 一、CentOS7.9上OceanBase4.0 Docker容器的快速部署

**测试环境配置：**

一台8C16G100G的虚拟机，系统是CentOS7.9

**1、安装docker**

安装所需要的系统工具

yum install -y yum-utils device-mapper-persistent-data lvm2

添加镜像源

yum-config-manager --add-repo <http://mirrors.aliyun.com/docker-ce/linux/centos/docker-ce.repo>

更新yum缓存

yum makecache fast

安装docker-ce

yum -y install docker-ce

启动docker服务

systemctl start docker

**2、安装oceanbase-ce**

查看oceanbase镜像

docker search oceanbase

拉取镜像

docker pull obpilot/oceanbase-ce:latest

运行

docker run -itd -m 12G --name oceanbase-ce obpilot/oceanbase-ce:latest

计算机生成了可选文字:
[ r001@0 e 一 ] # docker pull obpilot/oceanbase-ce:latest 
latest ： Pulling from obpilot/oceanbase-ce 
a1de07532777 ： Pull complete 
42e2faadf654: Pull complete 
b888a e21 ： Pull 0 懾 pl e 
25012g255 # 8 ： Pull complete 
a66beb3@5bg2 ： Pull complete 
2d18g 55ae1 ： Pull complete 
Digest ： a255 :4c2d161e6cdc4bd610baf155e43e@48d8g6de4d6bfc777534e75bfa@497bff4g 
St at u s ： Downloaded n ewe r Image fo r obpilot/oceanbase-ce:latest 
docker.io/obpilot/oceanbase-ce:latest 
You have n mail in /var/spooVmaiVroot 
[ r001@obtest 一 ] # docker run -itd -m 126 一 -n am e oceanbase-ce obpflot/oceanbase-ce:latest 
23ee42ae2e6a53d2dbd82e40c4dc63fe47be5aebag3a60067gIf65bc2@aa5@I 
[ r001@0 e 一 ] # docker ps 丨 g rep oceanbase 
230e42a@2@6a obpilot/oceanbase-ce:latest 
"/bin/bash " 
19 seconds ago Up 18 seconds 
2881 / ℃ p ， 
2883 / ℃ p ， 
300/tcp ， 
9090/tcp 
Oceanbase-ce 

**3、进入容器，OBD 命令完成OceanBase 集群部署**

docker exec -it oceanbase-ce bash

查看集群信息

[admin@230e42a0206a ~]$ obd cluster list

+------------------------------------------------------------+

| Cluster List |

+--------+---------------------------------+-----------------+

| Name | Configuration Path | Status (Cached) |

+--------+---------------------------------+-----------------+

| obdemo | /home/admin/.obd/cluster/obdemo | deployed |

+--------+---------------------------------+-----------------+

启动

[admin@230e42a0206a ~]$ obd cluster start obdemo

Get local repositories ok

Search plugins ok

Open ssh connection ok

Load cluster param plugin ok

Check before start observer ok

[WARN] (127.0.0.1) clog and data use the same disk (/)

Check before start obproxy ok

Start observer ok

observer program health check ok

Connect to observer ok

Initialize cluster ok

Start obproxy ok

obproxy program health check ok

Connect to obproxy ok

Initialize cluster ok

Wait for observer init ok

+---------------------------------------------+

| observer |

+-----------+---------+------+-------+--------+

| ip | version | port | zone | status |

+-----------+---------+------+-------+--------+

| 127.0.0.1 | 4.0.0.0 | 2881 | zone1 | ACTIVE |

+-----------+---------+------+-------+--------+

obclient -h127.0.0.1 -P2881 -uroot -prootPWD123 -Doceanbase

+---------------------------------------------+

| obproxy |

+-----------+------+-----------------+--------+

| ip | port | prometheus\_port | status |

+-----------+------+-----------------+--------+

| 127.0.0.1 | 2883 | 2884 | active |

+-----------+------+-----------------+--------+

obclient -h127.0.0.1 -P2883 -uroot -prootPWD123 -Doceanbase

obdemo running

**4、通过obproxy连接ob，创建业务租户、数据库和表**

obclient -h127.0.0.1 -P2883 -uroot -prootPWD123 -Doceanbase

创建资源单元 U2C4G 配置为 2 个 CPU，4G 内存，128- 个 IOPS，10G 日志盘

CREATE RESOURCE UNIT U2C4G MAX\_CPU 2, MEMORY\_SIZE '4G', MAX\_IOPS 1280,LOG\_DISK\_SIZE '10G', MIN\_IOPS=1024;

创建资源池时报错，内存不足

CREATE resource pool my\_pool unit = 'U2C4G', unit\_num = 1;

obclient [oceanbase]> CREATE resource pool my\_pool unit = 'U2C4G', unit\_num = 1;

ERROR 4733 (HY000): zone 'zone1' resource not enough to hold 1 unit. You can check resource info by views: DBA\_OB\_UNITS, GV$OB\_UNITS, GV$OB\_SERVERS.

server '"127.0.0.1:2882"' MEMORY resource not enough

将U2C4G调小为2G后重新创建资源池成功

Alter resource unit U2C4G MEMORY\_SIZE '2G';

计算机生成了可选文字:
obclient loceanbaseb select 
丨 gmt ℃ reate 
CPI-I 
丨 2e2 一 03 一 08 11 ： eg ： eg · 535581 
丨 2e2 一 03 一 08 11 ： 2 5 ： 40 · 7e2291 
CPI-I 
2e2 一 03 一 08 11 ： :øg · 535581 
2e2 一 03 一 08 13 ： 03 ： # ． 190807 
from 
a11 unit config; 
modified 
丨 unit 
2e2 一 03 一 08 11:09:09 · 535581 | 
2e2 一 03 一 08 11 ： 2 5 ： 40 · 7e2291 | 
config id 
1001 
config id 
1001 
name 
SYS un 11 
U2C4G 
name 
SYS un 11 
U2C4G 
config 
config 
m a X 
m a X 
CPI-I 
CPI-I 
ry SIZe 
21 7 835 # 8 
29 # g572g5 
ry SIZe 
21 7 835 # 8 
21 7 835 # 8 
1 匹 disk size 
21 7 835 # 8 
1e737 182 
1 匹 disk size 
21 7 835 # 8 
1e737 182 e 
丨 max_iOPS 
128e 
丨 max_iOPS 
128e 
丨 min_iops 丨 iops weight | 
1e2 丨 
丨 min_iops 丨 iops weight | 
1e2 丨 
2 rows in set (0.001 sec } 
obclient loceanbasel:• Iter resou rce unit U2C4G 丨 M 〔 Y SIZE · 26 · 
a11 unit config; 
丨 unit 
Query ， rows affected (0.003 sec) 
obclient loceanbaseb select 
from 
gmt ℃ reate 
2e2 一 03 一 08 11 ： ： ． 535581 
2e2 一 03 一 08 11 ： 26 ： 40 ． 7e2291 
set (0.001 sec } 
In 
modified 

创建业务租户

create tenant obmysql resource\_pool\_list=('my\_pool'), primary\_zone='RANDOM',comment 'mysql tenant/instance', charset='utf8mb4' set ob\_tcp\_invited\_nodes='%', ob\_compatibility\_mode='mysql';

登录obmysql租户，默认密码为空

obclient -h 127.1 -uroot@obmysql -P2883 -p -c

修改租户的root密码，下次登录后需要输入密码

ALTER USER root IDENTIFIED BY 'Hello123';

创建业务数据库和表

obclient [(none)]> create database testdb1;

Query OK, 1 row affected (0.035 sec)

obclient [(none)]> use testdb1;

Database changed

obclient [testdb1]> create table t1(id int,content varchar(10));

Query OK, 0 rows affected (0.076 sec)

obclient [testdb1]> insert into t1 values(1,'test');

Query OK, 1 row affected (0.020 sec)

obclient [testdb1]> select \* from t1;

+------+---------+

| id | content |

+------+---------+

| 1 | test |

+------+---------+

1 row in set (0.003 sec)