

---

# LightDB-A Install Manual

发行版本 23.1

LightDB

2023 年 07 月 06 日

## 目录:

<b>1</b>	<b>前言</b>	<b>2</b>
<b>2</b>	<b>安装前准备</b>	<b>2</b>
2.1	创建用户	2
2.2	防火墙配置	2
2.3	关闭 SELINUX	2
2.4	部署安装包	2
2.5	安装依赖包	3
2.6	配置 hostname	3
2.7	配置免密	4
<b>3</b>	<b>创建数据库实例</b>	<b>4</b>
3.1	实例目录	5
3.2	准备实例化配置文件	5
3.3	准备实例化 host 文件	6
3.4	创建实例	6
<b>4</b>	<b>segment 扩容</b>	<b>6</b>
4.1	准备实例目录	7
4.2	准备扩容 host 文件	7
4.3	生成扩容配置文件	7
4.4	修改扩容配置文件	8
4.5	执行扩容	8
4.6	数据重分布	8
<b>5</b>	<b>安装 LightDB-A 客户端</b>	<b>9</b>
5.1	命令行安装	9

## 1 前言

本文档说明 LightDB-A 安装步骤

## 2 安装前准备

以下操作在每台机器上进行

### 2.1 创建用户

创建 lightdb 用户与用户组

```
groupadd lightadb
useradd -g lightadb -m lightadb
passwd lightadb
```

为 lightadb 用户设置 sudo 免密，在 /etc/sudoers 中新增一行 lightadb ALL=(ALL) NOPASSWD:ALL

### 2.2 防火墙配置

集群所有机器之间需要相互访问，并且数据库需要有对外的服务端口。

如果您的环境可以关闭防火墙，则关闭防火墙软件。

例如：

```
systemctl stop firewalld.service
systemctl disable firewalld.service
```

如果不能关闭防火墙软件，则需要正确开放相关服务端口（具体端口见后面章节）。

### 2.3 关闭 SELINUX

```
sed -i "s/SELINUX=enforcing/SELINUX=disabled/g" /etc/selinux/config
setenforce 0
```

### 2.4 部署安装包

获取 LightDB-A 安装包，上传到每台服务器上，解压到相同目录下。

注意：每台服务器的安装包解压目录要相同。

解压完成后，配置 LightDB-A 的环境变量：

```
source <安装目录>/lightadb_path.sh
```

这个命令建议配置在 ~/.bashrc 文件中，方便以后使用。

## 2.5 安装依赖包

```
sudo yum install -y epel-release
sudo yum install -y \
    apr \
    bzip2 \
    gcc \
    gcc-c++ \
    krb5 \
    libcurl \
    libevent \
    libkadm5 \
    libyaml \
    libxml2 \
    libzstd \
    openssl \
    perl-ExtUtils-Embed \
    python3 \
    python3-pip \
    python3-devel \
    readline \
    xerces-c \
    zlib \
    rsync

# 创建一个python依赖文件，用于后续安装python依赖
tee python-dependencies.txt <<-'EOF'
psutil==5.7.0
pygresql==5.2
pyyaml==5.3.1
EOF

pip3 install -r python-dependencies.txt --user

# 如果需要指定python镜像源，则可以使用如下命令(注意替换成您可以访问的合适的镜像源)
pip3 install -r python-dependencies.txt --user \
    -i http://yum.hundsun.com/mirrors/pypi/simple/ \
    --trusted-host yum.hundsun.com
```

## 2.6 配置 hostname

在 `/etc/hosts` 中配置所有服务器的 ip 和主机名称，例如：

```
cat /etc/hosts

10.19.36.11    host11
10.19.36.10    host10
10.20.30.193  host193
```

这个也是在每台机器上配置，这样每台主机就有一个主机名。

为主机设置设置正确的 hostname，注意和 `/etc/hosts` 中保持一致

例如：

```
# 在 10.19.36.11
hostnamectl set-hostname host11

# 在 10.19.36.10
hostnamectl set-hostname host10

# 在 10.20.30.193
hostnamectl set-hostname host193
```

## 2.7 配置免密

然后需要配置所有机器两两 ssh 免密 (注意: 每台主机对自己也要配置免密)

以下是操作样例

1. 在 11 上执行对其他机器执行 ssh-copy-id 命令, 使得 11 可以免密登录到其他机器上。

```
ssh-copy-id host11
ssh-copy-id host10
ssh-copy-id host193
```

2. 创建一个 hostfile\_exkeys 文件, 填入所有主机名

```
cat hostfile_exkeys

host11
host10
host193
```

3. 使用 gpssh-exkeys 命令配置两两免密

```
gpssh-exkeys -f hostfile_exkeys
```

确认免密是否配置成功, 可以每台机器上执行如下命令, 如果不提示输密码, 则表示配置成功

```
gpssh -f hostfile_exkeys -e "ls"
```

上面 gpssh 命令也是 LightDB-A 提供的批量执行的便利性工具, 可以在所有节点上同时执行指定的 shell 命令。

## 3 创建数据库实例

以下部署以 1 个 coordinator 节点, 4 个 segment 节点为例, 其中 coordinator 节点在 host10 上, host11 和 host193 各有两个 segment 节点。

## 3.1 实例目录

需要在每台机器上为 LightDB-A 准备数据磁盘空间，并确保 lightadb 用户有对应目录的读写权限。

在 host10 上创建 coordinator 节点实例目录

```
mkdir -p /data/lightadb/coordinator
```

在所有 segment 主机上创建 segment 实例目录，host11 和 host193 上创建 segment 节点实例目录

```
mkdir -p /data/lightadb/primary1  
mkdir -p /data/lightadb/primary2
```

## 3.2 准备实例化配置文件

配置文件模板为 < 安装目录>/docs/cli\_help/gpconfigs/gpinitssystem\_config，拷贝一份出来做必要配置。

必须修改的配置项如下：

### 1. PORT\_BASE

segment 实例的端口，每个 segment 会递增分配，如果配置 6000，则会依次分配 6000,6001,...，注意检查下服务器上端口是否有被占用。

应用程序是不直接访问这些端口的。但是集群之间需要相互访问，如果开启了防火墙，则需要在集群机器之间开放这些端口。

### 2. DATA\_DIRECTORY

segment 节点的实例的目录列表，注意，这里配置几个目录，就表示在每个主机上安装几个 segment。

例如，如下所示，DATA\_DIRECTORY 指定了两个目录，表示每个主机需要安装 2 个 segment。假如 segment 主机有 2 个 (后面通过 host 文件指定)，则总共需要安装 4 个 segment。

注意所有目录都要先创建好，并且是空目录。

```
declare -a DATA_DIRECTORY=(/data/lightadb/primary1 /data/lightadb/primary2)
```

### 3. COORDINATOR\_HOSTNAME

表示 coordinator 节点所在的主机名，按前面的例子，需要配置成 host10。

注意 coordinator 可以和 segment 在同一台机器上，也可以分开。

### 4. COORDINATOR\_DIRECTORY

coordinator 节点的实例路径，注意需要先创建好，并且是空目录。

### 5. COORDINATOR\_PORT

coordinator 节点的端口，这个端口是对外服务的，安装成功后，应用程序需要连这个端口来访问数据库。

要确认下 coordinator 主机上这个端口是否被占用。

这个端口是对外服务的，如果开启了防火墙，则需要正确配置，以便应用程序可以访问。

### 3.3 准备实例化 host 文件

host 文件很简单，仅包含 segment 主机列表即可，例如，创建一个 hostfile\_gpinitssystem 文件，内容如下：

```
cat hostfile_gpinitssystem  
  
host11  
host193
```

这里的主机数量乘以 gpinitssystem\_config 中的 DATA\_DIRECTORY 的目录数量，就是 segment 的总数量。

### 3.4 创建实例

使用如下命令创建实例

```
gpinitssystem -c gpinitssystem_config -h hostfile_gpinitssystem
```

安装成功后，会有类似如下输出

```
-LightDB-A Database instance successfully created  
-----  
-To complete the environment configuration, please  
-update lightadb .bashrc file with the following  
-1. Ensure that the lightadb_path.sh file is sourced  
-2. Add "export COORDINATOR_DATA_DIRECTORY=/data/lightadb/zhj/coordinator/gpseg-1"  
- to access the LightDB-A scripts for this instance:  
- or, use -d /data/lightadb/zhj/coordinator/gpseg-1 option for the LightDB-A scripts  
- Example gpstate -d /data/lightadb/zhj/coordinator/gpseg-1  
-Script log file = /home/lightadb/ltaAdminLogs/gpinitssystem_20230322.log  
-To remove instance, run gpdeletesystem utility  
-To initialize a Standby Coordinator Segment for this LightDB-A instance  
-Review options for gpinitstandby
```

按照上面的提示配置环境变量，可以直接配置到 ~/.bashrc 中：

```
# 前面成功消息中有提示  
export COORDINATOR_DATA_DIRECTORY=/data/lightadb/zhj/coordinator/gpseg-1  
  
# COORDINATOR_PORT是前面gpinitssystem_config文件中配置的  
export PGPORT=<COORDINATOR_PORT>
```

## 4 segment 扩容

上面章节的安装方法要求每个主机的 segment 数量一致。如果 segment 主机配置相差较大，希望在不同主机上安装不同数量的 segment，则可以按照本章节方法。则可以先按照前面章节的标准方法安装一个最小实例，然后再扩容。

以下是操作样例：

在 host10,host11,host193,host199 上各有 4 个 segment，总共是 16 个 segment，coordinator 在 host11 上。现在计划在 host11，host10 上各新增 4 个 segment，host199，host193 上不变，扩容完成后总共有 24 个 segment。

## 4.1 准备实例目录

在 host10,host11 上面各新增 4 个目录，用于安装新的 segment

```
mkdir -p /data/gpadmin/lightdba/data5
mkdir -p /data/gpadmin/lightdba/data6
mkdir -p /data/gpadmin/lightdba/data7
mkdir -p /data/gpadmin/lightdba/data8
```

## 4.2 准备扩容 host 文件

host 文件很简单，仅包含要扩容的主机列表即可，例如，创建一个 hostfile\_expand 文件，内容如下：

```
cat hostfile_expand

host11
host10
```

## 4.3 生成扩容配置文件

```
gpexpand -f hostfile_expand
```

这个是交互式的程序，主要输入要扩容 4 个节点，然后再输入 4 个路径 (前面创建好的)。

```
How many new primary segments per host do you want to add? (default=0):
> 4
Enter new primary data directory 1:
> /data/gpadmin/lightdba/data5
Enter new primary data directory 2:
> /data/gpadmin/lightdba/data6
Enter new primary data directory 3:
> /data/gpadmin/lightdba/data7
Enter new primary data directory 4:
> /data/gpadmin/lightdba/data8

Generating configuration file...

20230322:20:38:33:3439364 gpexpand:hs-10-19-36-11:gpadmin-[INFO]:-Generating input_
↪file...

Input configuration file was written to 'gpexpand_inputfile_20230322_203833'.

Please review the file and make sure that it is correct then re-run
with: gpexpand -i gpexpand_inputfile_20230322_203833
```

此时会生成一个扩容配置文件

## 4.4 修改扩容配置文件

默认情况下，所有节点都要扩容成数量一样的 segment，比如我们前面输入 4 的含义是每个主机都扩容 4 个 segment(包含未指定的 host193,host199)。

所以我们需要修改修改配置文件，去掉不必要的节点。仅保留 11 和 10 的配置

```
hs-10-19-36-10|hs-10-19-36-10|49005|/data/gpadmin/lightdba/data5/gpseg16|18|16|p
hs-10-19-36-10|hs-10-19-36-10|49006|/data/gpadmin/lightdba/data6/gpseg17|19|17|p
hs-10-19-36-10|hs-10-19-36-10|49007|/data/gpadmin/lightdba/data7/gpseg18|20|18|p
hs-10-19-36-10|hs-10-19-36-10|49008|/data/gpadmin/lightdba/data8/gpseg19|21|19|p
hs-10-19-36-11|hs-10-19-36-11|49005|/data/gpadmin/lightdba/data5/gpseg20|22|20|p
hs-10-19-36-11|hs-10-19-36-11|49006|/data/gpadmin/lightdba/data6/gpseg21|23|21|p
hs-10-19-36-11|hs-10-19-36-11|49007|/data/gpadmin/lightdba/data7/gpseg22|24|22|p
hs-10-19-36-11|hs-10-19-36-11|49008|/data/gpadmin/lightdba/data8/gpseg23|25|23|p
```

## 4.5 执行扩容

```
gpexpand -i gpexpand_inputfile_20230322_203833
```

如果出现失败，则可以执行 `gpexpand -r` 命令回滚。

## 4.6 数据重分布

如果扩容前已经有数据，则可以执行重分布操作。这个操作会锁表，需要在业务低峰期进行。

```
# 开始数据重分布
gpexpand
```

开始后重分布后，可以通如下 `gpstate -x` 查询重分布进度:

```
Obtaining Segment details from coordinator...
Cluster Expansion State = Data Distribution - Active
-----
Number of tables to be redistributed
  Database      Count of Tables to redistribute
  tpch           76
  postgres      12
  pgbench        6
  hs              8
  jdbc           4
  template1      2
  metrics        2
-----
Active redistributions = 1
  Action          Database      Table
  Redistribute    tpch          public.part
-----
```

或者通过表来查看

```
SELECT * FROM gpexpand.expansion_progress;
SELECT * FROM gpexpand.status_detail order by status;
```

重分布完成后，通过 `gpexpand -c` 清理掉扩容 schema。



## 5 安装 LightDB-A 客户端

### 5.1 命令行安装

- 目前 LightDB-A 数据库服务端只有 CentOS7 版本，所以 LightDB-A 的客户端也只有 CentOS7 版本。
- 将 LightDB-A client 安装包拷贝到服务器的安装目录下，对于本文档演示所用的范例来说安装包是 lightdb-a-client-23.2-12178-el7.x86\_64.zip，安装目录是 /root。

```
[root@localhost ~]# ls
anaconda-ks.cfg Desktop Documents Downloads lightdb-a-client-23.2-12178-el7.
↪x86_64.zip Music original-ks.cfg Pictures Public Templates Videos
[root@localhost ~]#
[root@localhost ~]# pwd
/root
[root@localhost ~]#
```

- 解压 LightDB-A client 安装包。

```
[root@localhost ~]# unzip lightdb-a-client-23.2-12178-el7.x86_64.zip
Archive: lightdb-a-client-23.2-12178-el7.x86_64.zip
  creating: lightdb-a-client-23.2-12178-el7.x86_64/
  creating: lightdb-a-client-23.2-12178-el7.x86_64/bin/
 inflating: lightdb-a-client-23.2-12178-el7.x86_64/bin/psql
 inflating: lightdb-a-client-23.2-12178-el7.x86_64/bin/pg_restore
 inflating: lightdb-a-client-23.2-12178-el7.x86_64/bin/pg_dumpall
 inflating: lightdb-a-client-23.2-12178-el7.x86_64/bin/pg_dump
 inflating: lightdb-a-client-23.2-12178-el7.x86_64/bin/gpfdist
 inflating: lightdb-a-client-23.2-12178-el7.x86_64/bin/gpload.py
 inflating: lightdb-a-client-23.2-12178-el7.x86_64/bin/pg_config
  creating: lightdb-a-client-23.2-12178-el7.x86_64/lib/
   linking: lightdb-a-client-23.2-12178-el7.x86_64/lib/libpq.so -> libpq.so.5.12
   linking: lightdb-a-client-23.2-12178-el7.x86_64/lib/libpq.so.5 -> libpq.so.5.
↪12
 inflating: lightdb-a-client-23.2-12178-el7.x86_64/lib/libpq.so.5.12
```

- cd 到 LightDB-A client 解压后生成的目录中，本例中就是 lightdb-a-client-23.2-12178-el7.x86\_64 目录，bin 目录中存放的就是客户能够使用的脚本和可执行文件，lib 目录存放的是被依赖的 so 文件，include 是依赖的头文件。

```
[root@localhost ~]# ls lightdb-a-client-23.2-12178-el7.x86_64
bin include lib
[root@localhost ~]#
```

- 配置环境变量。

```
export GPHOME=/root/lightdb-a-client-23.2-12178-el7.x86_64
export PATH=${GPHOME}/bin:${PATH}
export LD_LIBRARY_PATH=${GPHOME}/lib:${LD_LIBRARY_PATH}
export PYTHONPATH=${GPHOME}/lib/python:${PYTHONPATH}
```

- 安装依赖包。

```
sudo yum install -y epel-release
sudo yum install -y python3 python3-pip python3-devel
sudo yum install -y apr libzstd
```

(续下页)

(接上页)

```
tee python-dependencies.txt <<-'EOF'  
psutil==5.7.0  
pygresql==5.2  
pyyaml==5.3.1  
EOF  
  
pip3 install -r python-dependencies.txt --user \  
-i http://yum.hundsun.com/mirrors/pypi/simple/ \  
--trusted-host yum.hundsun.com
```

- 以上步骤都执行成功，则 LightDB-A client 安装成功。