

# MPI 编程实验环境配置及示例

## 一、 MPI 编程实验流程概述

1. 购买三台华为鲲鹏云服务器及必要的网络资源，参照“二、 实验环境配置”中的“1. 购买华为鲲鹏云服务器”；
2. 三台机器均使用 ssh 连接，并进行免密配置，安装 mpi 及依赖包，参照“二、 实验环境配置”中的“2. 环境配置”； 注意请参照本文档配置购买资源，不要购买价格较高的额外资源，如果在实验、作业期间，代金券快用完了，请及时联系老师或助教。
3. 进行 HelloWorld 实验，验证环境安装正确，参照“三、 HelloWorld 实验示例”；
4. 完成所有实验、作业后，**必须停止/关闭所有华为云资源**，方法参照“四、 停止/关闭华为云资源”，否则将会继续扣费导致欠费！
5. 以防有学生没用过 shell 命令、vim 等，可以观看视频教程来跟着操作（注意我们购买三台机器实验即可），视频地址：

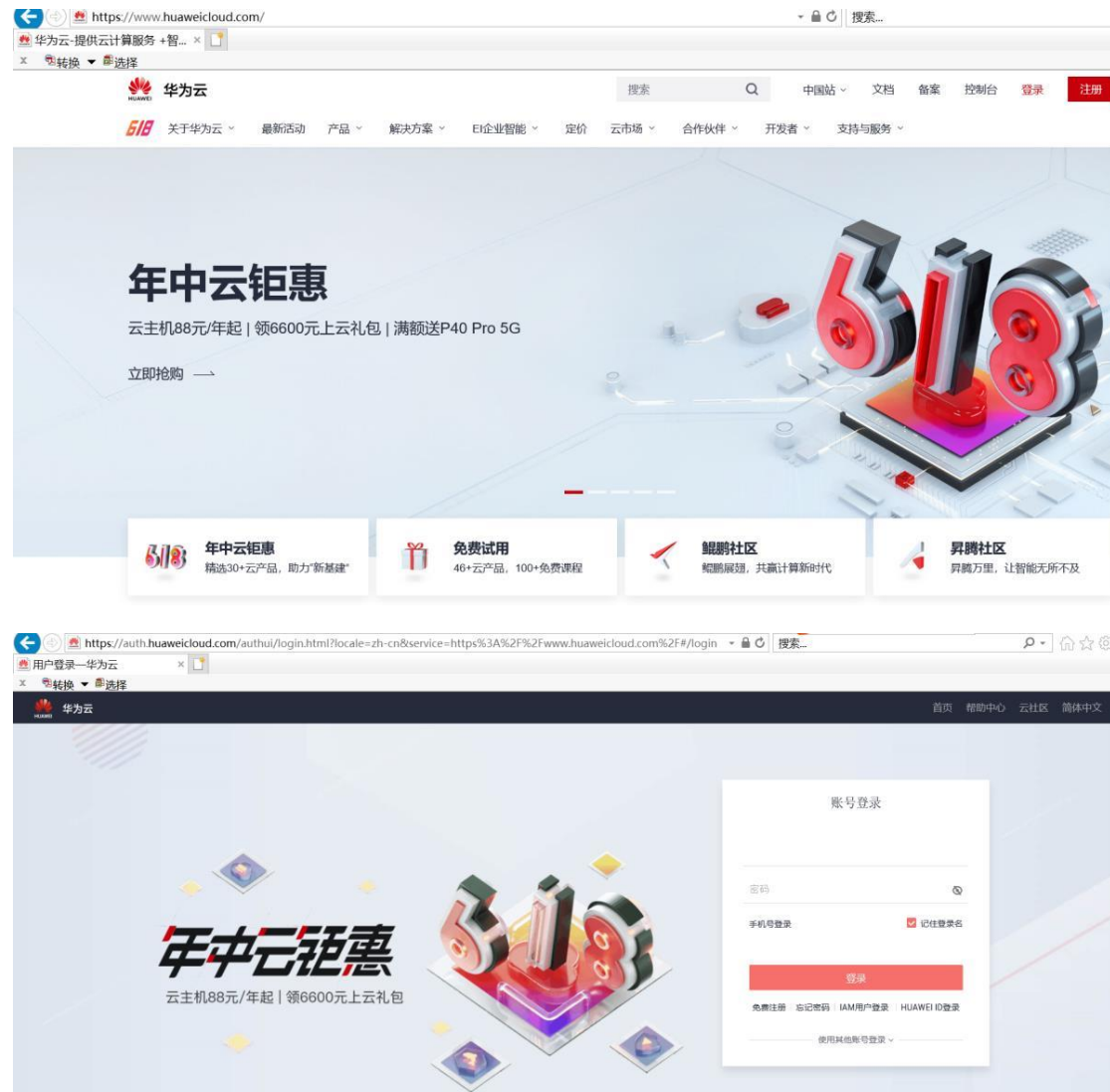
<https://education.huaweicloud.com/courses/course-v1:HuaweiX+CBUCNXK037+Self-paced/courseware/ab16d696bb4a458ca21a37396c4a56c0/3074a26390154e4fab6ea610d9b00975/>

## 二、实验环境配置

### 1. 购买华为鲲鹏云服务器

#### 步骤 1 登录华为云

打开浏览器，输入华为云的域名：<https://www.huaweicloud.com>，点击右上角登录按钮，输入用户名与密码。

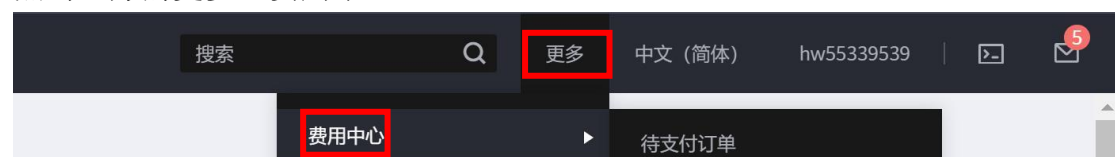


点击控制台。

最新活动 产品 解决方案 云市场 合作伙伴 开发者 支持与服务 了解华为云

Q 文档 备案 控制台

点击上方的更多，费用中心。



点击代金券中的立即使用。

总览

现金余额

充值

提现

¥0.00

汇款认领

可用额度

预警

(预警阈值: ¥50.00 修改)

当您的可用额度、通用代金券和现送短信和邮件提醒。

¥0.00 = 现金余额 ¥0.00 - 欠费金额 ¥0.00

待支付订单

0 个待支付

立即支付 ▶

代金券

1 张(¥200.00)

立即使用 ▶

可开票 ?

¥0.00

去开发票 ▶

点击操作中的立即使用。

| 余额(¥)  | 金额限制 | 有效期  | 适用产品      | 计费模式  | 备注                      | 操作   |
|--------|------|--|-----------|-------|-------------------------|------|
| 200.00 | 不限金额 | 2021/12/03 00:00:00...<br>2022/03/03 23:59:00... | 指定产品可用... | 包月、按需 | 不可与商务授权折扣/伙伴授权折扣同享, ... | 立即使用 |

## 步骤 2 购买弹性云服务器（ECS）

点击“弹性云服务器 ECS”。

华为云 | 控制台

搜索

更多

自定义 评价

收藏服务 ?

弹性云服务器 ECS

云硬盘 EVS

弹性负载均衡 ELB

云耀云服务器 HECS

云备份 CBR

弹性公网IP EIP

裸金属服务器 BMS

对象存储服务 OBS

云数据库 RDS

弹性伸缩 AS

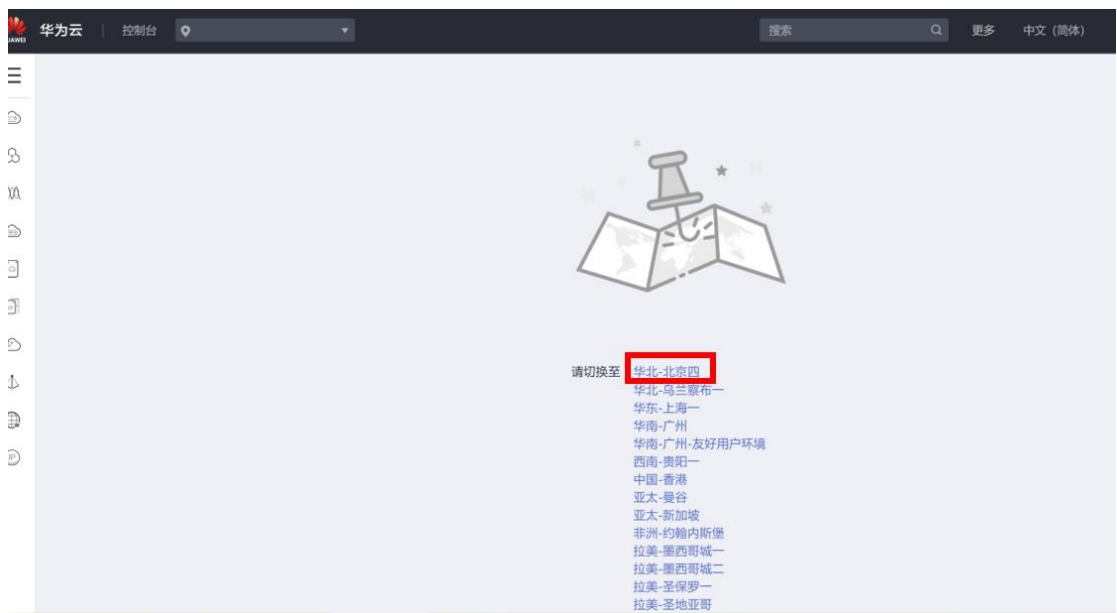
虚拟私有云 VPC

域名注册 Domains

0

查看更多服务

如果出现下面界面，点击“华北-北京四”：



点击界面左侧的“弹性云服务器”，再点右侧“购买弹性云服务器”：



### 步骤 3 基础配置

按照下表配置进行购买，一共购买 3 台。

| 计费模式 | 区域     | CPU架构 | 规格          | 镜像                          | 系统盘    |
|------|--------|-------|-------------|-----------------------------|--------|
| 按需计费 | 华北-北京四 | 鲲鹏计算  | kc1.large.2 | 公共镜像：<br>openEuler<br>20.03 | 至少40GB |

参照截图如下：

弹性云服务器

自定义购买

快速购买

1 基础配置

2 网络配置

3 高级配置

4 确认配置

计费模式

包年/包月

按量计费

竞价计费

区域

华北-北京四

推荐区域

华北-乌兰察布一

西南-贵阳一

华北-北京四

华东-广州

华东-上海一

可用区

随机分配

可用区1

可用区2

可用区3

可用区7

CPU架构

x86计算

规格

最新系列

vCPUs

全部

内存

全部

规格名称

AI加速型

鲲鹏通用计算增强型

鲲鹏内存优化型

鲲鹏超密IO型

| 规格名称         | vCPUs   内存(GiB) | CPU                       | 基准 / 最大带宽      | 内网收发包   | 规格参考价    |
|--------------|-----------------|---------------------------|----------------|---------|----------|
| kc1.small.1  | 1vCPUs   1 GiB  | Huawei Kunpeng 920 2.6GHz | 0.5 / 2 Gbit/s | 200,000 | ¥0.12/小时 |
| kc1.large.2  | 2vCPUs   4 GiB  | Huawei Kunpeng 920 2.6GHz | 0.8 / 3 Gbit/s | 300,000 | ¥0.30/小时 |
| kc1.large.4  | 2vCPUs   8 GiB  | Huawei Kunpeng 920 2.6GHz | 0.8 / 3 Gbit/s | 300,000 | ¥0.41/小时 |
| kc1.xlarge.2 | 4vCPUs   8 GiB  | Huawei Kunpeng 920 2.6GHz | 1.5 / 5 Gbit/s | 500,000 | ¥0.60/小时 |
| kc1.xlarge.4 | 4vCPUs   16 GiB | Huawei Kunpeng 920 2.6GHz | 1.5 / 5 Gbit/s | 500,000 | ¥0.81/小时 |

当前规格

鲲鹏通用计算增强型 | kc1.large.2 | 2vCPUs | 4 GiB

镜像

公共镜像

私有镜像

共享镜像

市场镜像

openEuler

openEuler 20.03 64bit with ARM(40GB)

主机安全

开通主机安全

基础版

企业版

系统盘

高IO

40

GiB

IOPS上限2,120, IOPS突发上限5,000

购买量

3

台

配置费用 ¥0.9588/小时

参考价格, 具体扣费请以账单为准。了解详情

下一步: 网络配置

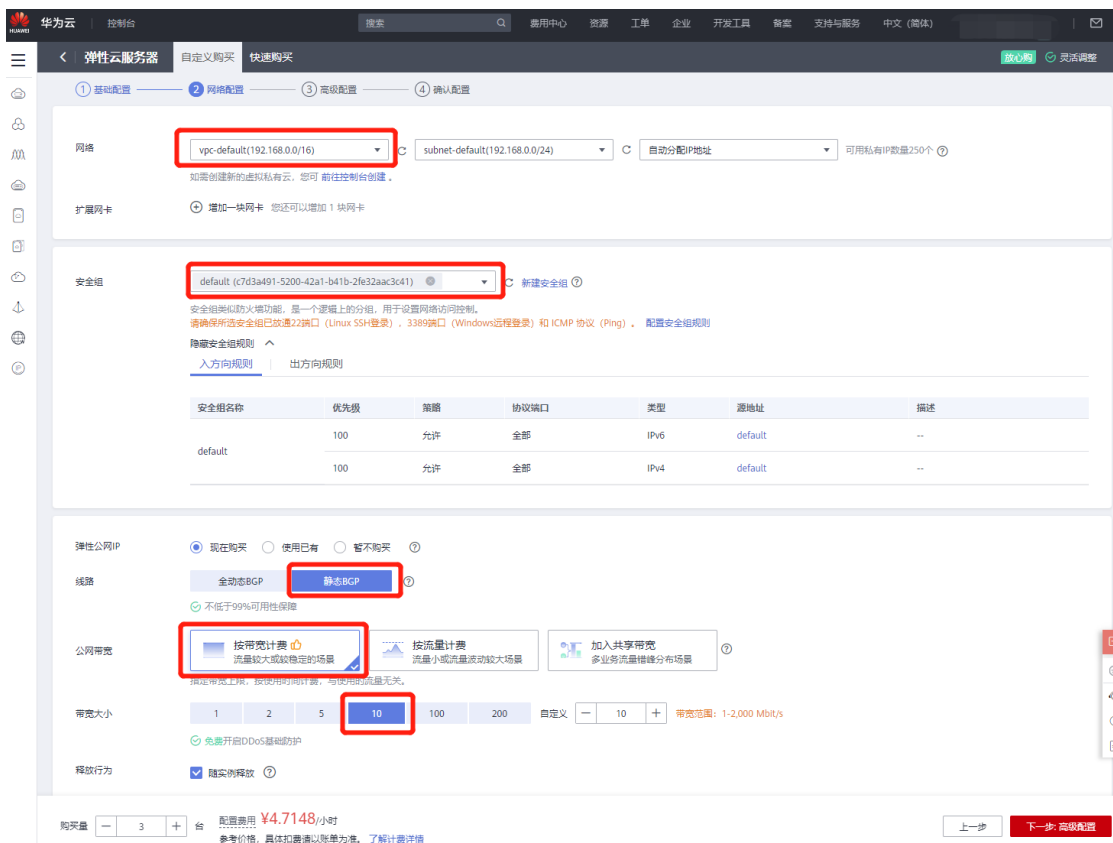
点击下一步:网络配置。

## 步骤 4 网络配置

按照下表进行网络配置

| 网络     | 安全组     | 弹性公网IP               |
|--------|---------|----------------------|
| 默认的VPC | default | 现在购买，带宽大小选择10Mbits/s |

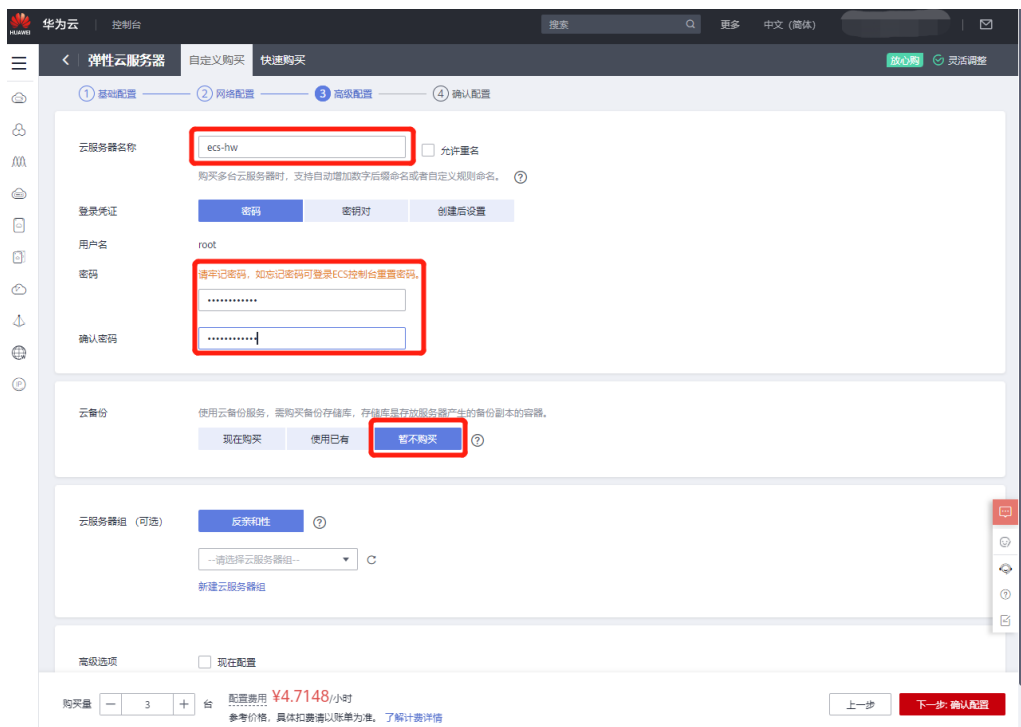
参照截图如下：



点击“下一步 高级配置”。

## 步骤 5 高级配置

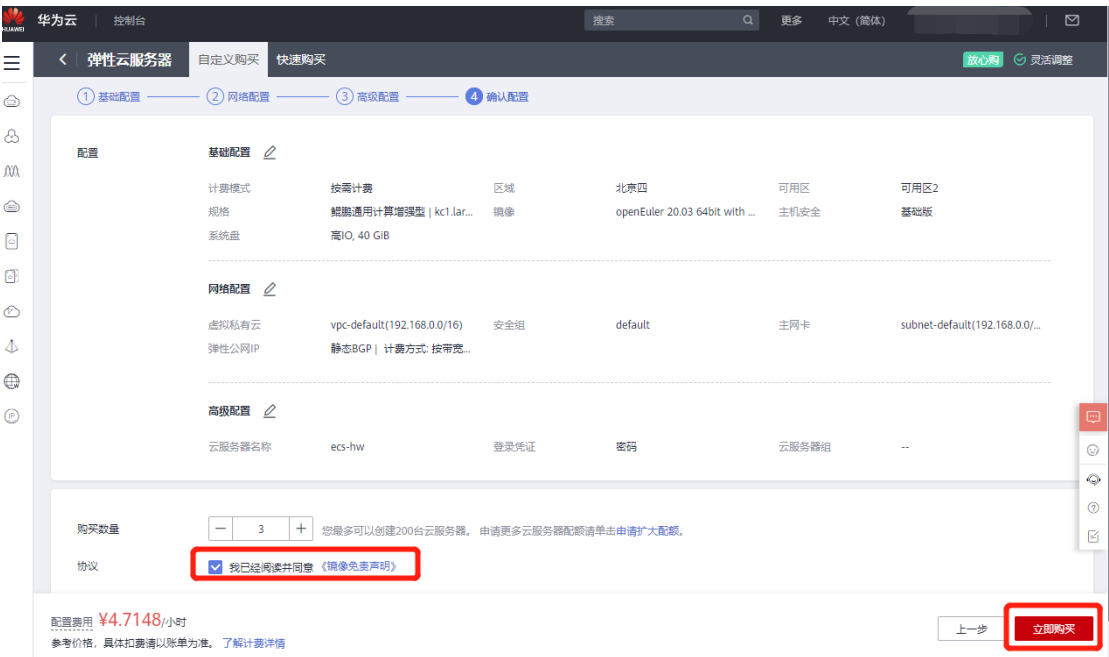
设置云服务器名称(eg: ecs-hw)、密码(Parallel2021)，云备份选“暂不购买”。



点击“下一步：确认配置”。

### 步骤 6 确认配置

勾选我已阅读并同意《华为镜像免责声明》，点击确认配置。



购买成功后可去“我的资源”中查看资源，如下图查看 弹性云服务器资源。



注意：此时“我的资源”中除了刚刚购买的“弹性云服务器 ECS”之外，还会由“虚拟私有云 VPC”、“云硬盘 EVS”等资源，等完成实验、作业之后，一定要把所有资源全部关停（参照“四”部分内容），否则代金券扣完还会继续扣费，你将欠费！

## 2. 环境配置

### 步骤 1 下载 ssh 工具

推荐 Xshell (Windows)、vscode + ssh remote 插件、Royal TSX (mac), mac OS 的 terminal 自带 ssh 功能，或从微软应用商店中下载安装 Ubuntu 子系统（Windows10）。

## 步骤 2 登录 ECS

创建好 ECS 后，可以在控制台 -> 弹性云服务器中看到弹性公网 IP。使用 SSH 工具，输入公网 IP、用户名和密码，或 `ssh usr@IP` 即可登陆，例如 `ssh root@114.116.218.203`，再根据提示输入密码，即可登录 `ecs-hw-0001`。

信息如下：

| 主机名         | 私网IP          | 公网IP            |
|-------------|---------------|-----------------|
| ecs-hw-0001 | 192.168.0.219 | 114.116.218.203 |
| ecs-hw-0002 | 192.168.0.75  | 114.116.210.104 |
| ecs-hw-0003 | 192.168.0.167 | 114.116.199.71  |

## 步骤 3 环境配置说明

为了防止大家的文件混乱，建议大家在每一台机器下都建立个人账户，不建议统一使用 root 账户，下面以用户名 `zhangsan` 为例，三台机器名分别为 `ecs-hw-0001`, `ecs-hw-0002`, `ecs-hw-0003`，三台机器的私网 ip 为 `192.168.0.219`, `192.168.0.75`, `192.168.0.167`。环境配置时在每一台主机上 都需要重复执行，步骤 4 和步骤 5 三台机器都需要执行，ip 需要根据实际分配进行相应调整，建议打开三个 shell 窗口同时进行操作。



## 步骤 4 创建用户

每台主机都需要在 root 账户下建立个人的账户 `zhangsan`，并添加到 `wheel` 用户组（改组具有 `sudo` 权限），命令如下：

```
adduser zhangsan
passwd zhangsan
usermod -aG wheel zhangsan
```

说明：`zhangsan` 密码也设置为 `Parallel2021`

## 步骤 5 免密配置

以下所有步骤在三台机器上均需要重复执行。



(1)配置三台机器主机名和 ip 解析（各主机 ip 可以通过 ifconfig 或者控制台界面查看）

```
vim /etc/hosts
```

注释文件原先本身的信息并添加三台机器的“私网 IP 主机”信息（注释的目的是因为会对本实验程序的运行产生报错），添加完后信息如下：

```
#:1 localhost localhost.localdomain localhost6
localhost6.localhost6
#127.0.0.1 localhost localhost.localdomain localhost
ost4 localhost4.localhost4
#127.0.0.1 localhost localhost
#127.0.0.1 ecs-hw-0001 ecs-hw-0001
192.168.0.219 ecs-hw-0001
192.168.0.75 ecs-hw-0002
192.168.0.167 ecs-hw-0003
```

注释不生效

添加内容

(2) 登录新账户

首先退出 root 账户重新登录到新建立的账户下：

```
su - zhangsan
```

(3) 本地生成密钥：

```
ssh-keygen -t rsa -b 4096
```

说明：会让输入信息，保持默认，回车即可（三次回车）。

```
[zhangsan@ecs-hw-0001 ~]$ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/home/zhangsan/.ssh/id_rsa):
Created directory '/home/zhangsan/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/zhangsan/.ssh/id_rsa.
Your public key has been saved in /home/zhangsan/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:0196aYTMEdZZD49yWtecaZccbtUXgfU377K0ZopYhU zhangsan@ecs-hw-0001
The key's randomart image is:
+---[RSA 4096]-----+
|      . oo .o=*|
|      o o  =o+*|
|      . . . +X*|
|      . E .o +|
|      oSo. . o |
|      .+... . |
|      .o. . . |
|      . =+.. =. |
|      +o. .*. |
+---[SHA256]-----+
[zhangsan@ecs-hw-0001 ~]$
```

(4) 添加公钥至所有主机（根据提示输入 yes、密码）：

```
ssh-copy-id zhangsan@ecs-hw-0001
ssh-copy-id zhangsan@ecs-hw-0002
ssh-copy-id zhangsan@ecs-hw-0003
```

(5) 安装依赖包：

```
sudo yum -y install gcc-gfortran
```

(6) 源码编译安装 mpi (如果执行下面命令提示没有权限就执行 sudo 命令，如“sudo wget ...” )：

```
wget http://www.mpich.org/static/downloads/3.3.2/mpich-3.3.2.tar.gz
tar -zxvf mpich-3.3.2.tar.gz
cd mpich-3.3.2
./configure
sudo make && sudo make install
```

### 三、 HelloWorld 实验示例

#### 1. 创建示例程序源码（注意：这里我们只有 3 台机器，示例是 4 台）

以下步骤均在 ecs-hw-0001 上执行。

执行以下命令，创建 hello 目录存放该程序的所有文件，并进入 hello 目录（四台主机都执行）

```
mkdir /home/zhangsan/hello.  
cd /home/zhangsan/hello.
```

执行以下命令，创建示例程序源码 mpi\_hello\_world.c（四台主机都执行）

```
vim mpi_hello_world.c
```

代码如下：

```
#include <mpi.h>  
#include <stdio.h>  
int main(int argc, char** argv) {  
    // Initialize the MPI environment. The two arguments to MPI Init are not  
    // currently used by MPI implementations, but are there in case future  
    // implementations might need the arguments.  
    MPI_Init(NULL, NULL);  
    // Get the number of processes  
    int world_size;  
    MPI_Comm_size(MPI_COMM_WORLD, &world_size);  
    // Get the rank of the process  
    int world_rank;  
    MPI_Comm_rank(MPI_COMM_WORLD, &world_rank);  
    // Get the name of the processor  
    char processor_name[MPI_MAX_PROCESSOR_NAME];  
    int name_len;  
    MPI_Get_processor_name(processor_name, &name_len);  
    // Print off a hello world message  
    printf("Hello world from processor %s, rank %d out of %d processors\n",  
           processor_name, world_rank, world_size);  
    // Finalize the MPI environment. No more MPI calls can be made after this  
    MPI_Finalize();  
}
```

```
#include <mpi.h>  
#include <stdio.h>  
int main(int argc, char** argv) {  
    // Initialize the MPI environment. The two arguments to MPI Init are not  
    // currently used by MPI implementations, but are there in case future  
    // implementations might need the arguments.  
    MPI_Init(NULL, NULL);  
    // Get the number of processes  
    int world_size;  
    MPI_Comm_size(MPI_COMM_WORLD, &world_size);  
    // Get the rank of the process  
    int world_rank;  
    MPI_Comm_rank(MPI_COMM_WORLD, &world_rank);  
    // Get the name of the processor  
    char processor_name[MPI_MAX_PROCESSOR_NAME];  
    int name_len;  
    MPI_Get_processor_name(processor_name, &name_len);  
    // Print off a hello world message  
    printf("Hello world from processor %s, rank %d out of %d processors\n", processor_name, world_rank, world_size);  
    // Finalize the MPI environment. No more MPI calls can be made after this  
    MPI_Finalize();  
}
```

## 2. 创建 makefile（注意：这里我们只有 3 台机器，示例是 4 台）

执行以下命令，创建 makefile（四台主机都执行）

```
vim makefile.
```

代码内容如下：

```
EXECS=mpi_hello_world
MPICC?=mpicc
all: ${EXECS}
mpi_hello_world: mpi_hello_world.c
${MPICC} -o mpi_hello_world mpi_hello_world.c
clean:
rm -f ${EXECS}
```

```
EXECS=mpi_hello_world
MPICC?=mpicc
all: ${EXECS}
mpi_hello_world: mpi_hello_world.c
${MPICC} -o mpi_hello_world mpi_hello_world.c
clean:
rm -f ${EXECS}
```

## 3. 进行编译（注意：这里我们只有 3 台机器，示例是 4 台）

执行以下命令，进行编译（四台主机都执行）

```
cd /home/zhangsan/hello.
make.
```

```
[zhangsan@ecs-hw-0001 hello]$
[zhangsan@ecs-hw-0001 hello]$ ll
total 8.0K
-rw----- 1 zhangsan zhangsan 155 Jul  3 19:37 makefile
-rw----- 1 zhangsan zhangsan 871 Jul  3 19:36 mpi_hello_world.c
[zhangsan@ecs-hw-0001 hello]$
[zhangsan@ecs-hw-0001 hello]$ make
mpicc -o mpi_hello_world mpi_hello_world.c
[zhangsan@ecs-hw-0001 hello]$
[zhangsan@ecs-hw-0001 hello]$ ll
total 24K
-rw----- 1 zhangsan zhangsan 155 Jul  3 19:37 makefile
-rwx----- 1 zhangsan zhangsan 70K Jul  3 19:41 mpi_hello_world
-rw----- 1 zhangsan zhangsan 871 Jul  3 19:36 mpi_hello_world.c
[zhangsan@ecs-hw-0001 hello]$
```

## 4. 建立主机配置文件（注意：这里我们只有 3 台机器，示例是 4 台）

执行以下命令，建立主机配置文件（四台主机都执行）

```
vim /home/zhangsan/hello/config
```

添加内容如下:

```
ecs-hw-0001:2  
ecs-hw-0002:2  
ecs-hw-0003:2  
ecs-hw-0004:2
```

```
[zhangsan@ecs-hw-0001 hello]$ more config  
ecs-hw-0001:2  
ecs-hw-0002:2  
ecs-hw-0003:2  
ecs-hw-0004:2  
  
[zhangsan@ecs-hw-0001 hello]$
```

## 5. 运行监测

执行以下命令, 查看运行结果 (只需要在 **ecs-hw-0001** 上执行)

```
mpiexec -n 8 -f /home/zhangsan/hello/config /home/zhangsan/hello/mpi_hello_world
```

结果如下:

```
[zhangsan@ecs-hw-0001 hello]$ mpiexec -n 8 -f /home/zhangsan/hello/config /home/zhangsan/hello/mpi_hello_world  
Authorized users only. All activities may be monitored and reported.  
Authorized users only. All activities may be monitored and reported.  
Authorized users only. All activities may be monitored and reported.  
Hello world from processor ecs-hw-0001, rank 0 out of 8 processors  
Hello world from processor ecs-hw-0004, rank 6 out of 8 processors  
Hello world from processor ecs-hw-0003, rank 4 out of 8 processors  
Hello world from processor ecs-hw-0002, rank 2 out of 8 processors  
Hello world from processor ecs-hw-0001, rank 1 out of 8 processors  
Hello world from processor ecs-hw-0004, rank 7 out of 8 processors  
Hello world from processor ecs-hw-0003, rank 5 out of 8 processors  
Hello world from processor ecs-hw-0002, rank 3 out of 8 processors  
[zhangsan@ecs-hw-0001 hello]$  
[zhangsan@ecs-hw-0001 hello]$ mpiexec -n 8 -f /home/zhangsan/hello/config /home/zhangsan/hello/mpi_hello_world  
Authorized users only. All activities may be monitored and reported.  
Authorized users only. All activities may be monitored and reported.  
Authorized users only. All activities may be monitored and reported.  
Hello world from processor ecs-hw-0001, rank 0 out of 8 processors  
Hello world from processor ecs-hw-0001, rank 1 out of 8 processors  
Hello world from processor ecs-hw-0003, rank 4 out of 8 processors  
Hello world from processor ecs-hw-0004, rank 6 out of 8 processors  
Hello world from processor ecs-hw-0002, rank 2 out of 8 processors  
Hello world from processor ecs-hw-0003, rank 5 out of 8 processors  
Hello world from processor ecs-hw-0004, rank 7 out of 8 processors  
Hello world from processor ecs-hw-0002, rank 3 out of 8 processors  
[zhangsan@ecs-hw-0001 hello]$
```

通过上述代码运行, 可以看出, 编写的 hello-world 程序已经在华为鲲鹏上运行起来, 程序在集群之间并行计算处理。

## 四、 停止/关闭华为云资源

教学或实验完成之后/不使用华为云资源时，请**务必先停止相关服务并删除所有资源**，再退出华为云账号，避免系统认为资源占用继续扣费并导致欠费。

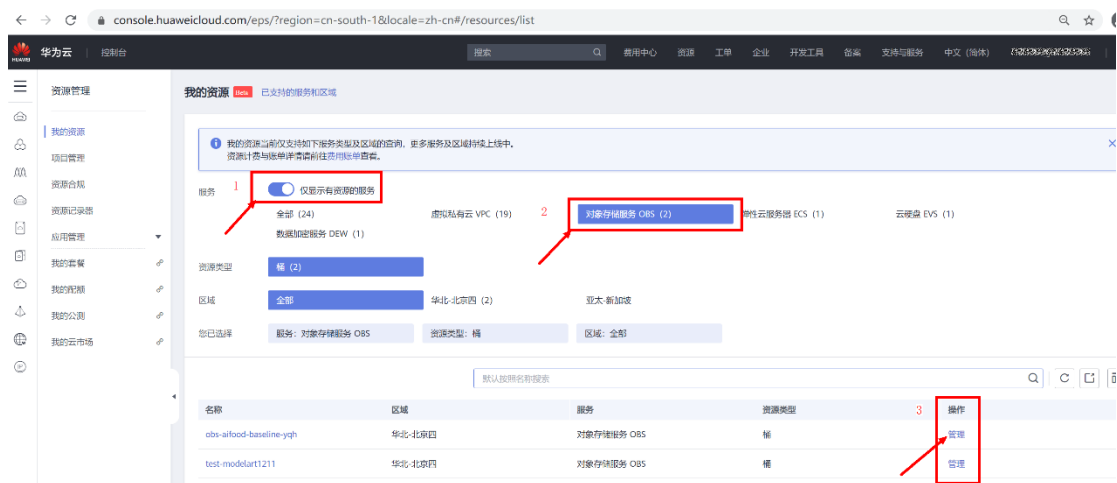
### 步骤 1

点击进入“控制台” “资源” “我的资源”。如下图所示：



### 步骤 2

打开“仅显示有资源的服务”按钮，逐个点击具体服务，在下方服务资源表格“操作”列中进入“管理”界面，进行各服务的停止、删除操作。（资源删除，数据也将被清除，操作前请务必先确认数据是否可清除）。



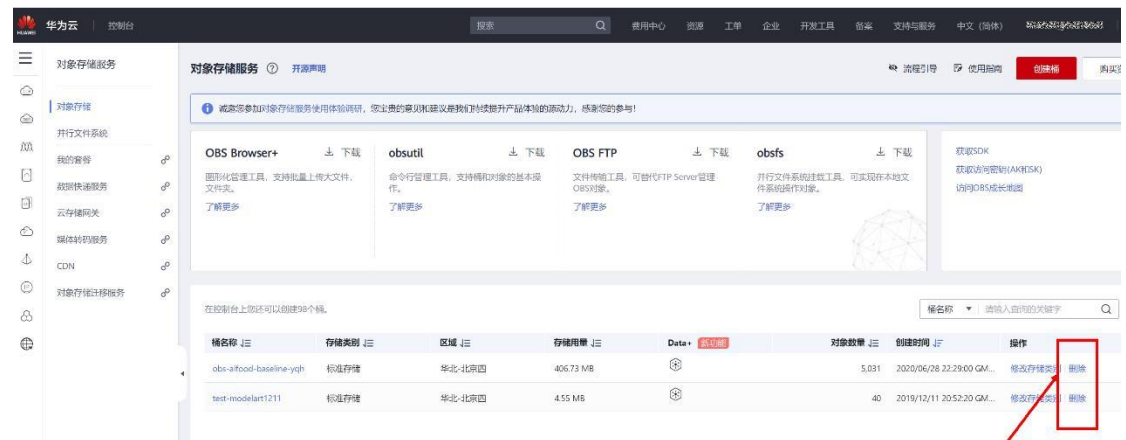
### 步骤 3

如“对象存储服务 OBS”服务所实例，进入“管理”界面后，点击服务名称的上一级目录入口，进入服务控制台。如下图所示：



## 步骤 4

对各 OBS 实例逐个进行“删除”操作。OBS 实例桶中如果有数据，需先清除，具体参考提示操作即可。如下图所示：



## 步骤 5

检查“我的资源”，如果还存在任何 1 项资源，务必关停并删除，直到没有任何资源为止。