**MogDB 及HA安装使用指南**

**（2020-10-10）**

云和恩墨(北京)信息技术有限公司

技术顾问 李宏达

http://www.enmotech.com

**文档控制：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序** | **版本号** | **更改人** | **日期** | **备注** |
| 1 | 1.0版 | 李宏达 | 2020-08-30 | 初始版本 |
| 2 |  |  | 2020-10-19 | 增加HA内容 |
|  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **编制** | 李宏达 | （签字） | 日期 | 2020-08-30 |
| **校对** |  | （签字） | 日期 |  |
| **审核** |   | （签字） | 日期 |  |
| **批准** |  | （签字） | 日期 |  |

目录

[1. 安装概述 - 4 -](#_Toc54683826)

[2. 安装准备 - 4 -](#_Toc54683827)

[2.1 环境要求 - 4 -](#_Toc54683828)

[2.2 修改操作系统配置 - 5 -](#_Toc54683829)

[2.2.1 关闭防火墙，selinux - 5 -](#_Toc54683830)

[2.2.2 设置字符集参数 - 5 -](#_Toc54683831)

[2.2.3 设置时区和时间 - 6 -](#_Toc54683832)

[2.2.4 关闭swap交换内存 - 6 -](#_Toc54683833)

[2.2.5 设置网卡MTU值 - 6 -](#_Toc54683834)

[3. 安装MogDB - 6 -](#_Toc54683835)

[3.1 创建XML配置文件 - 6 -](#_Toc54683836)

[3.2 示例 - 7 -](#_Toc54683837)

[3.2.1 单节点配置文件 - 7 -](#_Toc54683838)

[3.2.2 一主一备配置文件 - 8 -](#_Toc54683839)

[3.2.3 一主二备配置文件 - 9 -](#_Toc54683840)

[3.2.4 一主三备配置文件 - 10 -](#_Toc54683841)

[3.2.5 一主四备配置文件 - 12 -](#_Toc54683842)

[3.3 环境初始化 - 14 -](#_Toc54683843)

[3.4 初始化脚本 - 14 -](#_Toc54683844)

[3.5 执行安装 - 16 -](#_Toc54683845)

[4. HA部署(一主一备) - 17 -](#_Toc54683846)

[1.4 部署前准备 - 17 -](#_Toc54683847)

[1.4 修改配置信息 - 18 -](#_Toc54683848)

[1.4 运行服务 - 19 -](#_Toc54683849)

[5. HA切换测试 - 20 -](#_Toc54683850)

**[5.1](#_Toc54683854)****[主动切换](#_Toc54683854)** [- 20 -](#_Toc54683854)

**[5.2](#_Toc54683855)****[被动切换](#_Toc54683855)** [- 23 -](#_Toc54683855)

[6. 卸载MogDB - 27 -](#_Toc54683856)

**[6.1](#_Toc54683861)****[执行卸载](#_Toc54683861)** [- 27 -](#_Toc54683861)

**[6.2](#_Toc54683862)****[一键式清理环境](#_Toc54683862)** [- 27 -](#_Toc54683862)

# 安装概述

MogDB支持单机部署和单机HA部署两种部署方式。单机部署时，可在一个主机部署多个数据库实例，但为了数据安全，不建议用户这样部署。单机HA部署支持一台主机和最少一台备机，备机一共最多4台的配置方式。

单机部署和单机HA部署均支持精简模式和兼容模式。精简模式下，您可以安装MogDB数据库并正常使用；兼容模式是在精简模式的基础上做了兼容增强，安装兼容包后会兼容主流数据库的接口名称，便于习惯主流数据库的用户使用MogDB。

# 安装准备

## 2.1 环境要求

硬件环境要求

| 项目 | 配置描述 |
| --- | --- |
| 最小内存 | 功能调试32GB以上。性能测试和商业部署时，单实例部署建议128GB以上。复杂的查询对内存的需求量比较高，在高并发场景下，可能出现内存不足。此时建议使用大内存的机器，或使用负载管理限制系统的并发。 |
| CPU | 功能调试最小1×8 核 2.0GHz。性能测试和商业部署时，单实例部署建议1×16核 2.0GHz。CPU超线程和非超线程两种模式都支持。但是，MogDB各节点的设置需保持一致。 |
| 硬盘 | 用于安装MogDB的硬盘需最少满足如下要求：* 至少1GB用于安装MogDB的应用程序包。
* 每个主机需大约300MB用于元数据存储。
* 预留70%以上的磁盘剩余空间用于数据存储。

建议系统盘配置为Raid1，数据盘配置为Raid5，且规划4组Raid5数据盘用于安装MogDB。有关Raid的配置方法在本手册中不做介绍。请参考硬件厂家的手册或互联网上的方法进行配置，其中Disk Cache Policy一项需要设置为Disabled，否则机器异常掉电后有数据丢失的风险。MogDB支持使用SSD盘作为数据库的主存储设备，支持SAS接口和NVME协议的SSD盘，以RAID的方式部署使用。 |
| 网络要求 | 300兆以上以太网。建议网卡设置为双网卡冗余bond。有关网卡冗余bond的配置方法在本手册中不做介绍。请参考硬件厂商的手册或互联网上的方法进行配置。MogDB网络如果配置bond，请保证bond模式一致，不一致的bond配置可能导致MogDB工作异常。 |

软件环境要求

| 软件类型 | 配置描述 |
| --- | --- |
| Linux操作系统 | openEuler 20.3LTS和CentOS 7.6 |
| Linux文件系统 | 剩余inode个数 > 15亿（推荐） |
| 工具 | Huawei JDK 1.8.0、psmisc、bzip2 |
| Python | * openEuler：支持Python 3.7.X
* CentOS：支持Python 3.6.X
 |

软件依赖要求

| 所需软件 | 建议版本 |
| --- | --- |
| libaio-devel | 建议版本：0.3.109-13 |
| flex | 要求版本：2.5.31 以上 |
| bison | 建议版本：2.7-4 |
| ncurses-devel | 建议版本：5.9-13.20130511 |
| glibc.devel | 建议版本：2.17-111 |
| patch | 建议版本：2.7.1-10 |

注：实际测试4G内存也能安装（需要修改配置文件内存参数），4G以上内存不需要修改。

## 2.2 修改操作系统配置

### 关闭防火墙，selinux

systemctl disable firewalld.service

systemctl stop firewalld.service

setenforce=0

sed -i '/^SELINUX=/c'SELINUX=disabled /etc/selinux/config

ps:红旗系统需要关闭防火墙，行里默认CentOS没有 firewalld服务。

### 设置字符集参数

将各数据库节点的字符集设置为相同的字符集，可以在/etc/profile文件中添加"export LANG=XXX"（XXX为Unicode编码）。

Ps:可暂时不设置

### 设置时区和时间

将各数据库节点的时区设置为相同时区，可以将/usr/share/zoneinfo/目录下的时区文件拷贝为/etc/localtime文件。

cp /usr/share/zoneinfo/$地区/$时区 /etc/localtime使用date -s命令将各主机的时间设置为统一时间，举例如下。

date -s Mon May 11 16:42:11 CST 2020

### 关闭swap交换内存

在各数据库节点上，使用swapoff -a命令将交换内存关闭。

swapoff -a

ps：内存较大时关闭交换内存（建议32G以上）

### 设置网卡MTU值

将各数据库节点的网卡MTU值设置为相同大小。MTU值推荐8192，要求不小于1500

ifconfig 网卡编号 mtu 值

# 安装MogDB

## 3.1 创建XML配置文件

安装MogDB前需要创建XML文件。XML文件包含部署MogDB的服务器信息、安装路径、IP地址以及端口号等。用于告知MogDB如何部署。用户需根据不同场配置对应的XML文件。

参数说明

| 实例类型 | 参数 | 说明 |
| --- | --- | --- |
| 整体信息 | clusterName | MogDB名称。 |
| nodeNames | MogDB中主机名称。 |
| backIp1s | 主机在后端存储网络中的IP地址（内网IP）。所有MogDB主机使用后端存储网络通讯。 |
| gaussdbAppPath | MogDB程序安装目录。此目录应满足如下要求：* 磁盘空间>1GB
* 与数据库所需其它路径相互独立，没有包含关系。
 |
| gaussdbLogPath | MogDB运行日志和操作日志存储目录。此目录应满足如下要求：* 磁盘空间建议根据主机上的数据库节点数规划。数据库节点预留1GB空间的基础上，再适当预留冗余空间。
* 与MogDB所需其它路径相互独立，没有包含关系。

此路径可选。不指定的情况下，MogDB安装时会默认指定“$GAUSSLOG/安装用户名”作为日志目录。 |
| tmpdbPath | 数据库临时文件存放目录。若不配置tmpdbPath，默认存放在/opt/huawei/wisequery/perfadm\_db目录下。 |
| gaussdbToolPath | MogDB系统工具目录，主要用于存放互信工具等。此目录应满足如下要求：* 磁盘空间>100MB
* 固定目录，与数据库所需其它目录相互独立，没有包含关系。

此目录可选。不指定的情况下，MogDB安装时会默认指定“/opt/huawei/wisequery”作为数据库系统工具目录。 |
| corePath | MogDB core文件的指定目录。 |
| clusterType | MogDB类型，MogDB拓扑类型；可选字段。“single-inst”表示单机一主多备部署形态。 |

## 3.2 示例

### 单节点配置文件

<?xml version="1.0" encoding="UTF-8"?>

<ROOT>

 <!-- MogDB整体信息 -->

 <CLUSTER>

 <PARAM name="clusterName" value="dbCluster" />

 <PARAM name="nodeNames" value="node1" />

 <PARAM name="backIp1s" value="192.168.122.101"/>

 <PARAM name="gaussdbAppPath" value="/opt/gaussdb/app" />

 <PARAM name="gaussdbLogPath" value="/var/log/gaussdb" />

 <PARAM name="gaussdbToolPath" value="/opt/huawei/wisequery" />

 <PARAM name="corePath" value="/opt/MogDB/corefile"/>

 <PARAM name="clusterType" value="single-inst"/>

 </CLUSTER>

<!-- 每台服务器上的节点部署信息 -->

 <DEVICELIST>

 <!-- node1上的节点部署信息 -->

 <DEVICE sn="1000001">

 <PARAM name="name" value="node1"/>

 <PARAM name="azName" value="AZ1"/>

 <PARAM name="azPriority" value="1"/>

 <!-- 如果服务器只有一个网卡可用，将backIP1和sshIP1配置成同一个IP -->

 <PARAM name="backIp1" value="192.168.122.101"/>

 <PARAM name="sshIp1" value="192.168.122.101"/>

 <!--dbnode-->

 <PARAM name="dataNum" value="1"/>

 <PARAM name="dataPortBase" value="26000"/>

 <PARAM name="dataNode1" value="/opt/gaussdb/data/db1"/>

 </DEVICE>

 </DEVICELIST>

</ROOT>

### 一主一备配置文件

<?xml version="1.0" encoding="UTF-8"?>

<ROOT>

 <!-- MogDB整体信息 -->

 <CLUSTER>

 <PARAM name="clusterName" value="dbCluster" />

 <PARAM name="nodeNames" value="node1,node2" />

 <PARAM name="backIp1s" value="192.168.122.101,192.168.122.102"/>

 <PARAM name="gaussdbAppPath" value="/opt/gaussdb/app" />

 <PARAM name="gaussdbLogPath" value="/var/log/gaussdb" />

 <PARAM name="gaussdbToolPath" value="/opt/huawei/wisequery" />

 <PARAM name="corePath" value="/opt/MogDB/corefile"/>

 <PARAM name="clusterType" value="single-inst"/>

 </CLUSTER>

 <!-- 每台服务器上的节点部署信息 -->

 <DEVICELIST>

 <!-- node1上的节点部署信息 -->

 <DEVICE sn="1000001">

 <PARAM name="name" value="node1"/>

 <PARAM name="azName" value="AZ1"/>

 <PARAM name="azPriority" value="1"/>

 <!-- 如果服务器只有一个网卡可用，将backIP1和sshIP1配置成同一个IP -->

 <PARAM name="backIp1" value="192.168.122.101"/>

 <PARAM name="sshIp1" value="192.168.122.101"/>

 <!--dbnode-->

 <PARAM name="dataNum" value="1"/>

 <PARAM name="dataPortBase" value="26000"/>

 <PARAM name="dataNode1" value="/opt/gaussdb/data/db1,node2,/opt/gaussdb/data/db1"/>

 </DEVICE>

 <!-- node2上的节点部署信息，其中“name”的值配置为主机名称 -->

 <DEVICE sn="1000002">

 <PARAM name="name" value="node2"/>

 <PARAM name="azName" value="AZ1"/>

 <PARAM name="azPriority" value="1"/>

 <!-- 如果服务器只有一个网卡可用，将backIP1和sshIP1配置成同一个IP -->

 <PARAM name="backIp1" value="192.168.122.102"/>

 <PARAM name="sshIp1" value="192.168.122.102"/>

 </DEVICE>

 </DEVICELIST>

</ROOT>

注：替换主机名和IP，其他参数默认即可。

### 一主二备配置文件

<?xml version="1.0" encoding="UTF-8"?>

<ROOT>

 <!-- MogDB整体信息 -->

 <CLUSTER>

 <PARAM name="clusterName" value="dbCluster" />

 <PARAM name="nodeNames" value="node1,node2,node3" />

 <PARAM name="backIp1s" value="192.168.122.101,192.168.122.102, 192.168.122.103"/>

 <PARAM name="gaussdbAppPath" value="/opt/gaussdb/app" />

 <PARAM name="gaussdbLogPath" value="/var/log/gaussdb" />

 <PARAM name="gaussdbToolPath" value="/opt/huawei/wisequery" />

 <PARAM name="corePath" value="/opt/MogDB/corefile"/>

 <PARAM name="clusterType" value="single-inst"/>

 </CLUSTER>

 <!-- 每台服务器上的节点部署信息 -->

 <DEVICELIST>

 <!-- node1上的节点部署信息 -->

 <DEVICE sn="1000001">

 <PARAM name="name" value="node1"/>

 <PARAM name="azName" value="AZ1"/>

 <PARAM name="azPriority" value="1"/>

 <!-- 如果服务器只有一个网卡可用，将backIP1和sshIP1配置成同一个IP -->

 <PARAM name="backIp1" value="192.168.122.101"/>

 <PARAM name="sshIp1" value="192.168.122.101"/>

 <!--dbnode-->

 <PARAM name="dataNum" value="1"/>

 <PARAM name="dataPortBase" value="26000"/>

 <PARAM name="dataNode1" value="/opt/gaussdb/data/db1,node2,/opt/gaussdb/data/db1, node3,/opt/gaussdb/data/db1 "/>

 </DEVICE>

 <!-- node2上的节点部署信息，其中“name”的值配置为主机名称 -->

 <DEVICE sn="1000002">

 <PARAM name="name" value="node2"/>

 <PARAM name="azName" value="AZ1"/>

 <PARAM name="azPriority" value="1"/>

 <!-- 如果服务器只有一个网卡可用，将backIP1和sshIP1配置成同一个IP -->

 <PARAM name="backIp1" value="192.168.122.102"/>

 <PARAM name="sshIp1" value="192.168.122.102"/>

 </DEVICE>

<!—node3上的节点部署信息，其中“name”的值配置为主机名称 -->

 <DEVICE sn="1000002">

 <PARAM name="name" value="node3"/>

 <PARAM name="azName" value="AZ1"/>

 <PARAM name="azPriority" value="1"/>

 <!-- 如果服务器只有一个网卡可用，将backIP1和sshIP1配置成同一个IP -->

 <PARAM name="backIp1" value="192.168.122.103"/>

 <PARAM name="sshIp1" value="192.168.122.103"/>

 </DEVICE>

 </DEVICELIST>

</ROOT>

ps：替换主机名和IP，其他参数默认即可。/opt/gaussdb/data/db1为数据存储路径，可根据需要替换

### 一主三备配置文件

<?xml version="1.0" encoding="UTF-8"?>

<ROOT>

 <!-- MogDB整体信息 -->

 <CLUSTER>

 <PARAM name="clusterName" value="dbCluster" />

 <PARAM name="nodeNames" value="node1,node2,node3,node4" />

 <PARAM name="backIp1s" value="192.168.122.101,192.168.122.102, 192.168.122.103,192.168.122.104"/>

 <PARAM name="gaussdbAppPath" value="/opt/gaussdb/app" />

 <PARAM name="gaussdbLogPath" value="/var/log/gaussdb" />

 <PARAM name="gaussdbToolPath" value="/opt/huawei/wisequery" />

 <PARAM name="corePath" value="/opt/MogDB/corefile"/>

 <PARAM name="clusterType" value="single-inst"/>

 </CLUSTER>

 <!-- 每台服务器上的节点部署信息 -->

 <DEVICELIST>

 <!-- node1上的节点部署信息 -->

 <DEVICE sn="1000001">

 <PARAM name="name" value="node1"/>

 <PARAM name="azName" value="AZ1"/>

 <PARAM name="azPriority" value="1"/>

 <!-- 如果服务器只有一个网卡可用，将backIP1和sshIP1配置成同一个IP -->

 <PARAM name="backIp1" value="192.168.122.101"/>

 <PARAM name="sshIp1" value="192.168.122.101"/>

 <!--dbnode-->

 <PARAM name="dataNum" value="1"/>

 <PARAM name="dataPortBase" value="26000"/>

 <PARAM name="dataNode1" value="/opt/gaussdb/data/db1,node2,/opt/gaussdb/data/db1, node3,/opt/gaussdb/data/db1, node4,/opt/gaussdb/data/db1"/>

 </DEVICE>

 <!-- node2上的节点部署信息，其中“name”的值配置为主机名称 -->

 <DEVICE sn="1000002">

 <PARAM name="name" value="node2"/>

 <PARAM name="azName" value="AZ1"/>

 <PARAM name="azPriority" value="1"/>

 <!-- 如果服务器只有一个网卡可用，将backIP1和sshIP1配置成同一个IP -->

 <PARAM name="backIp1" value="192.168.122.102"/>

 <PARAM name="sshIp1" value="192.168.122.102"/>

 </DEVICE>

<!—node3上的节点部署信息，其中“name”的值配置为主机名称 -->

 <DEVICE sn="1000002">

 <PARAM name="name" value="node3"/>

 <PARAM name="azName" value="AZ1"/>

 <PARAM name="azPriority" value="1"/>

 <!-- 如果服务器只有一个网卡可用，将backIP1和sshIP1配置成同一个IP -->

 <PARAM name="backIp1" value="192.168.122.103"/>

 <PARAM name="sshIp1" value="192.168.122.103"/>

 </DEVICE>

<!—node4上的节点部署信息，其中“name”的值配置为主机名称 -->

 <DEVICE sn="1000002">

 <PARAM name="name" value="node4"/>

 <PARAM name="azName" value="AZ1"/>

 <PARAM name="azPriority" value="1"/>

 <!-- 如果服务器只有一个网卡可用，将backIP1和sshIP1配置成同一个IP -->

 <PARAM name="backIp1" value="192.168.122.104"/>

 <PARAM name="sshIp1" value="192.168.122.104"/>

 </DEVICE>

 </DEVICELIST>

</ROOT>

### 一主四备配置文件

<?xml version="1.0" encoding="UTF-8"?>

<ROOT>

 <!-- MogDB整体信息 -->

 <CLUSTER>

 <PARAM name="clusterName" value="dbCluster" />

 <PARAM name="nodeNames" value="node1,node2,node3,node4,node5" />

 <PARAM name="backIp1s" value="192.168.122.101,192.168.122.102, 192.168.122.103, 192.168.122.104, 192.168.122.105"/>

 <PARAM name="gaussdbAppPath" value="/opt/gaussdb/app" />

 <PARAM name="gaussdbLogPath" value="/var/log/gaussdb" />

 <PARAM name="gaussdbToolPath" value="/opt/huawei/wisequery" />

 <PARAM name="corePath" value="/opt/MogDB/corefile"/>

 <PARAM name="clusterType" value="single-inst"/>

 </CLUSTER>

 <!-- 每台服务器上的节点部署信息 -->

 <DEVICELIST>

 <!-- node1上的节点部署信息 -->

 <DEVICE sn="1000001">

 <PARAM name="name" value="node1"/>

 <PARAM name="azName" value="AZ1"/>

 <PARAM name="azPriority" value="1"/>

 <!-- 如果服务器只有一个网卡可用，将backIP1和sshIP1配置成同一个IP -->

 <PARAM name="backIp1" value="192.168.122.101"/>

 <PARAM name="sshIp1" value="192.168.122.101"/>

 <!--dbnode-->

 <PARAM name="dataNum" value="1"/>

 <PARAM name="dataPortBase" value="26000"/>

 <PARAM name="dataNode1" value="/opt/gaussdb/data/db1,node2,/opt/gaussdb/data/db1, node3,/opt/gaussdb/data/db1, node4,/opt/gaussdb/data/db1, node5,/opt/gaussdb/data/db1"/>

 </DEVICE>

 <!-- node2上的节点部署信息，其中“name”的值配置为主机名称 -->

 <DEVICE sn="1000002">

 <PARAM name="name" value="node2"/>

 <PARAM name="azName" value="AZ1"/>

 <PARAM name="azPriority" value="1"/>

 <!-- 如果服务器只有一个网卡可用，将backIP1和sshIP1配置成同一个IP -->

 <PARAM name="backIp1" value="192.168.122.102"/>

 <PARAM name="sshIp1" value="192.168.122.102"/>

 </DEVICE>

<!—node3上的节点部署信息，其中“name”的值配置为主机名称 -->

 <DEVICE sn="1000002">

 <PARAM name="name" value="node3"/>

 <PARAM name="azName" value="AZ1"/>

 <PARAM name="azPriority" value="1"/>

 <!-- 如果服务器只有一个网卡可用，将backIP1和sshIP1配置成同一个IP -->

 <PARAM name="backIp1" value="192.168.122.103"/>

 <PARAM name="sshIp1" value="192.168.122.103"/>

 </DEVICE>

<!—node4上的节点部署信息，其中“name”的值配置为主机名称 -->

 <DEVICE sn="1000002">

 <PARAM name="name" value="node4"/>

 <PARAM name="azName" value="AZ1"/>

 <PARAM name="azPriority" value="1"/>

 <!-- 如果服务器只有一个网卡可用，将backIP1和sshIP1配置成同一个IP -->

 <PARAM name="backIp1" value="192.168.122.104"/>

 <PARAM name="sshIp1" value="192.168.122.104"/>

 </DEVICE>

<!—node5上的节点部署信息，其中“name”的值配置为主机名称 -->

 <DEVICE sn="1000002">

 <PARAM name="name" value="node5"/>

 <PARAM name="azName" value="AZ1"/>

 <PARAM name="azPriority" value="1"/>

 <!-- 如果服务器只有一个网卡可用，将backIP1和sshIP1配置成同一个IP -->

 <PARAM name="backIp1" value="192.168.122.105"/>

 <PARAM name="sshIp1" value="192.168.122.105"/>

 </DEVICE>

 </DEVICELIST>

</ROOT>

## 3.3 环境初始化

**操作步骤**

**a：**以root用户登录待安装MogDB的任意主机，并按规划创建存放安装包的目录。

 mkdir -p /opt/software/MogDB

chmod 755 -R /opt/software
 **b**：将安装包“MogDB-1.0.0-CentOS-64bit .tar.gz”和配置文件“clusterconfig.xml”都上传至上一步所创建的目录中。

**c**：在安装包所在的目录下，解压安装包。

cd /opt/software/MogDB

tar -zxvf MogDB-1.0.0-CentOS-64bit .tar.gz

**d：**进入到工具脚本存放目录下。

cd /opt/software/MogDB/script

**e：**为确保openssl版本正确，执行预安装前请加载安装包中lib库。执行命令如下，其中{packagePath}为用户安装包放置的路径，本示例中为/opt/software/MogDB。

export LD\_LIBRARY\_PATH={packagePath}/script/gspylib/clib:$LD\_LIBRARY\_PATH

**f：**为确保成功安装，检查 hostname 与 /etc/hostname 是否一致。预安装过程中，会对hostname进行检查。

g：安装python3.6.5

yum -y install zlib-devel bzip2-devel openssl-devel ncurses-devel sqlite-devel readline-devel tk-devel gdbm-devel db4-devel libpcap-devel xz-devel bzip2 libaio

wget https://www.python.org/ftp/python/3.6.5/Python-3.6.5.tgz

mkdir -p /usr/local/python3

tar -zxvf Python-3.6.5.tgz

cd Python-3.6.5

./configure --prefix=/usr/local/python3 && make && make install

ln -s /usr/local/python3/bin/python3 /usr/bin/python3

ln -s /usr/local/python3/bin/pip3 /usr/bin/pip3

cp libpython3.6m.so.1.0 /usr/lib64

h:红旗系统需要处理release

 rm -f /etc/asianux-release

 rm -f /etc/centos-release

 echo “CentOS Linux release 7.6.1810 (Core)” > /etc/centos-release

## 3.4 初始化脚本

/opt/software/MogDB/script/gs\_preinstall -U omm -G dbgrp -X /opt/software/MogDB/clusterconfig.xml

Parsing the configuration file.
Successfully parsed the configuration file.
Installing the tools on the local node.
Successfully installed the tools on the local node.
Distributing package.
Begin to distribute package to tool path.
Successfully distribute package to tool path.
Begin to distribute package to package path.
Successfully distribute package to package path.
Successfully distributed package.
Installing the tools in the cluster.
Successfully installed the tools in the cluster.
Checking hostname mapping.
Successfully checked hostname mapping.
Checking OS version.
Successfully checked OS version.
Creating cluster's path.
Successfully created cluster's path.
Setting SCTP service.
Successfully set SCTP service.
Set and check OS parameter.
Setting OS parameters.
Successfully set OS parameters.
Set and check OS parameter completed.
Preparing CRON service.
Successfully prepared CRON service.
Preparing SSH service.
Successfully prepared SSH service.
Setting user environmental variables.
Successfully set user environmental variables.
Configuring alarms on the cluster nodes.
Successfully configured alarms on the cluster nodes.
Setting the dynamic link library.
Successfully set the dynamic link library.
Setting Cgroup.
Successfully set Cgroup.
Set ARM Optimization.
Successfully set ARM Optimization.
Setting finish flag.
Successfully set finish flag.
Preinstallation succeeded.

Ps:运行过程中需要输入相关密码，需保证复杂度要求。（大小写，字符数字）

## 3.5 执行安装

cd /opt/software/MogDB/script
chmod -R 755 /opt/software/MogDB

chown -R omm:dbgrp /opt/software/MogDB

su - omm

/opt/software/MogDB/script/gs\_install -X /opt/software/MogDB/clusterconfig.xml

Parsing the configuration file.
Check preinstall on every node.
Successfully checked preinstall on every node.
Creating the backup directory.
Successfully created the backup directory.
begin deploy..
Installing the cluster.
begin prepare Install Cluster..
Checking the installation environment on all nodes.
begin install Cluster..
Installing applications on all nodes.
Successfully installed APP.
begin init Instance..
encrypt ciper and rand files for database.
Please enter password for database:
Please repeat for database:
begin to create CA cert files
The sslcert will be generated in /opt/gaussdb/cluster/app/share/sslcert/om
Cluster installation is completed.
Configuring.
Deleting instances from all nodes.
Successfully deleted instances from all nodes.
Checking node configuration on all nodes.
Initializing instances on all nodes.
Updating instance configuration on all nodes.
Check consistence of memCheck and coresCheck on DN nodes.
Successful check consistence of memCheck and coresCheck on all nodes.
Configuring pg\_hba on all nodes.
Configuration is completed.
Successfully started cluster.
Successfully installed application.

注：在执行过程中，用户需根据提示输入数据库的密码，密码具有一定的复杂度，为保证用户正常使用该数据库，请记住输入的数据库密码。

# HA部署(一主一备)

## 部署前准备

* 防火墙

检查关闭防火墙，实际环境可按端口配置

* 机器信息

node1 192.168.122.101 --主库

node2 192.168.122.102 --备库

node3 192.168.122.103 --仲裁

* 时间校对

使用ntp或chronyd校对主库和备库以及仲裁机器的时间

* 配置用户的sudo权限(主备及etcd仲裁节点)

chmod +w /etc/sudoers

which ifconfig

/usr/sbin/ifconfig

vi /etc/sudoers

omm ALL=(ALL) NOPASSWD: /usr/sbin/ifconfig

chmod -w /etc/sudoers

* 所需安装包

etcd-v3.3.25-linux-amd64.tar

MogDBHA.tar.gz

venv.tar

以omm用户上传如上三个包到/home/omm/路径下并解压

检查权限应为

* 数据库安装xml

由于目前HA只支持一主一备，所以本例使用如下xml配置

cat /data/clusterconfig.xml

<?xml version="1.0" encoding="UTF-8"?>

<ROOT>

 <!-- MogDB整体信息 -->

 <CLUSTER>

 <PARAM name="clusterName" value=" ms1" />

 <PARAM name="nodeNames" value="node1,node2" />

 <PARAM name="backIp1s" value="192.168.122.101,192.168.122.102"/>

 <PARAM name="gaussdbAppPath" value="/opt/gaussdb/app" />

 <PARAM name="gaussdbLogPath" value="/var/log/gaussdb" />

 <PARAM name="gaussdbToolPath" value="/opt/huawei/wisequery" />

 <PARAM name="corePath" value="/opt/MogDB/corefile"/>

 <PARAM name="clusterType" value="single-inst"/>

 </CLUSTER>

 <!-- 每台服务器上的节点部署信息 -->

 <DEVICELIST>

 <!-- node1上的节点部署信息 -->

 <DEVICE sn="1000001">

 <PARAM name="name" value="node1"/>

 <PARAM name="azName" value="AZ1"/>

 <PARAM name="azPriority" value="1"/>

 <!-- 如果服务器只有一个网卡可用，将backIP1和sshIP1配置成同一个IP -->

 <PARAM name="backIp1" value="192.168.122.101"/>

 <PARAM name="sshIp1" value="192.168.122.101"/>

 <!--dbnode-->

 <PARAM name="dataNum" value="1"/>

 <PARAM name="dataPortBase" value="26000"/>

 <PARAM name="dataNode1" value="/opt/gaussdb/data/db1,node2,/opt/gaussdb/data/db1"/>

 </DEVICE>

 <!-- node2上的节点部署信息，其中“name”的值配置为主机名称 -->

 <DEVICE sn="1000002">

 <PARAM name="name" value="node2"/>

 <PARAM name="azName" value="AZ1"/>

 <PARAM name="azPriority" value="1"/>

 <!-- 如果服务器只有一个网卡可用，将backIP1和sshIP1配置成同一个IP -->

 <PARAM name="backIp1" value="192.168.122.102"/>

 <PARAM name="sshIp1" value="192.168.122.102"/>

 </DEVICE>

 </DEVICELIST>

</ROOT>

## 修改配置信息

**上传venv包文件到omm用户home目录，设置环境变量(主备节点 IP 192.168.122.101 192.168.122.102 以及etcd仲裁节点IP 192.168.122.103)**

su - omm

cd venv/bin

删除自带的软连接

rm python\*

将python 指向实际python3

which python3

/bin/python3.6

ln -s /bin/python3.6 python3

**配置仲裁节点（IP 192.168.122.103）**

vi MogDBHA/etcdConfig.py

ETCD\_HOST="192.168.122.103"

ETCD\_PORT=2379

WEB\_PORT=8080

**初始化配置HAconfig(主节点配置)**

vi MogDBHA/HAconfig.py

CLUSTERNAME = 'ms1'

PRIMARY="192.168.122.101"

STANDBY="192.168.122.102"

VIP="192.168.122.111"

ETCD\_WEB = "192.168.122.103:8080"

**配置NodeConfig.py(主节点配置)**

vi MogDBHA/NodeConfig.py

CLUSTERNAME = 'ms1'

VIP = '192.168.122.111'

ARPING='192.168.122.254'

OS\_USER='omm'

HOSTS=['192.168.122.101','192.168.122.102']

HEARTBEAT\_HOSTS=['192.168.122.101','192.168.122.102']

WEB\_PORT=8081

PING\_CHECK=['192.168.122.111','192.168.122.101','192.168.122.102']

ETCD\_WEB="192.168.122.103:8080"

DB\_PORT=26000

复制配置文件到每一个节点(备机和仲裁)

scp MogDBHA/\* omm@192.168.122.102:/home/omm/MogDBHA/

scp MogDBHA/\* omm@192.168.122.103:/home/omm/MogDBHA/

## 运行服务

启动etcd server(仲裁)

nohup ./etcd-v3.3.25-linux-amd64/etcd --name=default --listen-client-urls=http://192.168.122.103:2379 --advertise-client-urls=http://192.168.122.103:2379 &

执行etcd\_run.py脚本(仲裁)

./venv/bin/python3 MogDBHA/etcd\_run.py

注册HA信息(主/备)

./venv/bin/python3 MogDBHA/HAconfig.py

使用如下命令重新注册HA信息

./venv/bin/python3 MogDBHA/HAconfig.py force-new

运行node监控(主、备)

./venv/bin/python3 MogDBHA/node\_run.py

当前版本若主备库的node\_run监控脚本没有新滚动日志生成的话则HA部署正常

# HA切换测试

1.
2.
3. 1. **主动切换**
		1. **switchover**

[omm@node1 ~]$ gs\_om -t status –detail 记录集群状态

[ Cluster State ]

cluster\_state : Normal

redistributing : No

current\_az : AZ\_ALL

[ Datanode State ]

node node\_ip instance state

----------------------------------------------------------------------------------------

1 node1 192.168.122.101 6001 /opt/gaussdb/data/db1 P Primary Normal

2 node2 192.168.122.102 6002 /opt/gaussdb/data/db1 S Standby Normal

用omm用户在备库(node2)上执行switchover切换

[omm@node2 ~]$ ./venv/bin/python3 MogDBHA/Switch.py switchover



查询VIP状态



* + 1. **failover**

[omm@node2 ~]$ gs\_om -t status –detail 记录集群状态



用omm用户在备库(node1)上执行failover切换

[omm@node1 ~]$ ./venv/bin/python3 MogDBHA/Switch.py failover

[omm@node1 ~]$ gs\_om -t status –detail 记录集群状态



随后非法存活的老主库(node2)被杀掉



vip切换正常



恢复：

在老主库(node2)执行build重建为新备库加入集群

[omm@node2 ~]$ gs\_ctl -b full build -D /opt/gaussdb/data/db1



查询集群状态，发现恢复正常

[omm@node1 ~]$ gs\_om -t status --detail

[ Cluster State ]

cluster\_state : Normal

redistributing : No

current\_az : AZ\_ALL

[ Datanode State ]

node node\_ip instance state

----------------------------------------------------------------------------------------

1 node1 192.168.122.101 6001 /opt/gaussdb/data/db1 P Primary Normal

2 node2 192.168.122.102 6002 /opt/gaussdb/data/db1 S Standby Normal

* 1. **被动切换**
		1. **对实例状态敏感**

主库正常(手动关闭)/异常(宕机、kill进程)关闭

相关参数

vi MogDBHA/NodeConfig.py

CHECK\_INSTANCE\_RUNNING = True



主备的ha脚本NodeConfig.py文件都要修改，修改完毕重启主备node\_run脚本

本例测试为手动关闭数据库

[omm@node1 ~]$ gs\_om -t status –detail 记录集群状态

[omm@node1 ~]$ gs\_om -t status --detail

[ Cluster State ]

cluster\_state : Normal

redistributing : No

current\_az : AZ\_ALL

[ Datanode State ]

node node\_ip instance state

----------------------------------------------------------------------------------------

1 node1 192.168.122.101 6001 /opt/gaussdb/data/db1 P Primary Normal

2 node2 192.168.122.102 6002 /opt/gaussdb/data/db1 S Standby Normal

手动关闭主库(node1)

[omm@node1 ~]$ gs\_ctl stop -D /opt/gaussdb/data/db1

查询集群状态，观察到备库成为新主库，vip飘到新主库



恢复：

以新备库身份启动老主库

[omm@node1 ~]$ gs\_ctl start -D /opt/gaussdb/data/db1 -M standby



查看集群状态，发现恢复正常

[omm@node1 ~]$ gs\_om -t status --detail

[ Cluster State ]

cluster\_state : Normal

redistributing : No

current\_az : AZ\_ALL

[ Datanode State ]

node node\_ip instance state

----------------------------------------------------------------------------------------

1 node1 192.168.122.101 6001 /opt/gaussdb/data/db1 P Standby Normal

2 node2 192.168.122.102 6002 /opt/gaussdb/data/db1 S Primary Normal

* + 1. **对实例状态不敏感**

主库正常(手动关闭)/异常(宕机、kill进程)关闭

相关参数

vi MogDBHA/NodeConfig.py

CHECK\_INSTANCE\_RUNNING = False



主备的ha脚本NodeConfig.py文件都要修改，修改完毕重启主备node\_run脚本

本例测试为kill掉数据库进程

[omm@node1 ~]$ gs\_om -t status –detail 记录集群状态

[ Cluster State ]

cluster\_state : Normal

redistributing : No

current\_az : AZ\_ALL

[ Datanode State ]

node node\_ip instance state

----------------------------------------------------------------------------------------

1 node1 192.168.122.101 6001 /opt/gaussdb/data/db1 P Primary Normal

2 node2 192.168.122.102 6002 /opt/gaussdb/data/db1 S Standby Normal

查询数据库进程ID并kill掉进程



查询集群状态，发现并没有发生切换，备库处于disconnected状态



vip依旧在主库上



恢复：

启动主库

[omm@node1 ~]$ gs\_ctl start -D /opt/gaussdb/data/db1 -M primary

查询集群状态，发现恢复正常

[omm@node1 ~]$ gs\_om -t status --detail

[ Cluster State ]

cluster\_state : Normal

redistributing : No

current\_az : AZ\_ALL

[ Datanode State ]

node node\_ip instance state

----------------------------------------------------------------------------------------

1 node1 192.168.122.101 6001 /opt/gaussdb/data/db1 P Primary Normal

2 node2 192.168.122.102 6002 /opt/gaussdb/data/db1 S Standby Normal

# 卸载MogDB

1.
2.
3.
4. 1. **执行卸载**

MogDB提供了卸载脚本帮助用户完整的卸载MogDB。

操作步骤

以操作系统用户omm登录数据库主节点。

使用gs\_uninstall卸载MogDB。

gs\_uninstall --delete-data

或者在MogDB中每个节点执行本地卸载。

gs\_uninstall --delete-data -L

**示例**

使用gs\_uninstall脚本进行卸载MogDB。

gs\_uninstall --delete-data

Checking uninstallation.

Successfully checked uninstallation.

Stopping the cluster.

Successfully stopped the cluster.

Successfully deleted instances.

Uninstalling application.

Successfully uninstalled application.

Uninstallation succeeded.

单机卸载场景使用gs\_uninstall脚本进行卸载。

gs\_uninstall --delete-data

Checking uninstallation.

Successfully checked uninstallation.

Stopping the cluster.

Successfully stopped the cluster.

Successfully deleted instances.

Uninstalling application.

Successfully uninstalled application.

Uninstallation succeeded.

* 1. **一键式清理环境**

在MogDB卸载完成后，如果不需要在环境上重新部署MogDB，可以运行脚本gs\_postuninstall对MogDB服务器上环境信息做清理。MogDB环境清理是对环境准备脚本gs\_preinstall所做设置的清理。

**前提条件**

MogDB卸载执行成功。

root用户互信可用。

只能使用root用户执行gs\_postuninstall命令。

/opt/software/MogDB/script/gs\_postuninstall -U omm -X/opt/software/MogDB/clusterconfig.xml --delete-user

**示例**

清理主机的环境

Parsing the configuration file.

Successfully parsed the configuration file.

Check log file path.

Successfully checked log file path.

Checking unpreinstallation.

Successfully checked unpreinstallation.

Deleting Cgroup.

Successfully deleted Cgroup.

Deleting the instance's directory.

Successfully deleted the instance's directory.

Deleting the installation directory.

Successfully deleted the installation directory.

Deleting the temporary directory.

Successfully deleted the temporary directory.

Deleting remote OS user.

Successfully deleted remote OS user.

Deleting software packages and environmental variables of other nodes.

Successfully deleted software packages and environmental variables of other nodes.

Deleting logs of other nodes.

Successfully deleted logs of other nodes.

Deleting software packages and environmental variables of the local node.

Successfully deleted software packages and environmental variables of the local nodes.

Deleting local OS user.

Successfully deleted local OS user.

Deleting local node's logs.

Successfully deleted local node's logs.

Successfully cleaned environment.