

鼎甲迪备

备份服务端高可用

最佳实践用户指南

Release V8.0-9

October, 2024



目录

1	环境准备	1
1.1	修改主机名, host 文件解析	1
1.2	配置免密	1
1.3	时间同步	1
1.4	修改 shell 编译器	1
1.5	系统日志 (可选)	2
2	备份系统安装	3
3	Catalog 配置 (MySQL/MariaDB)	5
3.1	配置 Catalog 参数文件	5
3.2	主节点到备节点 Catalog 恢复	6
3.3	同步测试	7
4	Catalog 配置 (GreatDB)	9
4.1	配置 Catalog 参数文件	9
4.2	主节点到备节点 Catalog 恢复	10
4.3	同步测试	10
5	keepalived 配置文件和脚本管理	13
5.1	keepalived 配置文件生成	13
5.2	脚本管理	13
6	集群管理	15
6.1	集群初始化	15
6.2	集群启停管理	15
6.2.1	集群启动	15
6.2.2	集群停止	16
6.2.3	手动切换集群	16
6.3	集群重配置	16
6.4	集群删除	16
6.5	集群监控	16
6.6	故障处理	17
6.7	其他	17

1.1 修改主机名，host 文件解析

主机名 info234、info236:

```
hostnamectl set-hostname info234
hostnamectl set-hostname info234 --pretty
hostnamectl set-hostname info234 --static
hostnamectl set-hostname info234 --transient
```

解析，多网卡时，为每个地址做别名解析：

```
cat /etc/hosts
127.0.0.1 localhost
192.168.87.234 info234
192.168.87.236 info236
192.168.71.224 info224ens224
192.168.71.226 info226ens224
```

1.2 配置免密

2 个节点均执行一遍：

```
sudo ssh-keygen -t rsa -P ''
sudo ssh-copy-id -i /root/.ssh/id_rsa.pub root@info234
sudo ssh-copy-id -i /root/.ssh/id_rsa.pub root@info236
```

1.3 时间同步

2 个节点均连接 ntp 服务器

1.4 修改 shell 编译器

仅 Ubuntu 配置，麒麟 V10 不需要。

```
dpkg-reconfigure dash
选择 no
```

1.5 系统日志（可选）

仅 Ubuntu 配置，麒麟 V10 不需要。

```
vi /etc/rsyslog.d/50-default.conf
```

去掉下面四行的注释

```
*.=info;*.=notice;*.=warn;\n#\n# auth,authpriv.none;\n# cron,daemon.none;\n# mail,news.none          -/var/log/messages
```

重启系统

参考《服务端安装用户指南》，区别：备节点不安装 licensed。

麒麟 V10 使用 SP3 自带 MariaDB 10.3.39，或使用 DBackup 离线包安装同版本 MariaDB。

主节点：

表 1：主节点对应安装包

安装包	安装包名称
dbackup3-agent	DBackup agent package
dbackup3-agent-file	DBackup agent file plugin
dbackup3-agent-mysql	DBackup agent mysql plugin
dbackup3-backupd	DBackup backup server
dbackup3-common	DBackup common package
dbackup3-infokist	DBackup server plugin
dbackup3-keepalived	DBackup keepalived
dbackup3-licensed	DBackup license server
dbackup3-nginx	DBackup nginx web/proxy server
dbackup3-storaged	DBackup storage server

备节点：

表 2：从节点对应安装包

安装包	安装包名称
dbackup3-agent	DBackup agent package
dbackup3-agent-file	DBackup agent file plugin
dbackup3-agent-mysql	DBackup agent mysql plugin
dbackup3-backupd	DBackup backup server
dbackup3-common	DBackup common package
dbackup3-infokist	DBackup server plugin
dbackup3-keepalived	DBackup keepalived
dbackup3-nginx	DBackup nginx web/proxy server
dbackup3-storaged	DBackup storage server

storaged 和 agent 地址均临时指向本地 IP（全部完成后指向虚拟 IP），主节点申请许可并完成初始化。

停止主备节点 backupd、Keepalived 服务：

```
systemctl stop dbackup3-backupd
systemctl disable dbackup3-backupd
systemctl stop dbackup3-keepalived
systemctl disable dbackup3-keepalived
```

3 Catalog 配置 (MySQL/MariaDB)

3.1 配置 Catalog 参数文件

节点 1 修改配置文件

```
vi /etc/mysql/mysql.conf.d/mysqld.cnf #MySQL 配置文件位置
vi /etc/my.cnf.d/mariadb-server.cnf #MariaDB 配置文件位置
```

修改内容:

```
[mysqld]
pid-file          = /var/run/mysqld/mysqld.pid #此参数麒麟V10 MariaDB 需去掉
socket            = /var/run/mysqld/mysqld.sock #此参数麒麟V10 MariaDB 需去掉
datadir           = /var/lib/mysql #此参数麒麟V10 MariaDB 需去掉
log-error         = /var/log/mysql/error.log #此参数麒麟V10 MariaDB 需去掉

server-id=234 #2个节点不重复
bind-address = 0.0.0.0
mysqlx-bind-address = 0.0.0.0 #此参数麒麟V10 MariaDB 需去掉
binlog-format = mixed
sync_binlog = 1
log_bin=/var/lib/mysql/mysql234-bin #2个节点不重复
relay-log-index = /var/lib/mysql/slave-1-relay-bin.index #2个节点不重复
relay-log = /var/lib/mysql/slave-1-relay-bin #2个节点不重复
replicate-do-db = dbackup3
rpl_semi_sync_master_enabled=ON #MariaDB配置半同步, MySQL需去掉
rpl_semi_sync_slave_enabled=ON #MariaDB配置半同步, MySQL需去掉
rpl_semi_sync_master_wait_point = AFTER_SYNC #MariaDB配置半同步, MySQL需去掉
plugin-load="rpl_semi_sync_master=semisync_master.so;rpl_semi_sync_slave=semisync_slave.so
↪" #MySQL配置半同步, MariaDB需去掉
rpl_semi_sync_slave_enabled=1 #MySQL配置半同步, MariaDB需去掉
rpl_semi_sync_master_enabled=1 #MySQL配置半同步, MariaDB需去掉
```

节点 2 修改配置文件

```
[mysqld]
pid-file          = /var/run/mysqld/mysqld.pid #此参数麒麟V10 MariaDB 需去掉
socket            = /var/run/mysqld/mysqld.sock #此参数麒麟V10 MariaDB 需去掉
datadir           = /var/lib/mysql #此参数麒麟V10 MariaDB 需去掉
log-error         = /var/log/mysql/error.log #此参数麒麟V10 MariaDB 需去掉

server-id=236 #2个节点不重复
bind-address      = 0.0.0.0
mysqlx-bind-address = 0.0.0.0 #此参数麒麟V10 MariaDB 需去掉
binlog-format = mixed
sync_binlog = 1
log_bin=/var/lib/mysql/mysql236-bin #2个节点不重复
relay-log-index = /var/lib/mysql/slave-2-relay-bin.index #2个节点不重复
relay-log = /var/lib/mysql/slave-2-relay-bin #2个节点不重复
replicate-do-db = dbackup3
rpl_semi_sync_master_enabled=ON #MariaDB配置半同步, MySQL需去掉
```

(续下页)

(接上页)

```
rpl_semi_sync_slave_enabled=ON #MariaDB配置半同步，MySQL需去掉
rpl_semi_sync_master_wait_point = AFTER_SYNC #MariaDB配置半同步，MySQL需去掉
plugin-load="rpl_semi_sync_master=semisync_master.so;rpl_semi_sync_slave=semisync_slave.so
↵" #MySQL配置半同步，MariaDB需去掉
rpl_semi_sync_slave_enabled=1 #MySQL配置半同步，MariaDB需去掉
rpl_semi_sync_master_enabled=1 #MySQL配置半同步，MariaDB需去掉
```

节点 1 和节点 2 重启 MySQL 服务生效

```
systemctl restart mysql.service
```

在节点 1 和节点 2 创建复制用户（Ubuntu MySQL）

```
create user 'replication_user'@'%' identified with mysql_native_password by '密码';
grant replication slave, replication client on *.* to 'replication_user'@'%;
grant replication slave, replication client, super, process, reload on *.* to 'replication_
↵user'@'%;
ALTER USER 'replication_user'@'%' IDENTIFIED WITH 'mysql_native_password' BY '密码';
GRANT SUPER ON *.* TO 'replication_user'@'%;
SHOW GRANTS FOR 'replication_user'@'%;
FLUSH PRIVILEGES;
```

在节点 1 和节点 2 创建复制用户（麒麟 V10 MariaDB）

```
CREATE USER 'replication_user'@'%' IDENTIFIED BY '密码';
grant replication slave, replication client on *.* to 'replication_user'@'%;
grant replication slave, replication client, super, process, reload on *.* to 'replication_
↵user'@'%;
GRANT SUPER ON *.* TO 'replication_user'@'%;
FLUSH PRIVILEGES;
```

在节点 1 和节点 2 允许 MySQL 远程登录（Ubuntu MySQL）

```
use mysql
update user set host='%' where user = 'root';
FLUSH PRIVILEGES;
GRANT ALL PRIVILEGES ON *.* TO 'root'@'%' WITH GRANT OPTION;
select user, host from user;
```

在节点 1 和节点 2 允许 MySQL 远程登录（麒麟 V10 MariaDB）

```
use mysql
grant all PRIVILEGES on *.* to root@'%' identified by '密码';
```

3.2 主节点到备节点 Catalog 恢复

节点 1 打开两个命令行窗口节点 1 窗口 1 打开只读锁

```
flush tables with read lock;
```

节点 1 窗口 2 备份数据库


```
cd /infokist
mysqldump -uroot -p dbackup3 > dbackup3.sql
scp dbackup3.sql root@info236:/infokist/
```

节点 1 窗口 1 关闭只读锁

```
unlock tables;
```

节点 2 导入备份

```
cd /infokist/
mysql -uroot -p dbackup3 < dbackup3.sql
```

3.3 同步测试

查询 1 节点日志 ID 和 POS

```
show master status;
```

在节点 2 执行同步测试

```
CHANGE MASTER TO
MASTER_HOST='info234',    #节点 1 主机名
MASTER_USER='replication_user',
MASTER_PASSWORD='密码',
MASTER_PORT=3306,
MASTER_LOG_FILE='mysql234-bin.000007',    #节点 1 查询的日志 ID
MASTER_LOG_POS=197;      #节点 1 查询的 POS 值

start slave;
show slave status\G
#重启 IO 线程，使其注册为半同步线程
STOP SLAVE IO_THREAD;
START SLAVE IO_THREAD;
```

如同步报错，说明节点 1 和 2 的数据不一致。重新导出、导入 Catalog。再重发起同步。

```
stop slave;
reset slave all;
```

重新执行上一步同步命令，注意替换日志 ID 和 POS 值

```
start slave;
show slave status\G
```


4.1 配置 Catalog 参数文件

节点 1 修改配置文件

```
vi /greatdb/conf/greatdb3306.cnf
```

修改内容

```
[mysqld]
bind-address = 0.0.0.0
server-id=234    #2 个节点不重复
replicate-do-db = dbackup3
gtid_mode=OFF
plugin-load="rpl_semi_sync_master=semisync_master.so;rpl_semi_sync_slave=semisync_slave.so
→" #GreatDB 配置半同步
rpl_semi_sync_slave_enabled=1 #GreatDB 配置半同步
rpl_semi_sync_master_enabled=1 #GreatDB 配置半同步
```

节点 2 修改配置文件

```
vi /greatdb/conf/greatdb3306.cnf
```

修改内容

```
[mysqld]
bind-address = 0.0.0.0
server-id=236    #2 个节点不重复
replicate-do-db = dbackup3
gtid_mode=OFF
plugin-load="rpl_semi_sync_master=semisync_master.so;rpl_semi_sync_slave=semisync_slave.so
→" #GreatDB 配置半同步
rpl_semi_sync_slave_enabled=1 #GreatDB 配置半同步
rpl_semi_sync_master_enabled=1 #GreatDB 配置半同步
```

节点 1 和节点 2 重启 MySQL 服务生效

```
systemctl restart greatdb-3306.service
```

在节点 1 和节点 2 创建复制用户

```
CREATE USER 'replication_user'@'%' IDENTIFIED BY '密码';
grant replication slave, replication client on *.* to 'replication_user'@'%';
grant replication slave, replication client, super, process, reload on *.* to 'replication_
→user'@'%';
GRANT SUPER ON *.* TO 'replication_user'@'%';
FLUSH PRIVILEGES;
```

在节点 1 和节点 2 配置远程用户登录

```
use mysql;
update user set host='%' where user = 'root';
FLUSH PRIVILEGES;
GRANT ALL PRIVILEGES ON *.* TO 'root'@'%' WITH GRANT OPTION;
select user, host from user;
```

在节点 1 和节点 2 修改复制用户的加密插件

```
ALTER USER 'replication_user'@'%' IDENTIFIED WITH mysql_native_password BY '密码';
```

4.2 主节点到备节点 Catalog 恢复

节点 1 备份数据库

```
cd /infokist
greatdbdump -hinfo234 -uroot -p密码 dbackup3 > dbackup3.sql
scp dbackup3.sql root@info236:/infokist/
```

节点 2 导入备份

```
cd /infokist/
greatdb -hinfo236 -uroot -p密码 dbackup3 < dbackup3.sql
```

4.3 同步测试

查询 1 节点日志 ID 和 POS

```
show master status;
```

在节点 2 执行同步测试

```
GreatDB V1.0> change replication source to
-> source_host='info214',      #节点1主机名
-> source_user='replication_user',
-> source_password='密码',
-> source_port=3306,
-> source_log_file='binlog.000003',    #节点 1 查询的日志 ID
-> source_log_pos=17588334;          #节点1查询的 POS 值

mysql> start slave;
mysql> show slave status\G
#重启 IO 线程，使其注册为半同步线程
STOP SLAVE IO_THREAD;
START SLAVE IO_THREAD;
```

如同步报错，说明节点 1 和 2 的数据不一致。重新导出、导入 Catalog。再重发起同步。

```
mysql> stop slave;
mysql> reset slave all;
```

重新执行上一步同步命令，注意替换日志 ID 和 POS 值

```
mysql> start slave;  
mysql> show slave status\G
```


5 keepalived 配置文件和脚本管理

将脚本压缩包解压，全部文件拷贝至/home/mysql 目录下。

创建以下目录，并赋权：

```
脚本位置：/home/mysql。
日志文件位置：/home/mysql/logs（自动创建）。
对端日志文件位置：/home/mysql/logs/syncposfile（自动创建）。
Ubuntu：脚本目录权限 744（chmod 744 ./*.sh），用户 root:root。
麒麟 V10：脚本目录权限 755（chmod 744 ./*.sh），用户 root:root。
```

GreatDB 在脚本中使用绝对路径 /greatdb/svr/greatdb/bin，如有变更，请修改配置脚本里的变量 DB_GREATDB_BIN_PATH。

5.1 keepalived 配置文件生成

使用 dbackup3-keepalived.sh 自动生成 keepalived.conf 文件，需要在 2 个节点各自执行：

```
./dbackup3-keepalived.sh -b
```

参数说明：

```
Please enter router_id: 集群主机 ID 使用当前主机名
How many network interfaces are there? : 根据具体 VIP 数量需求，填入网络配置数，每个 VIP
→对应一个物理网卡
Please enter vrrp_instance VI_1 interface: 当前节点的网卡名称，可使用 ipaddr 查看，如
→ens160
Please enter vrrp_instance VI_1 unicast_src_ip: 单播源地址，输入当前节点网卡对应 IP 地址
Please enter the ip of unicast_peer: 单播目标地址，输入对端节点网卡对应 IP 地址
Please enter the ip of VIP: 设置对应网卡 VIP 地址
```

5.2 脚本管理

编辑 config_def 文件

```
vi config_def
```

参数说明：

```
#Database product name. The optional names are mysql, greatdb, and mariadb.
DB_PRODUCT_NAME: 根据当前 Catalog 数据库类型填写 mysql、greatdb 或 mariadb

#Greatdb binary executable file installation directory.
DB_GREATDB_BIN_PATH="/greatdb/svr/greatdb/bin": 当数据库类型为 GreatDB 时，支持修改 bin
→目录

#Connection parameters for the local database server
DB_LOCAL_ADDRESS="127.0.0.1": 根据当前 Catalog 数据库配置填写（默认）
```

(续下页)

```
DB_LOCAL_PORT="3306": 根据当前 Catalog 数据库配置填写 (默认)
DB_LOCAL_USER="root": 根据当前 Catalog 数据库配置填写 (默认)
DB_LOCAL_PWD="root_pwd": 根据当前 Catalog 数据库配置填写 (数据库的 root 密码)

#Connection parameters for the peer database server.
DB_PEER_PORT="3306": 根据对端 Catalog 数据库配置填写 (默认)
DB_PEER_USER="root": 根据对端 Catalog 数据库配置填写 (默认)
DB_PEER_PWD="密码": 根据对端 Catalog 数据库配置填写 (数据库的 root 密码)

#The user name of the peer host operating system. The default is root.
PEER_HOST_USER_NAME="root": 对端主机系统用户名 (默认)

#The IP address that can reach the peer host, at least one address,
#multiple addresses are separated by spaces. Note that these peer host
#IP addresses must be configured with ssh public key password-free login.
PEER_HOST_ADDRESS_LIST='192.168.80.216 192.168.71.216 192.168.55.216 192.168.56.216 192.
↪168.57.216 192.168.58.216' : 填写对端的地址池, 地址间使用空格分隔, 确保每个地址都配置了
↪ssh 免密登录
PEER_HOST_SSH_PORT=22: 填写用于连接对端的 ssh 端口

#Account password used for database replication
DB_REPLICATION_USER="replica_user": 设置数据库复制专用用户名
DB_REPLICATION_USER_PASSWORD="pwd": 设置数据库复制专用用户密码

#The log path is defined by default in the logs file under the current path,
#but you can choose another path, for example: /var/log/dbbackup3.
BEST_PRACTICE_LOGS_PATH="$CURRENT_PATH/logs": 集群监控日志路径, 可填写绝对路径

#Reverse replication retry count value
REVERSE_REPLICATION_RETRY_COUNT=90: 切换后检测故障主机的连接性和数据库次数 (默认 90 次)

#Reverse replication fail sleep second
REVERSE_REPLICATION_FAIL_DELAY=10: 切换后检测故障主机的连接性和数据库可用性时间间隔 (默认
↪10s, 不建议修改)

#Switch master timeout for database apply binlog
MANUAL_SWITCH_DELAY=120: 手动切换等待 backupd 停止写入时间 (默认
↪120s, 可根据系统繁忙程度适当增加, 不建议减小, 会导致数据不一致, 反向复制创建失败)

#The time to wait for binlog to be applied before upgrading the slave node to master.
WAIT_BINLOG_DELAY=1: 从节点升主时检查数据库 binlog 是否全部应用的时间间隔 (默认
↪1s, 不建议修改)
```


在使用高可用集群时，务必使用 MySQL 模块备份 catalog，直到正式发布中解决。

使用 `best_practice.sh` 脚本进行管理

具体参数如下：

```
./best_practice.sh -s: 启动集群
./best_practice.sh -p: 检查集群复制关系
./best_practice.sh -r: 强制重同步
./best_practice.sh -m: 手动切换集群
./best_practice.sh --start-k: 启动 dbackup3-keepalived 服务
./best_practice.sh --stop-k: 停止 dbackup3-keepalived 服务
./best_practice.sh --stat-k: 检查 status dbackup3-keepalived 服务
./best_practice.sh --start-b: 启动 start dbackup3-backupd 服务
./best_practice.sh --stop-b: 停止 dbackup3-backupd 服务
./best_practice.sh --stat-b: 检查 dbackup3-backupd 服务
./best_practice.sh --add-rp: 自动将 MySQL 复制状态检查加入 crontab
./best_practice.sh --del-rp: 自动将 MySQL 复制状态检查移除 crontab
./best_practice.sh --start-c: 启动 crontab 服务
./best_practice.sh --stop-c: 停止 crontab 服务
./best_practice.sh --stat-c: 检查 crontab 服务状态
./best_practice.sh --stat-db: 检查当前节点数据库复制状态
```

6.1 集群初始化

1. 确保主节点没有创建从库复制关系，`show slave` 结果为空。
2. 重同步从节点，在从节点运行脚本 `./best_practice.sh -r`。
3. 在主节点启动 `keepalived`： `./best_practice.sh --start-k`，初始化使用 `dump` 数据量较大时，时间较长。

备注： 集群初始化过程中，以及初始化后，无需使用 `./best_practice.sh -s` 再执行启动集群。

6.2 集群启停管理

6.2.1 集群启动

1. 在任意节点运行脚本 `./best_practice.sh -s`。

6.2.2 集群停止

- 1. 在备节点停止集群服务 `./best_practice.sh --stop-k`。
- 2. 在主节点停止集群服务 `./best_practice.sh --stop-k`。

6.2.3 手动切换集群

- 1. 在主节点执行 `./best_practice.sh -m`。

6.3 集群重配置

- 1. 停止集群服务。
- 2. 修改集群配置文件。
- 3. 重新加载 Keepalived 配置。
- 4. 启动集群服务。

6.4 集群删除

- 1. 停止集群服务。
- 2. 清除集群配置文件。
- 3. 清空 MySQL 复制。

6.5 集群监控

- 1. 检查主备节点 `./best_practice.sh -p`。
- 2. 检查复制状态 `./best_practice.sh --stat-db`。
- 3. 定期检查复制状态，为主备节点添加 crontab 任务，运行 `./best_practice.sh --add-rp` 和 `./best_practice.sh --start-c`。
- 4. 日志输出：当前脚本运行路径下的 logs 目录。

表 3：日志信息

信息	说明
Slave has read all relay log; waiting for the slave I/O thread to update it Replica has read all relay log; waiting for more updates	从节点升级为主节点时，等待日志应用
remote xx ip systemctl stop dbbackup3-backupd error	从节点升级为主节点时，尝试关闭原主节点 backupd 服务失败
try to execute xx command on the remote host	切换后尝试在原主节点运行重建复制脚本
The remote host is not reachable.	切换后无法连接到原主节点，无法重建复制

续下页

表 3 – 接上页

信息	说明
MySQL service is active on remote host xx ip	切换后原主节点数据库服务未启动，无法重建复制
start try reverse replication to the remote xx ip	切换后尝试连接原主节点
dump dbackup3 error	初始化同步数据库时，Catalog 备份失败
mysql restore xx failed	初始化同步数据库时，Catalog 恢复失败
failed to execute ‘show master status;’ on remote xx ip	创建复制时，获取主节点 Catalog 状态失败
failed to change to slave	从节点创建复制失败
Failed to execute ‘SHOW SLAVE STATUS\G’	检查当前或对端节点的数据库复制状态失败
The local database service is currently a slave node and cannot be manually switched.	当前节点为从节点，不能执行手工切换

6.6 故障处理

1. 数据库复制发生异常时，在从节点使用 `./best_practice.sh -r` 重同步。
2. 发生集群脑裂时：
 - 2.1. 根据日志判断发生切换的时间点；
 - 2.2. 两个节点全部确认服务关闭 `./best_practice.sh --stop-k` 和 `./best_practice.sh --stop-b`；
 - 2.3. 确认可用的主节点，做集群初始化。参考 6.1 章节。

6.7 其他

在 MySQL 查看复制具体报错信息：

```
select * from performance_schema.replication_applier_status_by_worker\G
```



全国销售热线：400-650-0081

电话：+86 20 32053160

总部地址：广州市科学城科学大道243号总部经济区A5栋9楼

全国服务热线：400-003-3191

网址：www.scutech.com