麦聪 DaaS 平台

安装文档

版本 ：3.6.1

麦聪软件

目录

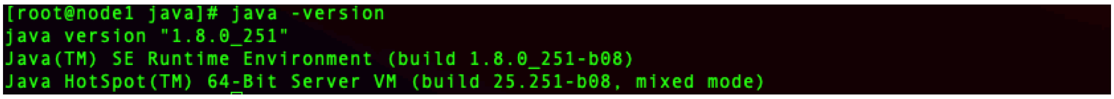
1. 环境检查

检查系统是否已经安装配置 JDK 1.8 以上版本，以及 PostgreSQL 数据库（推荐 PostgreSQL12）。

1.1 检查是否安装 JDK

- 打开终端输入检查 Java 版本,需要 jdk1.8 版本：

  ```
  java -version
  ```



```
[root@node1 java]# java -version
java version "1.8.0_251"
Java(TM) SE Runtime Environment (build 1.8.0_251-b08)
Java HotSpot(TM) 64-Bit Server VM (build 25.251-b08, mixed mode)
```

  查看 java 安装文件：

  ```
  rpm -qa | grep java
  ```

  如果非 jdk1.8 版本删除 java 相关文件.

- 如果未安装 Java 参考 （安装配置 jdk）

1.2 检查 PostgreSQL 安装情况

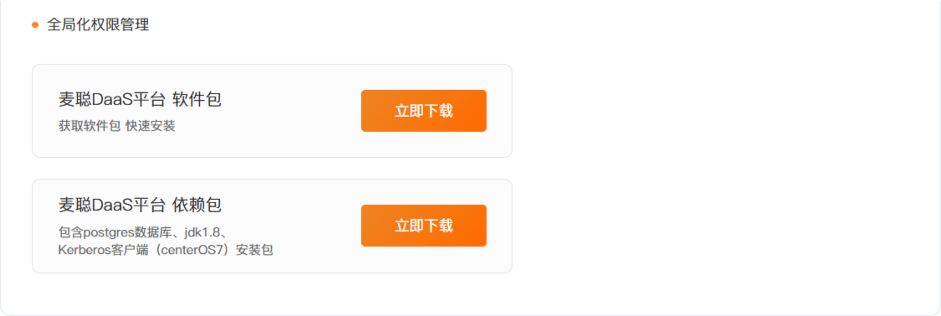
- 检查 Postgre SQL 是否安装成功，支持远程访问。

  如未安装 Postgre SQL 参考（PostgreSQL12 安装指南）

2. 安装麦聪 DaaS 软件

2.1 下载安装包

- 访问下载页面 http://www.maicongs.com/#/home/probation
  选择合适的软件版本，点击下载



- 全局化权限管理

麦聪DaaS平台 软件包

获取软件包 快速安装

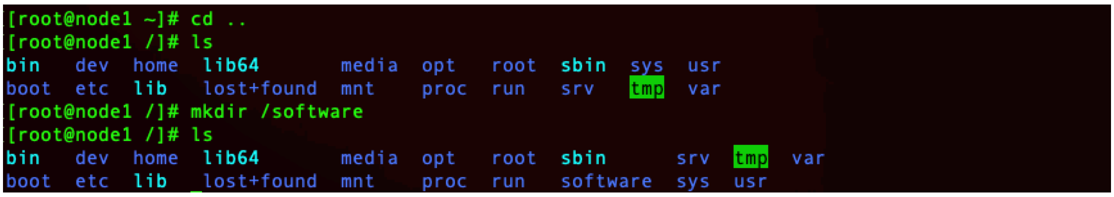
麦聪DaaS平台 依赖包

包含postgres数据库、jdk1.8、Kerberos客户端（centerOS7）安装包

2.2 复制解压缩软件

- 服务器上新建 software 文件夹

```
mkdir /software
```

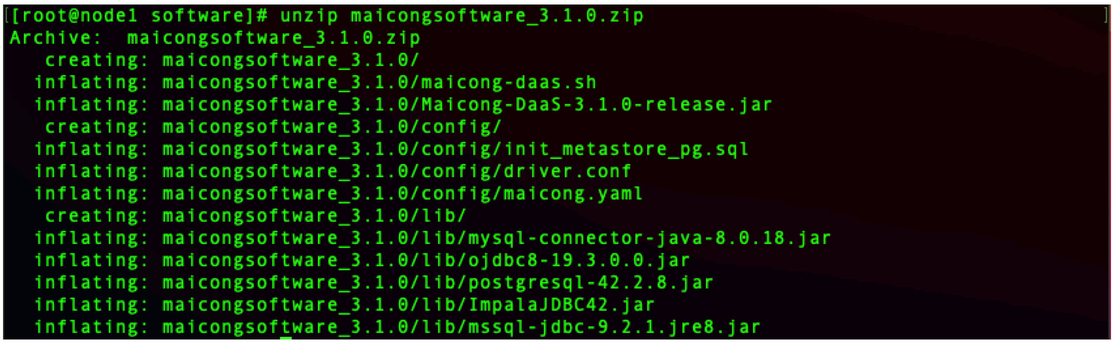


```
[root@node1 ~]# cd ..
[root@node1 /]# ls
bin   dev  home  lib64       media  opt   root  sbin  sys  usr
boot  etc  lib   lost+found  mnt    proc  run   srv   tmp  var
[root@node1 /]# mkdir /software
[root@node1 /]# ls
bin   dev  home  lib64       media  opt   root  sbin      srv  tmp  var
boot  etc  lib   lost+found  mnt    proc  run   software  sys  usr
```

将安装包复制到服务器上 software 文件夹

- 解压

```
unzip maicongsoftware_<实际版本号>.zip
```



```
[root@node1 software]# unzip maicongsoftware_3.1.0.zip
Archive:  maicongsoftware_3.1.0.zip
   creating: maicongsoftware_3.1.0/
  inflating: maicongsoftware_3.1.0/maicong-daas.sh
  inflating: maicongsoftware_3.1.0/Maicong-DaaS-3.1.0-release.jar
   creating: maicongsoftware_3.1.0/config/
  inflating: maicongsoftware_3.1.0/config/init_metastore_pg.sql
  inflating: maicongsoftware_3.1.0/config/driver.conf
  inflating: maicongsoftware_3.1.0/config/maicong.yaml
   creating: maicongsoftware_3.1.0/lib/
  inflating: maicongsoftware_3.1.0/lib/mysql-connector-java-8.0.18.jar
  inflating: maicongsoftware_3.1.0/lib/ojdbc8-19.3.0.0.jar
  inflating: maicongsoftware_3.1.0/lib/postgresql-42.2.8.jar
  inflating: maicongsoftware_3.1.0/lib/ImpalaJDBC42.jar
  inflating: maicongsoftware_3.1.0/lib/mssql-jdbc-9.2.1.jre8.jar
```

> Tip
>
> 若 unzip 未安装，可以按下列命令安装
>
> yum install -y unzip zip

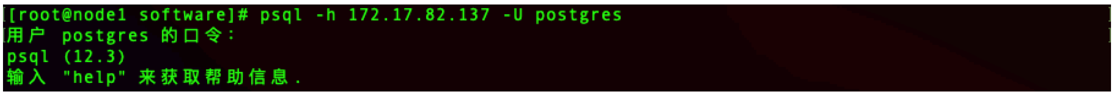
2.3 初始化数据库

以 PostgreSQL 和麦聪 DaaS 平台在同一台物理服务器为例。假设 PostgreSQL12 数据库已经安装完毕。

- 新建数据库：maicong（可用任何名字）

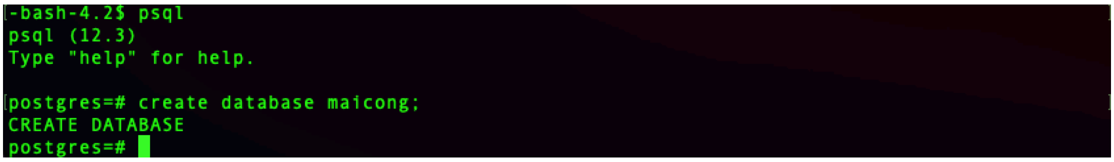
  进入 psql

  `psql -h <本机 IP> -U postgres`



```
[root@node1 software]# psql -h 172.17.82.137 -U postgres
用户 postgres 的口令：
psql (12.3)
输入 "help" 来获取帮助信息.
```

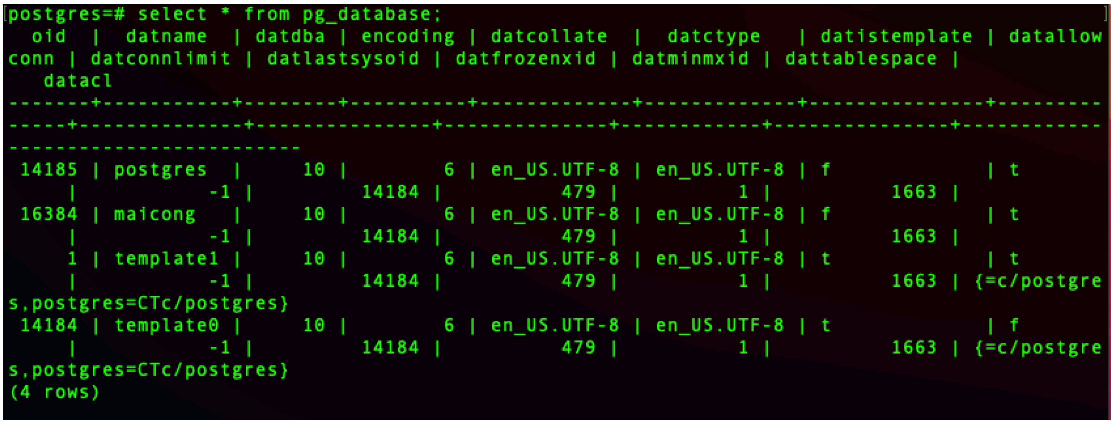
  执行命令：`create database maicong;`



```
-bash-4.2$ psql
psql (12.3)
Type "help" for help.

postgres=# create database maicong;
CREATE DATABASE
postgres=#
```

- 查看已经创建的数据库：

  `select * from pg_database;`



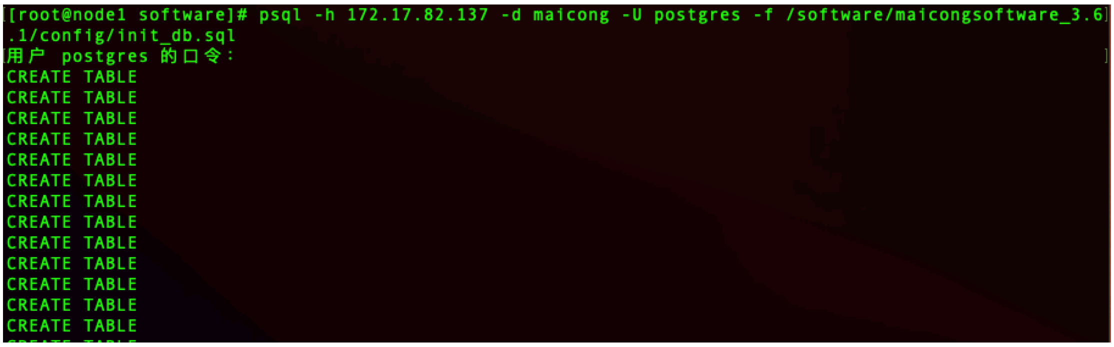
```
postgres=# select * from pg_database;
  oid  |  datname  | datdba | encoding | datcollate  |  datctype   | datistemplate | datallow
conn | datconnlimit | datlastsysoid | datfrozenxid | datminmxid | dattablespace |
   datacl
-------+-----------+--------+----------+-------------+-------------+---------------+---------
-----+--------------+---------------+--------------+------------+---------------+------------
-------------------------
 14185 | postgres  |     10 |        6 | en_US.UTF-8 | en_US.UTF-8 | f             | t
     |           -1 |         14184 |          479 |          1 |          1663 |
 16384 | maicong   |     10 |        6 | en_US.UTF-8 | en_US.UTF-8 | f             | t
     |           -1 |         14184 |          479 |          1 |          1663 |
     1 | template1 |     10 |        6 | en_US.UTF-8 | en_US.UTF-8 | t             | t
     |           -1 |         14184 |          479 |          1 |          1663 | {=c/postgre
s,postgres=CTc/postgres}
 14184 | template0 |     10 |        6 | en_US.UTF-8 | en_US.UTF-8 | t             | f
     |           -1 |         14184 |          479 |          1 |          1663 | {=c/postgre
s,postgres=CTc/postgres}
(4 rows)
```

- 初始化 PostgreSQL 的系统库,SQL 文件在 config/init_db.sql

  (或 init_metastore_pg.sql)

  执行数据库初始化

  命令：`psql -h <本机 IP> -d maicong -U postgres -f /<安装文件解压地址>/config/init_db.sql`



```
[root@node1 software]# psql -h 172.17.82.137 -d maicong -U postgres -f /software/maicongsoftware_3.6
.1/config/init_db.sql
用户 postgres 的口令:
CREATE TABLE
CREATE TABLE
CREATE TABLE
CREATE TABLE
CREATE TABLE
CREATE TABLE
CREATE TABLE
CREATE TABLE
CREATE TABLE
CREATE TABLE
CREATE TABLE
CREATE TABLE
CREATE TABLE
```

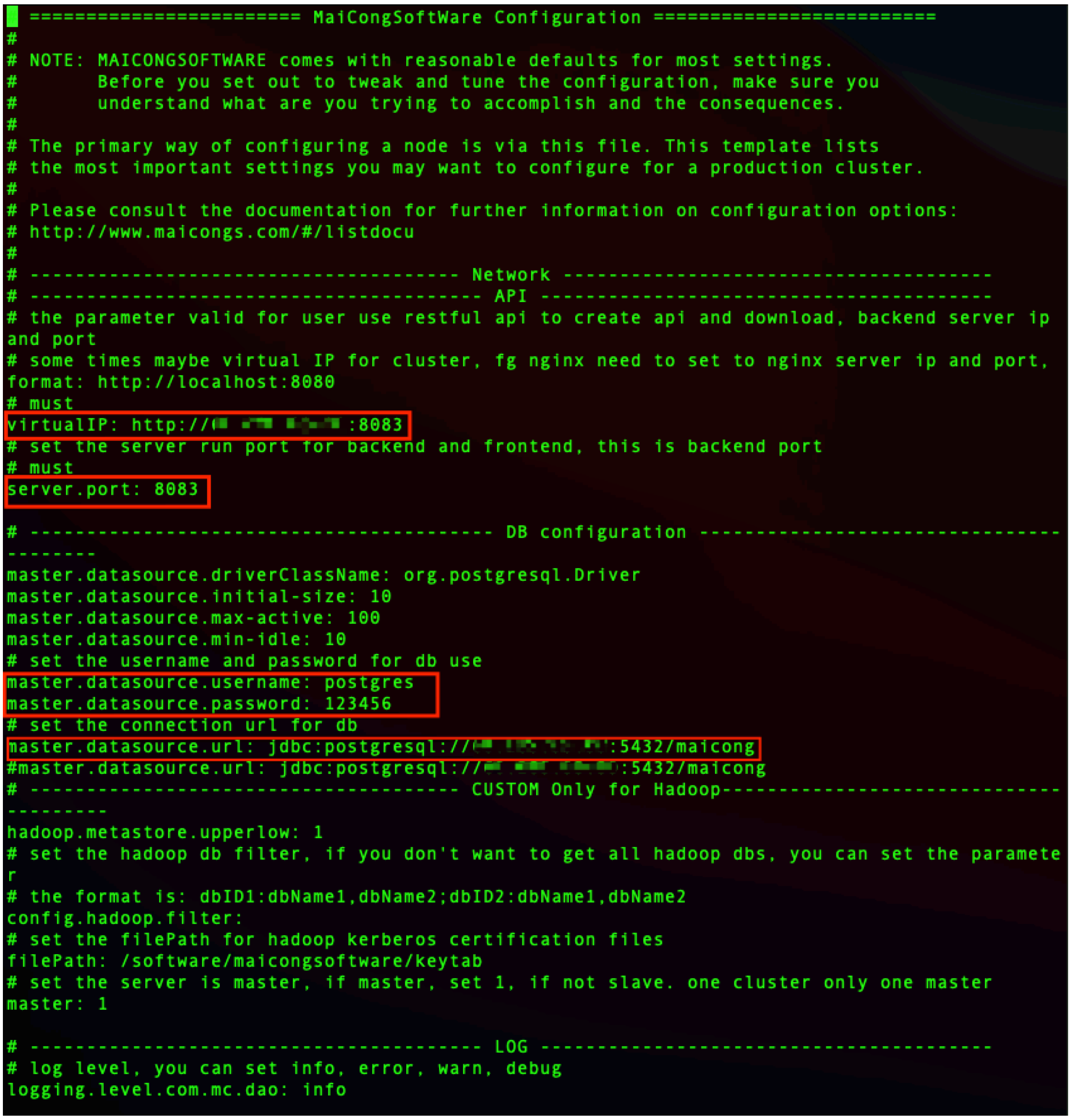
2.4 修改配置文件

2.4.1 修改 config/maicong.yaml 文件

进入 maicongsoftware_<实际版本号>目录，按照示例更新配置文件。

```
vi config/maicong.yaml
```

文件中的冒号“：”后面要有一个英文的空格。



```
# ======================== MaiCongSoftWare Configuration =========================
#
# NOTE: MAICONGSOFTWARE comes with reasonable defaults for most settings.
#       Before you set out to tweak and tune the configuration, make sure you
#       understand what are you trying to accomplish and the consequences.
#
# The primary way of configuring a node is via this file. This template lists
# the most important settings you may want to configure for a production cluster.
#
# Please consult the documentation for further information on configuration options:
# http://www.maicongs.com/#/listdocu
#
# -------------------------------------- Network --------------------------------------
# ---------------------------------------- API ----------------------------------------
# the parameter valid for user use restful api to create api and download, backend server ip
and port
# some times maybe virtual IP for cluster, fg nginx need to set to nginx server ip and port,
format: http://localhost:8080
# must
virtualIP: http://[redacted]:8083
# set the server run port for backend and frontend, this is backend port
# must
server.port: 8083

# ----------------------------------------- DB configuration --------------------------------
--------
master.datasource.driverClassName: org.postgresql.Driver
master.datasource.initial-size: 10
master.datasource.max-active: 100
master.datasource.min-idle: 10
# set the username and password for db use
master.datasource.username: postgres
master.datasource.password: 123456
# set the connection url for db
master.datasource.url: jdbc:postgresql://[redacted]:5432/maicong
#master.datasource.url: jdbc:postgresql://[redacted]:5432/maicong
# -------------------------------------- CUSTOM Only for Hadoop------------------------------
---------
hadoop.metastore.upperlow: 1
# set the hadoop db filter, if you don't want to get all hadoop dbs, you can set the paramete
r
# the format is: dbID1:dbName1,dbName2;dbID2:dbName1,dbName2
config.hadoop.filter:
# set the filePath for hadoop kerberos certification files
filePath: /software/maicongsoftware/keytab
# set the server is master, if master, set 1, if not slave. one cluster only one master
master: 1

# ---------------------------------------- LOG ----------------------------------------
# log level, you can set info, error, warn, debug
logging.level.com.mc.dao: info
```

virtualIP: 服务器地址:端口

server.port: 默认系统启动端口

master.datasource.password: PostgreSQL 的连接密码 (冒号后需带空格)

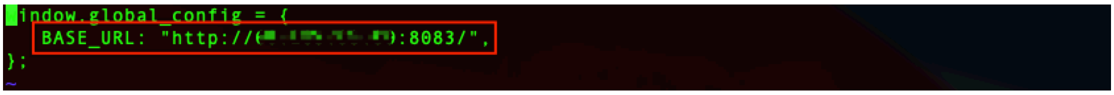
master.datasource.url: PostgreSQL 数据库中相应的连接字符串：有 IP，端口和数据库名称（此处为 maicong，应为初次安装 POSTGRESQL12 步骤创建的数据库名称为"maicong"）

filePath: Hadoop 存入 kerberos keytab 的路径（如果连接 Hadoop Kerberos 需要配置，否则不需要。

2.4.2 修改 static/config.js 文件

BASE_URL= “本机后台地址：端口”

`vi static/config.js`



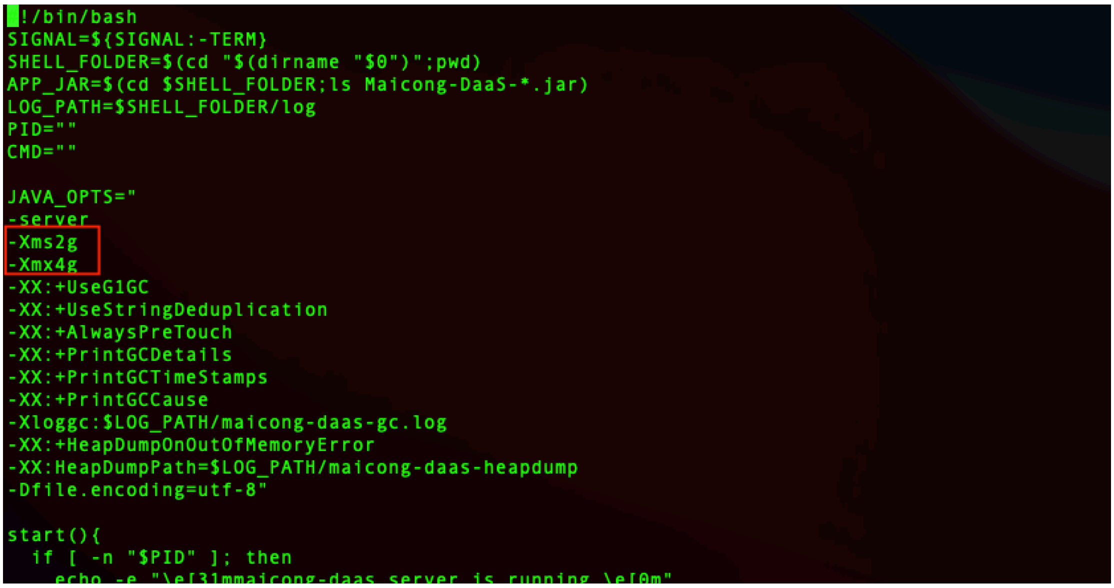
```
indow.global_config = {
  BASE_URL: "http://([redacted]):8083/",
};
~
```

2.5 启动软件

- 添加启动文件 app.sh 的执行权限

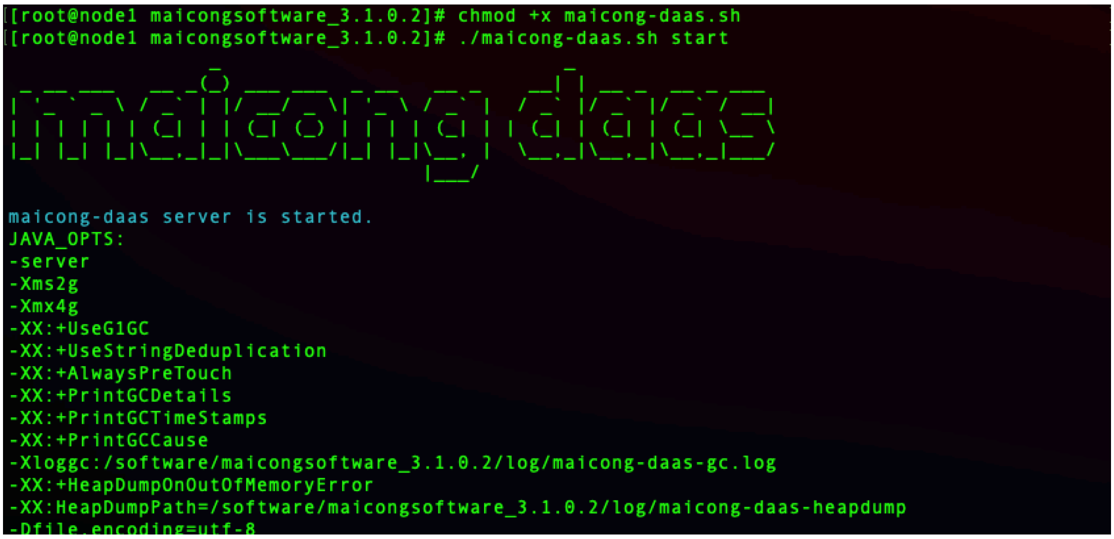
`chmod +x maicong-daas.sh`

- 配置 java 启动内存
  `vi maicong-daas.sh`
  修改-xms 和 -xmx 为启动内存和最大内存（根据实际服务器情况修改）



```
#!/bin/bash
SIGNAL=${SIGNAL:-TERM}
SHELL_FOLDER=$(cd "$(dirname "$0")";pwd)
APP_JAR=$(cd $SHELL_FOLDER;ls Maicong-DaaS-*.jar)
LOG_PATH=$SHELL_FOLDER/log
PID=""
CMD=""

JAVA_OPTS="
-server
-Xms2g
-Xmx4g
-XX:+UseG1GC
-XX:+UseStringDeduplication
-XX:+AlwaysPreTouch
-XX:+PrintGCDetails
-XX:+PrintGCTimeStamps
-XX:+PrintGCCause
-Xloggc:$LOG_PATH/maicong-daas-gc.log
-XX:+HeapDumpOnOutOfMemoryError
-XX:HeapDumpPath=$LOG_PATH/maicong-daas-heapdump
-Dfile.encoding=utf-8"

start(){
  if [ -n "$PID" ]; then
    echo -e "\e[31mmaicong-daas server is running \e[0m"
```

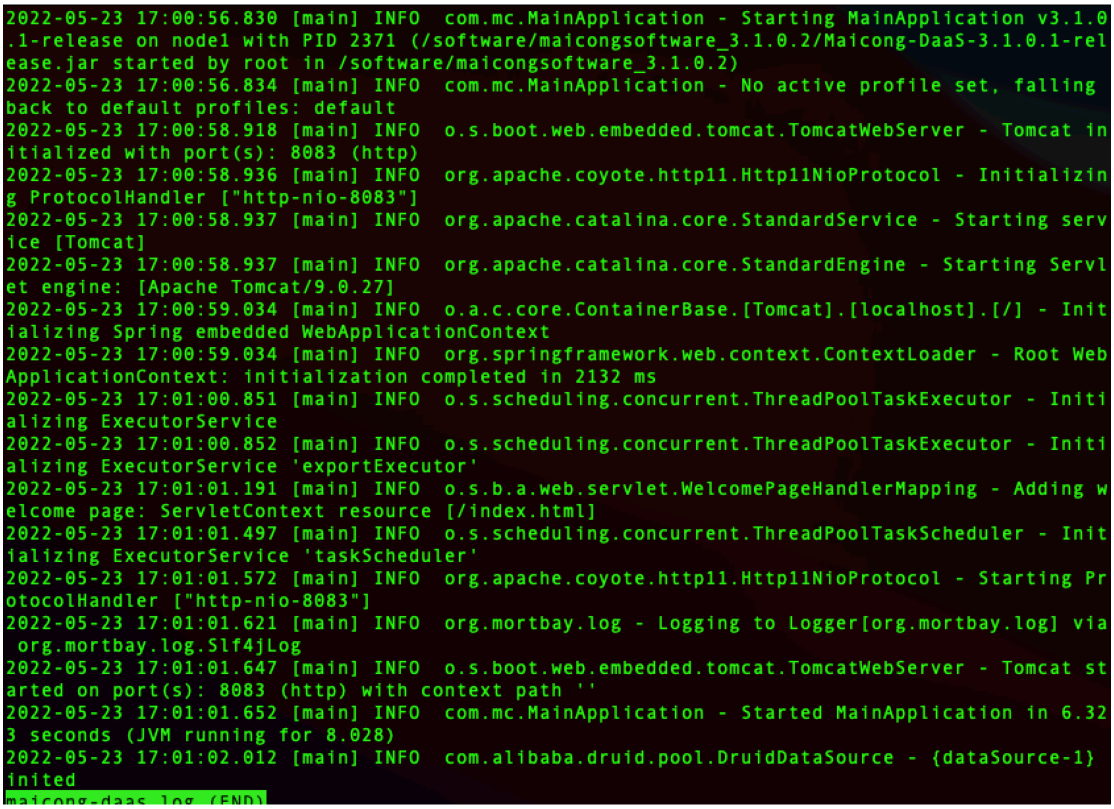
- 启动应用：`./maicong-daas.sh start`



```
[root@node1 maicongsoftware_3.1.0.2]# chmod +x maicong-daas.sh
[root@node1 maicongsoftware_3.1.0.2]# ./maicong-daas.sh start
                 _                                 _
 _ __ ___   __ _(_) ___ ___  _ __   __ _      __| | __ _  __ _ ___
| '_ ` _ \ / _` | |/ __/ _ \| '_ \ / _` |    / _` |/ _` |/ _` / __|
| | | | | | (_| | | (_| (_) | | | | (_| |   | (_| | (_| | (_| \__ \
|_| |_| |_|\__,_|_|\___\___/|_| |_|\__, |    \__,_|\__,_|\__,_|___/
                                   |___/

maicong-daas server is started.
JAVA_OPTS:
-server
-Xms2g
-Xmx4g
-XX:+UseG1GC
-XX:+UseStringDeduplication
-XX:+AlwaysPreTouch
-XX:+PrintGCDetails
-XX:+PrintGCTimeStamps
-XX:+PrintGCCause
-Xloggc:/software/maicongsoftware_3.1.0.2/log/maicong-daas-gc.log
-XX:+HeapDumpOnOutOfMemoryError
-XX:HeapDumpPath=/software/maicongsoftware_3.1.0.2/log/maicong-daas-heapdump
-Dfile.encoding=utf-8
```

> Tip
>
> 系统启动目前需要在 maicong-daas.sh 所在的文件夹下执行。

- 停止应用：`./maicong-daas.sh stop`
- 附:日志文件在 log/maicong-daas-console.log

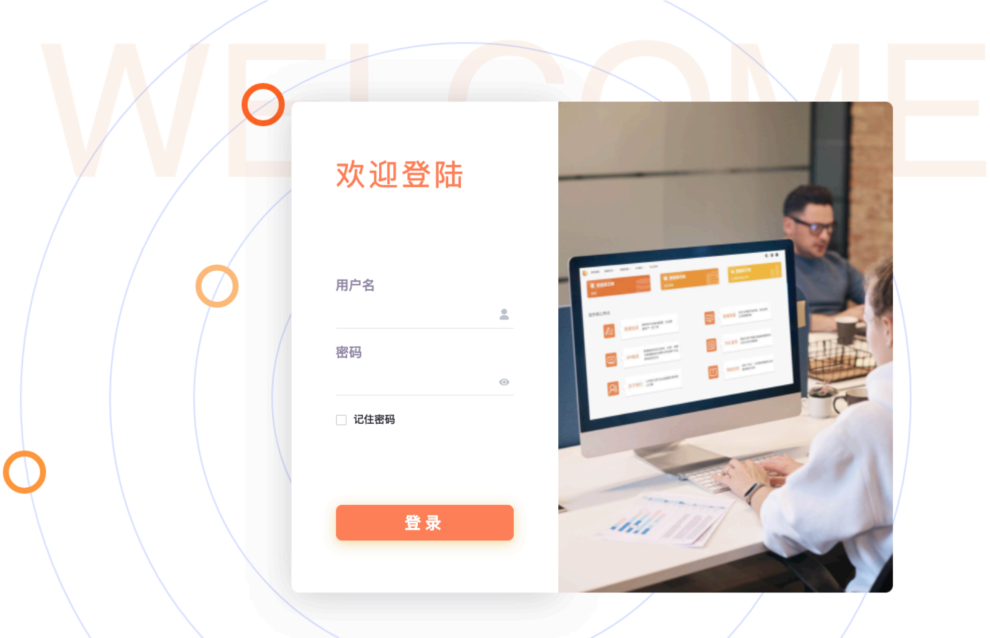


```
2022-05-23 17:00:56.830 [main] INFO  com.mc.MainApplication - Starting MainApplication v3.1.0
.1-release on node1 with PID 2371 (/software/maicongsoftware_3.1.0.2/Maicong-DaaS-3.1.0.1-rel
ease.jar started by root in /software/maicongsoftware_3.1.0.2)
2022-05-23 17:00:56.834 [main] INFO  com.mc.MainApplication - No active profile set, falling
back to default profiles: default
2022-05-23 17:00:58.918 [main] INFO  o.s.boot.web.embedded.tomcat.TomcatWebServer - Tomcat in
itialized with port(s): 8083 (http)
2022-05-23 17:00:58.936 [main] INFO  org.apache.coyote.http11.Http11NioProtocol - Initializin
g ProtocolHandler ["http-nio-8083"]
2022-05-23 17:00:58.937 [main] INFO  org.apache.catalina.core.StandardService - Starting serv
ice [Tomcat]
2022-05-23 17:00:58.937 [main] INFO  org.apache.catalina.core.StandardEngine - Starting Servl
et engine: [Apache Tomcat/9.0.27]
2022-05-23 17:00:59.034 [main] INFO  o.a.c.core.ContainerBase.[Tomcat].[localhost].[/] - Init
ializing Spring embedded WebApplicationContext
2022-05-23 17:00:59.034 [main] INFO  org.springframework.web.context.ContextLoader - Root Web
ApplicationContext: initialization completed in 2132 ms
2022-05-23 17:01:00.851 [main] INFO  o.s.scheduling.concurrent.ThreadPoolTaskExecutor - Initi
alizing ExecutorService
2022-05-23 17:01:00.852 [main] INFO  o.s.scheduling.concurrent.ThreadPoolTaskExecutor - Initi
alizing ExecutorService 'exportExecutor'
2022-05-23 17:01:01.191 [main] INFO  o.s.b.a.web.servlet.WelcomePageHandlerMapping - Adding w
elcome page: ServletContext resource [/index.html]
2022-05-23 17:01:01.497 [main] INFO  o.s.scheduling.concurrent.ThreadPoolTaskScheduler - Init
ializing ExecutorService 'taskScheduler'
2022-05-23 17:01:01.572 [main] INFO  org.apache.coyote.http11.Http11NioProtocol - Starting Pr
otocolHandler ["http-nio-8083"]
2022-05-23 17:01:01.621 [main] INFO  org.mortbay.log - Logging to Logger[org.mortbay.log] via
 org.mortbay.log.Slf4jLog
2022-05-23 17:01:01.647 [main] INFO  o.s.boot.web.embedded.tomcat.TomcatWebServer - Tomcat st
arted on port(s): 8083 (http) with context path ''
2022-05-23 17:01:01.652 [main] INFO  com.mc.MainApplication - Started MainApplication in 6.32
3 seconds (JVM running for 8.028)
2022-05-23 17:01:02.012 [main] INFO  com.alibaba.druid.pool.DruidDataSource - {dataSource-1}
inited
maicong-daas.log (END)
```

2.6 验证安装

- 测试登录

访问 ip:port



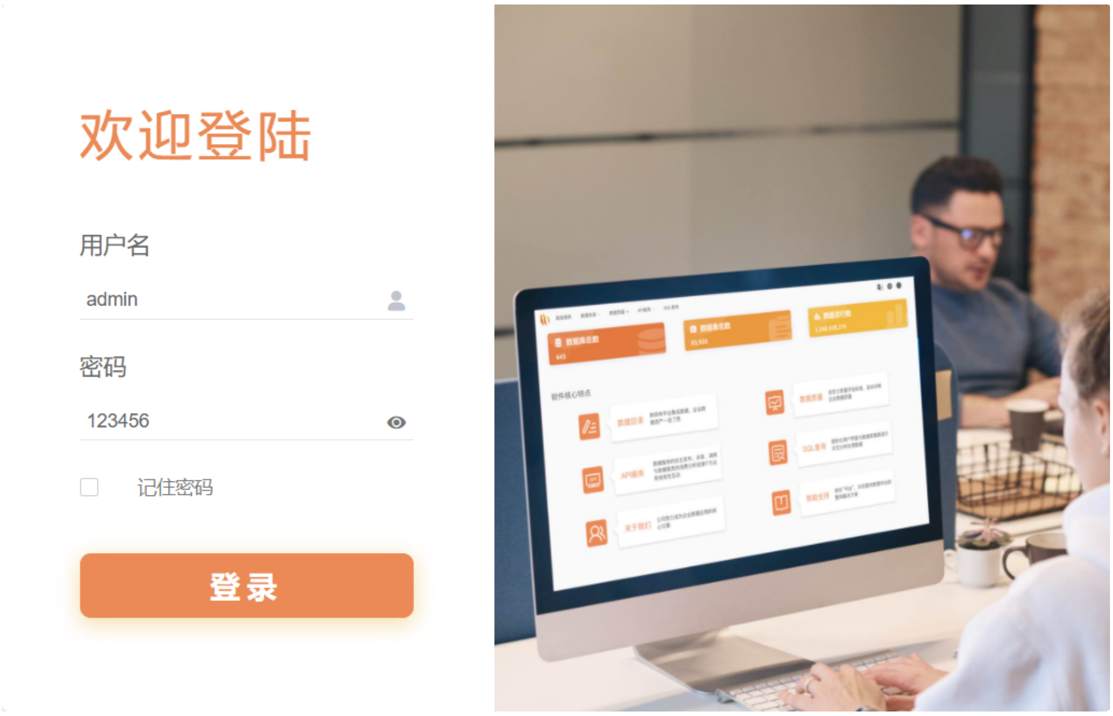
欢迎登陆

用户名

密码

□ 记住密码

登录


欢迎登陆

用户名

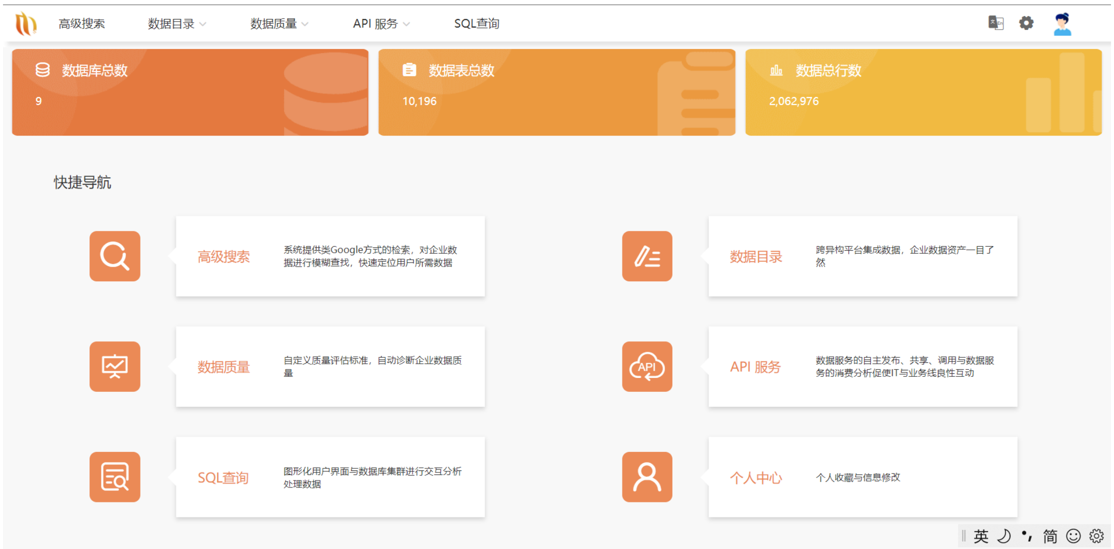
admin

密码

123456

□ 记住密码

显示该界面代表登录成功



数据库总数 9

数据表总数 10,196

数据总行数 2,062,976

快捷导航

高级搜索 系统提供类Google方式的检索，对企业数据进行模糊查找，快速定位用户所需数据

数据目录 跨异构平台集成数据，企业数据资产一目了然

数据质量 自定义质量评估标准，自动诊断企业数据质量

API 服务 数据服务的自主发布、共享、调用与数据服务的消费分析促使IT与业务线良性互动

SQL查询 图形化用户界面与数据库集群进行交互分析处理数据

个人中心 个人收藏与信息修改

输入初始用户名和密码：admin/123456 登陆进行后续配置。

3. 附录

3.1 安装 JDK

- 新建文件夹并复制 jdk 安装包

新建文件夹 /usr/java

```
mkdir /usr/java
```

- 复制 jdk 安装包到/usr/java 目录下

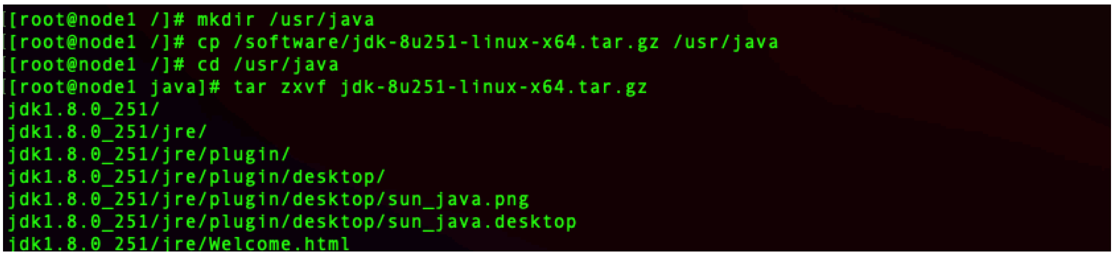
```
cp /software/jdk_8u251_linux_x64.tar.gz /usr/java
```

移动到/usr/java 目录下

```
cd /usr/java/
```

- 解压

```
tar zxvf jdk 8u251 linux x64.tar.gz
```



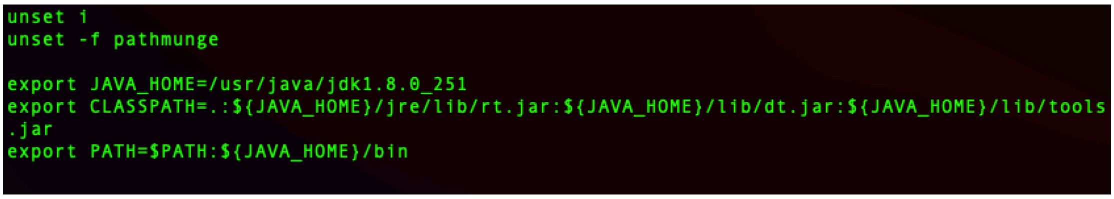
```
[root@node1 /]# mkdir /usr/java
[root@node1 /]# cp /software/jdk-8u251-linux-x64.tar.gz /usr/java
[root@node1 /]# cd /usr/java
[root@node1 java]# tar zxvf jdk-8u251-linux-x64.tar.gz
jdk1.8.0_251/
jdk1.8.0_251/jre/
jdk1.8.0_251/jre/plugin/
jdk1.8.0_251/jre/plugin/desktop/
jdk1.8.0_251/jre/plugin/desktop/sun_java.png
jdk1.8.0_251/jre/plugin/desktop/sun_java.desktop
jdk1.8.0_251/jre/Welcome.html
```

- 配置 java 环境，修改/etc/profile 文件

```
vi /etc/profile
```

在文件末尾添加如下

```
export JAVA_HOME=/usr/java/jdk1.8.0_251
export
CLASSPATH=.:${JAVA_HOME}/jre/lib/rt.jar:${JAVA_HOME}/lib/dt.jar:${JAVA_HOME}/lib/tools.jar
export PATH=$PATH:${JAVA_HOME}/bin
```



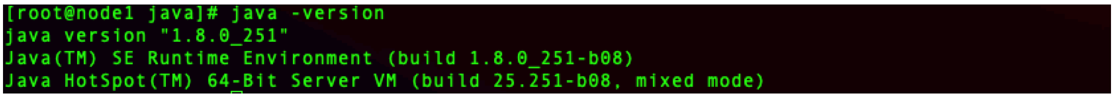
```
unset i
unset -f pathmunge

export JAVA_HOME=/usr/java/jdk1.8.0_251
export CLASSPATH=.:${JAVA_HOME}/jre/lib/rt.jar:${JAVA_HOME}/lib/dt.jar:${JAVA_HOME}/lib/tools
.jar
export PATH=$PATH:${JAVA_HOME}/bin
```

- 使环境变量生效，执行命令如下

```
source /etc/profile
```

- 测试 java 安装效果

```
java -version
```



```
[root@node1 java]# java -version
java version "1.8.0_251"
Java(TM) SE Runtime Environment (build 1.8.0_251-b08)
Java HotSpot(TM) 64-Bit Server VM (build 25.251-b08, mixed mode)
```

若出现上图信息，则代表安装成功。

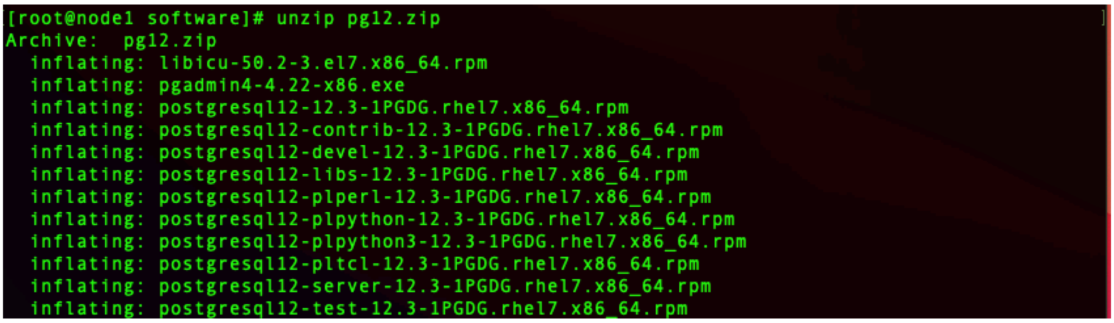
3.2 安装 POSTGRESQL12

3.2.1 安装包准备工作

- 解压 pg12.zip

解压

```
unzip pg12.zip
```



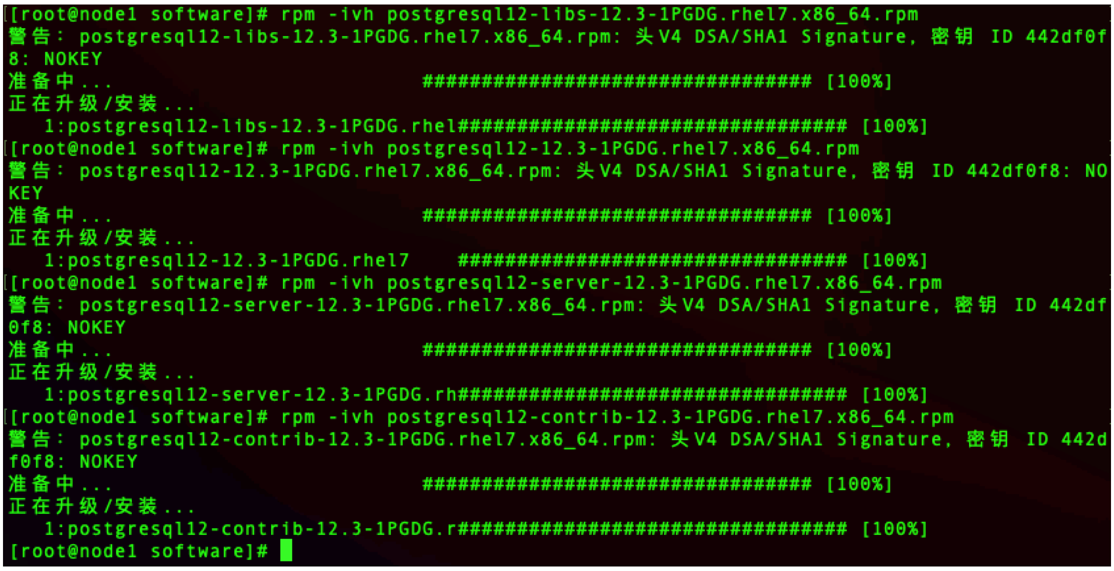
```
[root@node1 software]# unzip pg12.zip
Archive:  pg12.zip
  inflating: libicu-50.2-3.el7.x86_64.rpm
  inflating: pgadmin4-4.22-x86.exe
  inflating: postgresql12-12.3-1PGDG.rhel7.x86_64.rpm
  inflating: postgresql12-contrib-12.3-1PGDG.rhel7.x86_64.rpm
  inflating: postgresql12-devel-12.3-1PGDG.rhel7.x86_64.rpm
  inflating: postgresql12-libs-12.3-1PGDG.rhel7.x86_64.rpm
  inflating: postgresql12-plperl-12.3-1PGDG.rhel7.x86_64.rpm
  inflating: postgresql12-plpython-12.3-1PGDG.rhel7.x86_64.rpm
  inflating: postgresql12-plpython3-12.3-1PGDG.rhel7.x86_64.rpm
  inflating: postgresql12-pltcl-12.3-1PGDG.rhel7.x86_64.rpm
  inflating: postgresql12-server-12.3-1PGDG.rhel7.x86_64.rpm
  inflating: postgresql12-test-12.3-1PGDG.rhel7.x86_64.rpm
```

3.2.2 安装依赖

```
yum -y install libicu
yum -y install libxslt
```

按顺序安装 rpm 包

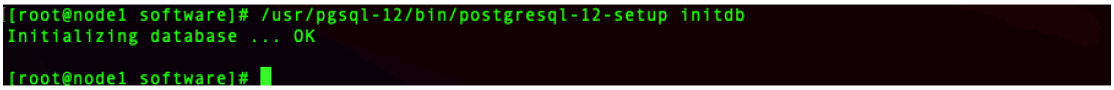
```
rpm -ivh postgresql12-libs-12.3-1PGDG.rhel7.x86_64.rpm
rpm -ivh postgresql12-12.3-1PGDG.rhel7.x86_64.rpm
rpm -ivh postgresql12-server-12.3-1PGDG.rhel7.x86_64.rpm
rpm -ivh postgresql12-contrib-12.3-1PGDG.rhel7.x86_64.rpm
```



```
[root@node1 software]# rpm -ivh postgresql12-libs-12.3-1PGDG.rhel7.x86_64.rpm
警告：postgresql12-libs-12.3-1PGDG.rhel7.x86_64.rpm: 头V4 DSA/SHA1 Signature, 密钥 ID 442df0f
8: NOKEY
准备中...                          ################################# [100%]
正在升级/安装...
   1:postgresql12-libs-12.3-1PGDG.rhel################################# [100%]
[root@node1 software]# rpm -ivh postgresql12-12.3-1PGDG.rhel7.x86_64.rpm
警告：postgresql12-12.3-1PGDG.rhel7.x86_64.rpm: 头V4 DSA/SHA1 Signature, 密钥 ID 442df0f8: NO
KEY
准备中...                          ################################# [100%]
正在升级/安装...
   1:postgresql12-12.3-1PGDG.rhel7    ################################# [100%]
[root@node1 software]# rpm -ivh postgresql12-server-12.3-1PGDG.rhel7.x86_64.rpm
警告：postgresql12-server-12.3-1PGDG.rhel7.x86_64.rpm: 头V4 DSA/SHA1 Signature, 密钥 ID 442df
0f8: NOKEY
准备中...                          ################################# [100%]
正在升级/安装...
   1:postgresql12-server-12.3-1PGDG.rh################################# [100%]
[root@node1 software]# rpm -ivh postgresql12-contrib-12.3-1PGDG.rhel7.x86_64.rpm
警告：postgresql12-contrib-12.3-1PGDG.rhel7.x86_64.rpm: 头V4 DSA/SHA1 Signature, 密钥 ID 442d
f0f8: NOKEY
准备中...                          ################################# [100%]
正在升级/安装...
   1:postgresql12-contrib-12.3-1PGDG.r################################# [100%]
[root@node1 software]#
```

3.2.3 数据库的初始化具体操作

```
/usr/pgsql-12/bin/postgresql-12-setup initdb
```

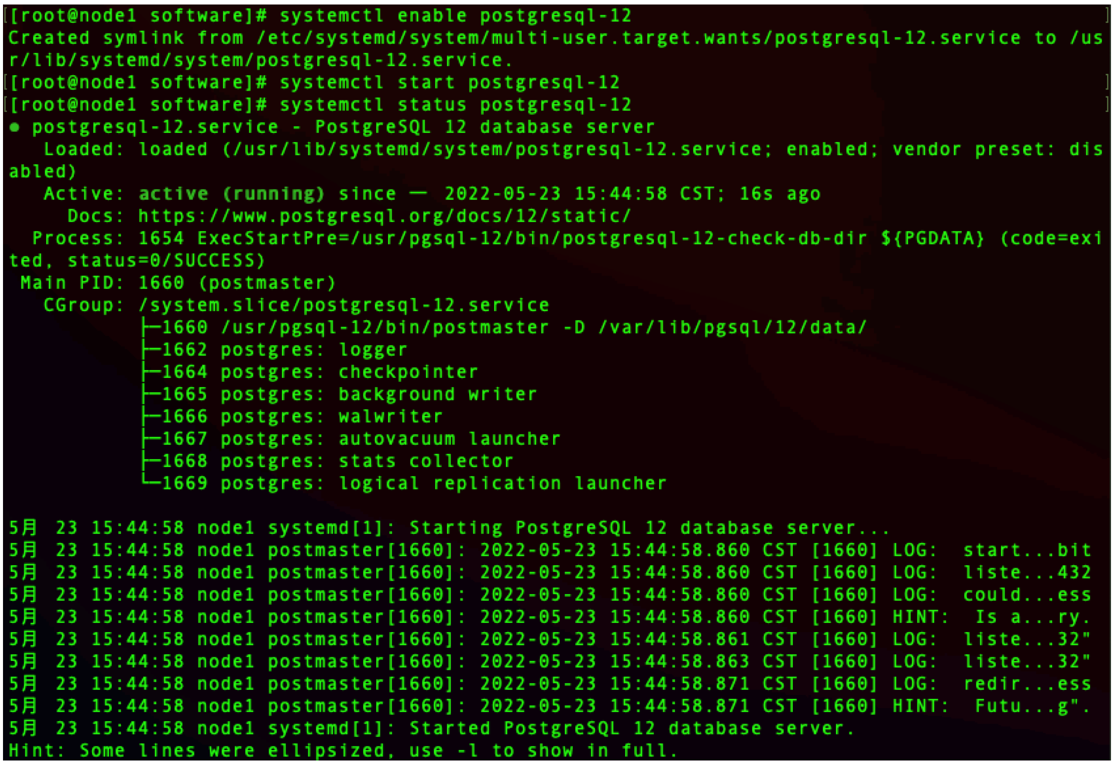


```
[root@node1 software]# /usr/pgsql-12/bin/postgresql-12-setup initdb
Initializing database ... OK

[root@node1 software]# 
```

- 配置开机启动与启动

```
systemctl enable postgresql-12
```

```
systemctl start postgresql-12
```



```
[root@node1 software]# systemctl enable postgresql-12
Created symlink from /etc/systemd/system/multi-user.target.wants/postgresql-12.service to /us
r/lib/systemd/system/postgresql-12.service.
[root@node1 software]# systemctl start postgresql-12
[root@node1 software]# systemctl status postgresql-12
● postgresql-12.service - PostgreSQL 12 database server
   Loaded: loaded (/usr/lib/systemd/system/postgresql-12.service; enabled; vendor preset: dis
abled)
   Active: active (running) since 一 2022-05-23 15:44:58 CST; 16s ago
     Docs: https://www.postgresql.org/docs/12/static/
  Process: 1654 ExecStartPre=/usr/pgsql-12/bin/postgresql-12-check-db-dir ${PGDATA} (code=exi
ted, status=0/SUCCESS)
 Main PID: 1660 (postmaster)
   CGroup: /system.slice/postgresql-12.service
           ├─1660 /usr/pgsql-12/bin/postmaster -D /var/lib/pgsql/12/data/
           ├─1662 postgres: logger
           ├─1664 postgres: checkpointer
           ├─1665 postgres: background writer
           ├─1666 postgres: walwriter
           ├─1667 postgres: autovacuum launcher
           ├─1668 postgres: stats collector
           └─1669 postgres: logical replication launcher

5月 23 15:44:58 node1 systemd[1]: Starting PostgreSQL 12 database server...
5月 23 15:44:58 node1 postmaster[1660]: 2022-05-23 15:44:58.860 CST [1660] LOG:  start...bit
5月 23 15:44:58 node1 postmaster[1660]: 2022-05-23 15:44:58.860 CST [1660] LOG:  liste...432
5月 23 15:44:58 node1 postmaster[1660]: 2022-05-23 15:44:58.860 CST [1660] LOG:  could...ess
5月 23 15:44:58 node1 postmaster[1660]: 2022-05-23 15:44:58.860 CST [1660] HINT:  Is a...ry.
5月 23 15:44:58 node1 postmaster[1660]: 2022-05-23 15:44:58.861 CST [1660] LOG:  liste...32"
5月 23 15:44:58 node1 postmaster[1660]: 2022-05-23 15:44:58.863 CST [1660] LOG:  liste...32"
5月 23 15:44:58 node1 postmaster[1660]: 2022-05-23 15:44:58.871 CST [1660] LOG:  redir...ess
5月 23 15:44:58 node1 postmaster[1660]: 2022-05-23 15:44:58.871 CST [1660] HINT:  Futu...g".
5月 23 15:44:58 node1 systemd[1]: Started PostgreSQL 12 database server.
Hint: Some lines were ellipsized, use -l to show in full.
```

- 修改密码

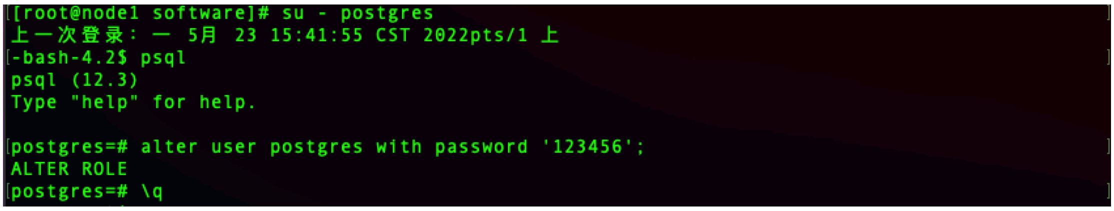
切换到 postgres 用户执行

```
su - postgres
```

```
psql
```

```
alter user postgres with password '123456';
```

```
\q
```



```
[root@node1 software]# su - postgres
上一次登录：一 5月 23 15:41:55 CST 2022pts/1 上
-bash-4.2$ psql
psql (12.3)
Type "help" for help.

postgres=# alter user postgres with password '123456';
ALTER ROLE
postgres=# \q
```

3.2.3 数据库其他配置

使用 root 用户执行

- 关闭防火墙

停止 firewall

```
systemctl stop firewalld.service
```

禁止 开机启动

```
systemctl disable firewalld.service
```

查看防火墙状态

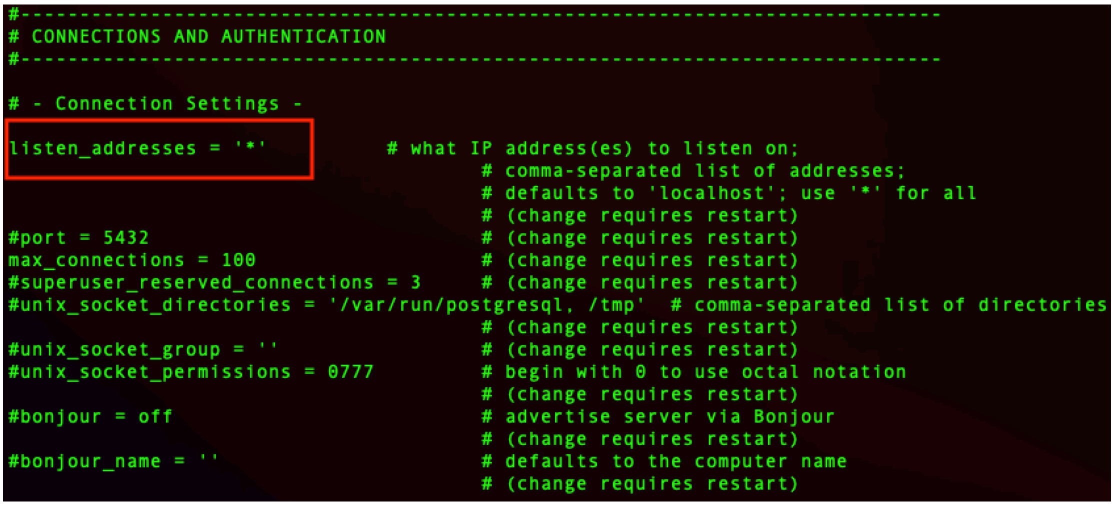
```
firewall -cmd -state
```

- 修改配置文件 postgresql.conf

  修改 IP 绑定，将监听地址修改为“*”

打开并编辑文件“/var/lib/pgsql/12/data/postgresql.conf” 将“#listen_addresses = ‘localhost’ ”改为“listen_addresses = * ”

```
vi /var/lib/pgsql/12/data/postgresql.conf
```



```
#------------------------------------------------------------------------------
# CONNECTIONS AND AUTHENTICATION
#------------------------------------------------------------------------------

# - Connection Settings -

listen_addresses = '*'          # what IP address(es) to listen on;
                                        # comma-separated list of addresses;
                                        # defaults to 'localhost'; use '*' for all
                                        # (change requires restart)
#port = 5432                            # (change requires restart)
max_connections = 100                   # (change requires restart)
#superuser_reserved_connections = 3     # (change requires restart)
#unix_socket_directories = '/var/run/postgresql, /tmp'  # comma-separated list of directories
                                        # (change requires restart)
#unix_socket_group = ''                 # (change requires restart)
#unix_socket_permissions = 0777         # begin with 0 to use octal notation
                                        # (change requires restart)
#bonjour = off                          # advertise server via Bonjour
                                        # (change requires restart)
#bonjour_name = ''                      # defaults to the computer name
                                        # (change requires restart)
```

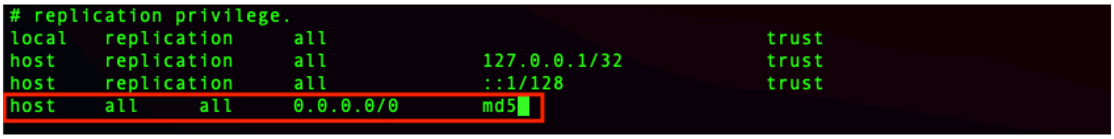
- 修改配置文件 pg_hba.conf

允许所有 IP 访问

打开并编辑文件“/var/lib/pgsql/12/data/pg_hba.conf”

```
vi /var/lib/pgsql/12/data/pg_hba.conf
```

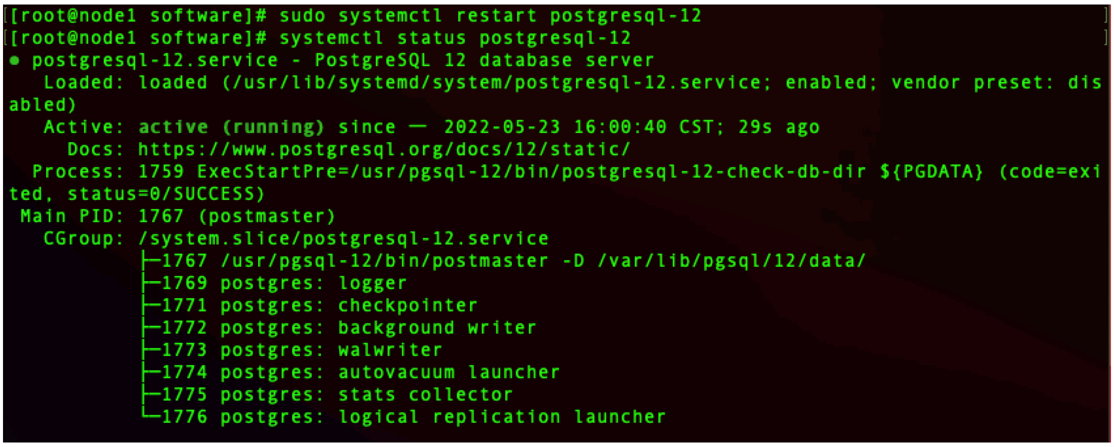
在文件的末尾添加
host all all 0.0.0.0/0 md5



```
# replication privilege.
local   replication     all                                     trust
host    replication     all             127.0.0.1/32            trust
host    replication     all             ::1/128                 trust
host    all     all     0.0.0.0/0       md5
```

3.2.4 重启 postgresql 服务器使设置生效

```
sudo systemctl restart postgresql-12
```



```
[root@node1 software]# sudo systemctl restart postgresql-12
[root@node1 software]# systemctl status postgresql-12
● postgresql-12.service - PostgreSQL 12 database server
   Loaded: loaded (/usr/lib/systemd/system/postgresql-12.service; enabled; vendor preset: dis
abled)
   Active: active (running) since — 2022-05-23 16:00:40 CST; 29s ago
     Docs: https://www.postgresql.org/docs/12/static/
  Process: 1759 ExecStartPre=/usr/pgsql-12/bin/postgresql-12-check-db-dir ${PGDATA} (code=exi
ted, status=0/SUCCESS)
 Main PID: 1767 (postmaster)
   CGroup: /system.slice/postgresql-12.service
           ├─1767 /usr/pgsql-12/bin/postmaster -D /var/lib/pgsql/12/data/
           ├─1769 postgres: logger
           ├─1771 postgres: checkpointer
           ├─1772 postgres: background writer
           ├─1773 postgres: walwriter
           ├─1774 postgres: autovacuum launcher
           ├─1775 postgres: stats collector
           └─1776 postgres: logical replication launcher
```