Slave_SQL_Running: Yes
25.  ...
```

4）master53数据库服务器配置文件

```
01.  [root@master53 mysql]# vim /etc/my.cnf
02.  plugin-load="rpl_semi_sync_master=semisync_master.so;rpl_semi_sync_slave=semisync_s
03.  rpl-semi-sync-master-enabled=1
04.  rpl-semi-sync-slave-enabled=1
05.  server_id=53
06.  log-bin=master53
07.  binlog-format="mixed"
08.
09.  [root@master53 mysql]# systemctl restart mysqld
10.  [root@master53 mysql]# mysql -u root -p123456
11.  mysql> set global relay_log_purge=off;
12.  Query OK, 0 rows affected (0.00 sec)
13.
14.  mysql> change master to
15.      -> master_host="192.168.4.51",
16.      -> master_user="repluser",
17.      -> master_password="123456",
18.      -> master_log_file="master51.000003",
19.      -> master_log_pos=441;
20.  Query OK, 0 rows affected, 2 warnings (0.01 sec)
21.  mysql> start slave;
22.  Query OK, 0 rows affected (0.00 sec)
23.  mysql> show slave status\G;
24.  ...
25.               Slave_IO_Running: Yes
26.              Slave_SQL_Running: Yes
27.  ...
```

5）slave54 数据库服务器配置文件

Top

```
01.  [root@slave54 mysql]# vim /etc/my.cnf
```

```
server_id=54
[root@master54 mysql]# systemctl restart mysqld
[root@master54 mysql]# mysql -u root -p123456
mysql> change master to
    -> master_host="192.168.4.51",
    -> master_user="repluser",
    -> master_password="123456",
    -> master_log_file="master51.000003",
    -> master_log_pos=441;
Query OK, 0 rows affected, 2 warnings (0.01 sec)
mysql> start slave;
Query OK, 0 rows affected (0.00 sec)
mysql> show slave status\G;
...
             Slave_IO_Running: Yes
            Slave_SQL_Running: Yes
...
```

6）slave55 数据库服务器配置文件

```
[root@slave55 mysql]# vim /etc/my.cnf
server_id=55

[root@master55 mysql]# systemctl restart mysqld
[root@master55 mysql]# mysql -u root -p123456
mysql> change master to
    -> master_host="192.168.4.51",
    -> master_user="repluser",
    -> master_password="123456",
    -> master_log_file="master51.000003",
    -> master_log_pos=441;
Query OK, 0 rows affected, 2 warnings (0.01 sec)
mysql> start slave;
Query OK, 0 rows affected (0.00 sec)
mysql> show slave status\G;
...
             Slave_IO_Running: Yes
            Slave_SQL_Running: Yes
...
```

Top

7）配置管理主机4.56

```
[root@mgm56 ~]# cd mysql/mha-soft-student/mha4mysql-manager-0.56/
[root@mgm56 mha4mysql-manager-0.56]#  cp bin/* /usr/local/bin/
//提示覆盖，说明安装的时候有，没有可以拷贝过来
[root@mgm56 mha4mysql-manager-0.56]# mkdir /etc/mha_manager    //创建工作目录
[root@mgm56 mha4mysql-manager-0.56]#  cp samples/conf/app1.cnf  /etc/mha_manag
//建立样板文件
[root@mgm56 mha4mysql-manager-0.56]# vim /etc/mha_manager/app1.cnf
//编辑主配置文件app1.cnf
[server default]
manager_workdir=/etc/mha_manager
manager_log=/etc/mha_manager/manager.log
master_ip_failover_script=/usr/local/bin/master_ip_failover

ssh_user=root
ssh_port=22
repl_user=repluser
repl_password=123456
user=root
password=123456

[server1]
hostname=192.168.4.51
port=3306

[server2]
hostname=192.168.4.52
port=3306
candidate_master=1

[server3]
hostname=192.168.4.53
port=3306
candidate_master=1

[server4]
hostname=192.168.4.54
no_master=1

[server5]
```

Top

```
hostname=192.168.4.55
no_master=1
[root@mgm56 mha4mysql-manager-0.56]# cp samples/scripts/master_ip_failover
/usr/local/bin/       //创建故障切换的脚本
```

3 案例3：测试MHA集群

3.1 问题

- 查看MHA集群状态
- 测试节点之间的SSH登录
- 测试集群VIP的故障切换功能

3.2 步骤

实现此案例需要按照如下步骤进行。

步骤一：验证配置

1）检查配置环境，在主机52，53检查是否有同步数据的用户repluser

主机52:

```
mysql> select user,host from mysql.user where user="repluser";
+----------+------+
| user     | host |
+----------+------+
| repluser | %    |
+----------+------+
1 row in set (0.00 sec)

mysql> show grants for repluser@"%";
+-------------------------------------------------+
| Grants for repluser@%                           |
+-------------------------------------------------+
| GRANT REPLICATION SLAVE ON *.* TO 'repluser'@'%' |
+-------------------------------------------------+
1 row in set (0.00 sec
```

主机53:

Top

```
mysql> select user,host from mysql.user where user="repluser";
+----------+------+
```

```
| user       | host |
+------------+------+
| repluser   | %    |
+------------+------+
1 row in set (0.00 sec)

mysql> show grants for repluser@"%";
+-------------------------------------------------------+
| Grants for repluser@%                                 |
+-------------------------------------------------------+
| GRANT REPLICATION SLAVE ON *.* TO 'repluser'@'%'      |
+-------------------------------------------------------+
1 row in set (0.00 sec)
```

2）在51的主机上做root的授权，其他的会同步（如果不做，在验证数据节点的主从同步配置时会出错）

```
mysql> grant all on *.* to root@"%" identified by "123456";
mysql> select user,host from mysql.user where user="root";
+------+-----------+
| user | host      |
+------+-----------+
| root | %         |
| root | localhost |
+------+-----------+
2 rows in set (0.00 sec)
```

3）验证ssh 免密登陆数据节点主机

```
[root@mgm56 mha4mysql-manager-0.56]# cd /usr/local/bin/
[root@mgm56 bin]# masterha_check_ssh --conf=/etc/mha_manager/app1.cnf
Wed Sep 19 09:09:33 2018 - [info] All SSH connection tests passed successfully.
//出现这个为成功
```

Top

4）验证数据节点的主从同步配置（先把自动failover时候的切换脚本注释掉）

```
[root@mgm56 bin]# masterha_check_repl --conf=/etc/mha_manager/app1.cnf
MySQL Replication Health is OK.  //验证成功
```

5）启动管理服务MHA_Manager

--remove_dead_master_conf //删除宕机主库配置

--ignore_last_failover //忽略xxx.health文件

```
[root@mgm56 bin]# masterha_manager --conf=/etc/mha_manager/app1.cnf \
--remove_dead_master_conf --ignore_last_failover

Wed Sep 19 09:24:41 2018 - [warning] Global configuration file /etc/masterha_default.cn
Wed Sep 19 09:24:41 2018 - [info] Reading application default configuration from /etc/mh
Wed Sep 19 09:24:41 2018 - [info] Reading server configuration from /etc/mha_manager,
```

6）查看状态（另开一个终端）

```
[root@mgm56 ~]# masterha_check_status --conf=/etc/mha_manager/app1.cnf
app1 (pid:15745) is running(0:PING_OK), master:192.168.4.51
```

7）停止服务

```
[root@mgm56 ~]# masterha_stop --conf=/etc/mha_manager/app1.cnf
Stopped app1 successfully.
```

步骤二：测试故障转移

1）在主库51上面配置VIP地址

```
[root@master51 ~]# ifconfig eth0:1 192.168.4.100/24
```

2）在配置文件里面把自动failover时候的切换脚本去掉注释

3）修改 master_ip_failover 脚本，设置如下内容

Top

```
34 my $vip = '192.168.4.100/24';
```

```
02.  35 my $key = "1";
03.  36 my $ssh_start_vip = "/sbin/ifconfig eth0:$key $vip";
04.  37 my $ssh_stop_vip = "/sbin/ifconfig eth0:$key down";
```

4）启动服务

```
01.  [root@mgm56 bin]# masterha_manager --conf=/etc/mha_manager/app1.cnf \
02.  --remove_dead_master_conf --ignore_last_failover
03.  Wed Sep 19 09:50:33 2018 - [warning] Global configuration file /etc/masterha_default.cn
04.  Wed Sep 19 09:50:33 2018 - [info] Reading application default configuration from /etc/ml
05.  Wed Sep 19 09:50:33 2018 - [info] Reading server configuration from /etc/mha_manager
```

5）查看状态

```
01.  [root@mgm56 ~]# masterha_check_status --conf=/etc/mha_manager/app1.cnf
02.  app1 master is down and failover is running(50:FAILOVER_RUNNING). master:192.168.4.52
```

验证数据节点的主从同步配置报错，如图-3所示：

```
01.  [root@mgm56 bin]# masterha_check_repl --conf=/etc/mha_manager/app1.cnf
```

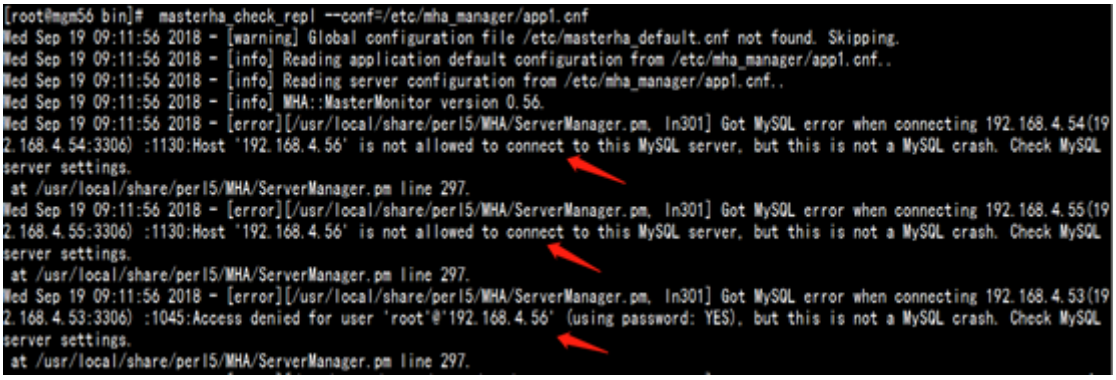

图-3

解决办法：

root用户没有授权，默认只能本地连接，在主机51上面授权root用户可以远程登录，其他主机会同步

Top

```
01.  mysql> grant all on *.* to root@"%" identified by "123456";
```

Top