华为认证 Kenpeng 系列教程

## HCIA-Kunpeng Application Developer

## 应用迁移实验指导手册

版本:1.0



华为技术有限公司

## 版权所有 © 华为技术有限公司 2019。 保留一切权利。

非经本公司书面许可,任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部,并不得以任何形式传播。

#### 商标声明

## 

huawe和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标,由各自的所有人拥有。

## 注意

您购买的产品、服务或特性等应受华为公司商业合同和条款的约束,本文档中描述的全部或部分产品、服务 或特性可能不在您的购买或使用范围之内。除非合同另有约定,华为公司对本文档内容不做任何明示或暗示 的声明或保证。

由于产品版本升级或其他原因,本文档内容会不定期进行更新。除非另有约定,本文档仅作为使用指导,本 文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

## 华为技术有限公司

地址: 深圳市龙岗区坂田华为总部办公楼 邮编:518129

网址: http://e.huawei.com



## 华为认证体系介绍

华为认证是华为公司基于"平台+生态"战略,围绕"云-管-端"协同的新ICT技术架构,打造的ICT技术架构认证、平台与服务认证、行业ICT认证三类认证,是业界唯一覆盖 ICT (Information and Communications Technology 信息通信技术)全技术领域的认证 体系。

根据ICT从业者的学习和进阶需求,华为认证分为工程师级别、高级工程师级别和专家 级别三个认证等级。华为认证覆盖ICT全领域,符合ICT融合的技术趋势,致力于提供领先的 人才培养体系和认证标准,培养数字化时代新型ICT人才,构建良性ICT人才生态。

华为认证HCIA-Kunpeng Application Developer V1.0定位于培养与认证能够应用华为鲲鹏计算平台进行业务部署与迁移,性能测试与调优,常见解决方案设计与规划的工程师。

通过HCIA-Kunpeng Application Developer V1.0认证,您将掌握对鲲鹏计算平台的使用与维护方法,具备对鲲鹏计算平台上的应用进行全生命周期管理的能力,能够胜任鲲鹏平台的应用开发和运维岗位。











## 简介

在 x86 上的可执行程序需要在基于鲲鹏处理器的平台上运行,需要经过源码的移植修改后方可在鲲鹏云服务器或泰山服务器上运行。x86 平台上所开发的容器镜像也是无法直接运行在鲲鹏平台上的,需要通过鲲鹏平台构建基于鲲鹏的镜像。本实验指导学员如何实现基于鲲鹏平台应用的源码迁移和基于鲲鹏平台的容器迁移。

## 内容描述

本实验指导包含两个部分, 云服务器环境上的应用迁移和容器应用的迁移。

实验一:从云环境准备开始,逐一介绍软件迁移的编译环境准备、源码获取、迁移分析、源码 修改和验证。

实验二:从 Docker 安装开始,逐一介绍 Docker 基础镜像的获取、镜像构建和镜像验证。

实验三:删除华为云资源

实验工具
------

名称	官网下载链接	用途
putty	https://www.putty.org/	远程登录工具
winscp	https://winscp.net/eng/download.php	远程传输工具
Porting Advisor 1.0	https://www.huaweicloud.com/kunpeng/ software/portingadvisor.html	代码迁移工具
PortingTest	https://hcia-kunpeng-application- developer.obs.cn-north- 1.myhwclouds.com/PortingTest.zip	代码迁移实验的源代 码文件



• 可选择以下任意一种服务器进行应用迁移实验,本实验以 x86 的弹性云服务器为例



推荐服务器	所需软件版本或规格
基于x86的弹性云服务器	CentOS 7.6 2vCPUs 4GB
基于鲲鹏计算的弹性云服务器	CentOS 7.6 2vCPUs 4GB
RH2288H V5	CentOS 7.8 GCC 4.8
TaiShan 200	CentOS 7.8 GCC 4.8

- 已注册并登陆华为云账号,账号中有一定金额供实验使用。
- 在本地已安装 putty 和 winscp 软件,并获取代码迁移工具和实验源码文件



## 目录

前	言	3
简介	7	. 3
内褶	客描述	. 3
实	金工具	. 3
实	检资源	. 3
1	源码迁移实验	6
1.1	实验介绍	. 6
1.2	准备虚拟私有云 VPC 环境	. 6
1.3	登录云服务器 ECS	. 9
1.4	Porting Advisor 应用迁移操作1	13
1.5	思考题2	20
2	容器迁移实验	2 <b>1</b>
2.1	实验介绍	21
2.2	登录云服务器 ECS	21
2.3	容器迁移操作	26
3	资源删除	3
3.1	删除弹性云服务器及相关资源	33



# ■ 源码迁移实验

## 1.1 实验介绍

## 1.1.1 关于本实验

本实验通过一个 C/C++的代码文件为例,实现用 Porting Advisor 工具对代码进行移植分析和修改建议,通过修改代码后,实现代码移植。

## 1.1.2 实验目的

- 理解 C/C++代码的基本原理。
- 掌握 Porting Advisor 工具进行迁移分析的方法。

## 1.2 准备虚拟私有云 VPC 环境

在后续创建 ECS 时,选择该步骤中所配置的 VPC、子网和安全组。

步骤 1 登录华为云账号, 在虚拟私有云 VPC 页面下, 点击"访问控制台"。

https://www.huaweicloud.com/product/vpc.html



步骤 2 在网络控制台界面中,点击"创建虚拟私有云"。



$\bigcirc$	虚拟私有云 ③				使用指南 + 创建虚拟私有云
网络控制台	IPv6双栈公测中,减整您立即体验:IPv6双	浅			
总览			名称	*	Q 标签搜索 ※ C ビ
虚拟私有云	名称	IPv4网段	状态	子网个数	操作

步骤 3 填写如下配置信息,然后点击"立即创建"。

基本信息

- 区域:华北-北京四
- 名称:myvpc
- 网段:默认

子网配置

- 可用区:可用区1
- 名称: subnet-myvpc
- 子网网段:默认

创建虚拟私有云	⑦ 《返回虚拟私有云列表
基本信息	
*区述	华北-北京四 ▼ 不同区域的资源之间内网不互通、读选择靠近您客户的区域,可以降低网络时延、提高访问速度。
*名称	vpс-туvpс
* 同段	192 · 168 · 0 · 0 / 16 ·
	<u>建议使用网段</u> : 10.0.0.0/8∼24, 172.16.0.0/12~24, 192.168.0.0/16~24
标签	如果您需要使用同一标签标识多种云资源,即所有服务均可在标签输入框下拉选择同一标签,建议在TMS中创建预定义标签。 <b>查看预定义标签</b> C
	标签键
	您还可以添加10个标签。
705199	
于网配宣	
默认子网	
*可用区 🕜	可用区1 可用区2 可用区3
*名称	subnet-myvpc
*子网网段	192     •     168     •     0     /     24     ▼       町田(P)取:251       子阿曾國主或語。子阿阿段无法修改
高级配置	默认配置 自定义配置



名称	IPv4网段	状态	子网个数	操作
vpc-headquater	192.168.0.0/16	正常	1个	修改删除
vpc-db-active	172.16.0.0/16	正常	2个	修改删除
my	192.168.0.0/16	正常	1个	修改删除
туурс	192.168.0.0/16	正常	1个	修改删除

## 步骤 5 在"安全组"页面中找到 Sys-default 安全组。点击"配置规则"。

Ó	安全组 ③					🗗 使用指南	+ 6	腱安全	组
网络控制台					名称 🔻			Q	С
总览	名称	安全组规则	关联实例	描述		操作			
虚拟私有云	SG-subnet-rds	7	1	SG-subnet-rds		配置规则	关联实例	更多,	
访问控制 ^	Sys-default 2	6	0	default	0	配置规则	关联实例	克隆	
• 安全组 🌗	SG-subnet-ftp	8	0	SG-subnet-ftp		配置规则	关联实例	更多,	
・网络ACL	SG-subnet-web	6	0	SG-subnet-web		配置规则	关联实例	更多,	

## 步骤 6 选择"入方向规则"页签,点击"添加规则"。

步骤 4 返回到网络控制台,可看见 VPC 与子网已创建。

基本信息 入方向规则 出方向规则 关联实例							
添加规则 快速添加规则 删除 一键放通	入方向规则: 4	教我设置					С
协议端口 🔽 🕐	类型	源地址 ⑦	描述	操作			
□ 全部	IPv4	Sys-default 🕐	-	修改	复制	删除	
TCP : 22	IPv4	0.0.0.0/0 ②	Permit default Linux SSH port.	修改	复制	删除	
TCP : 3389	IPv4	0.0.0.0/0 ⑦	Permit default Windows remote desktop port.	修改	复制	删除	
TCP : 1-65535	IPv4	0.0.0.0/0 ⑦		修改	复制	删除	

## 步骤 7 协议端口选择"全部放通",点击"确定",完成安全组规则的添加。

添加入方向规则 教我设置							
● 安全组入方向规则为白名单(允许),放通	① 安全组入方向规则为白名单(允许),放通入方向网络流量。						
安全组 Sys-default							
协议端口 ②	源地址 ②	描述	操作				
全部放通 ▼ 1-65535	IP地址     ・       0 · 0 · 0 · 0 / 0	]	复制删除				
	确定取消						



## 1.3 登录云服务器 ECS

步骤 1 登录华为云账号,在弹性云服务器 ECS 页面下,点击"立即购买"。

https://www.huaweicloud.com/product/ecs.html

总览 价格详情 	帮助文档 网站搭建指南
弹性云服	务器 ECS ▶
弹性云服务器(Elastic Clastic Clasti	oud Server ) 是一种可随时自助获取、可弹性伸缩的云服务器 , 帮助用户打造可靠、安全、灵
运、同X的应用环境 三年低至5折,多种配置可	「选
立即购买	价格计算器

步骤 2 填写如下基础配置信息,然后点击"下一步"。

- 计费模式:按需计费
- 区域:**华北-北京四**
- 可用区:随机分配
- CPU 架构: x86 计算
- 规格:通用计算型 s6.large.2|2vCPUs|4GB
- 镜像:公共镜像 CentOS7.6.64bit
- 系统盘:高IO|40G

计费模式	包年/包月 按需计费 克价计数	n (D					
区域	华北北京四						
可用区	小利在AmirAbishTal/用树用在小相思;制成ITB###USUSS	NSAL NACHABASE, SCAUNDEE.					
CPU架构	X86计算 截圆计算 ⑦						
规格	最新系列 vCPUs 全部	▼ 内存 全部	▼ 规格名称	Q			
	通用计算型 通用计算增强型 内存优化型	超大内存型 磁盘增强型	超高I/O型 GPU加速型 道用入	11型 ②			
	规格名称	vCPUs   内存 💠	CPU ¢	基准/最大带宽 ⑦ ≑	内网收发包 ⑦ 💠	规格参考价 ⑦	
	Sõ.small.1	1vCPUs   1GB	Intel Cascade Lake 2.6GHz	0.1/0.8 Gbit/s	100,000	¥0.07/小街	^
	o sõ.medium.2	1vCPUs   2GB	Intel Cascade Lake 2.6GHz	0.1/0.8 Gbit/s	100,000	¥0.17/j\8j	
	sõ.medium.4	1vCPUs   408	Intel Cascade Lake 2.6GHz	0.1/0.8 Gbit/s	100,000	¥0.34/小时	
	s6.large.2	2vCPUs   4GB	Intel Cascade Lake 2.6GHz	0.2/1.5 Gbit/s	150,000	¥0.36//j\8j	
	s6.large.4	2vCPUs   8GB	Intel Cascade Lake 2.6GHz	0.2/1.5 Gbit/s	150,000	¥0.68//j/8j	
	Ső.xlarge.2	4vCPUs   8GB	Intel Cascade Lake 2.6GHz	0.35/2 Gbit/s	250,000	¥0.73/小时	
	s6.xlarge.4	4vCPUs   16GB	Intel Cascade Lake 2.6GHz	0.35/2 Gbit/s	250,000	¥1.37/小时	
	o sö.2xlarge.2	SvCPUs   16GB	Intel Cascade Lake 2.6GHz	0.75/3 Gbit/s	500,000	¥1.40/j\8j	
	题 \$6.2xlarge.4 (已售量)	8vCPUs   32GB	Intel Cascade Lake 2.6GHz	0.75/3 Gbit/s	500,000	¥2.73/小时	$\sim$
	当前规格						



镜像	公共镜像	私有镜像	共享镜像	市场镜像	
	CentOS		i 64bit(40GB)	-	C
	□ 主机安全基础版	⑦ (基础版本限时免费)	1个月)		
系统盘	高10	• -	40 + GB IOPS	S上限1,440, IOPS突然	发上殿5,000 ⑦
	+ 增加一块数据盘	您还可以挂载 23 块碳值	t (云硬盘)		

步骤 3 填写如下网络配置信息,然后点击"下一步"。

- 网络:选择已创建的网络和子网,如 myvpc和 subnet-myvpc
- 安全组: Sys-default
- 弹性公网 IP:现在购买
- 规格:全动态 BGP
- 带宽类型:独享带宽
- 计费方式:按带宽计费
- 带宽:5 Mbit/s

网络	myvpc(192.168.0.0/16)     ▼     C     subnet-myvpc(192.168.0.0/24)     ▼     C     自动分配P地址       如需创建新的虚拟私有云,您可前往控制台创建。	▼ 可用私有IP数量247个 ⑦
扩展网卡	▲ 増加—块网卡 您还可以增加1块网卡	
安全组	Sys-default (入方向:TCP/22, 3389   出方向:- ) × ▼ C 新建安全組 ⑦ 入方向:TCP/22, 3389  出方向:-	
弹性公网IP	现在购买 使用已有 暂不购买 ⑦	
规格	全动态BGP 静态BGP	
带宽类型	独享带宽 共享带宽	
计费方式	按带宽计费 按流量计费 ⑦	
带宽	5 Mbit/s ■ 1 100 200 300 500 1,00∞,0	5 Mbit/s

## 步骤 4 填写如下高级配置信息,然后点击"下一步"。

- 云服务器名称:ecs-test
- 登录凭证:密码



- 密码/确认密码:XXXXX(按照密码规则自定义密码)
- 云备份:暂不购买

云服务器名称	ecs-test
	购买多台云服务器时,名称自动按序增加4位数字后缀。例如:输入ecs,从ecs-0001开始命名;若已有ecs-0010,从ecs-0011开始命名。
登录凭证	客码 密钥对 创建后设置
用户名	root
密码	请牢记密码,如忘记密码可登录ECS控制台重置密码。
(時代) (初日)	
104816/102111-3	
云备份	使用云备份服务,需购买备份存储库,存储库是存放服务器产生的备份副本的容器。
	现在购买 使用已有 暂不购买 ⑦
高级选项	现在配置

- 步骤 5 在确认配置界面,勾选"我已经阅读……"后,点击"立即购买"。
- 步骤 6 购买完成后,在云服务器控制台可以看到正常运行中的 ECS,记录弹性 IP 地址。

名称/ID	可用区	状态	规格/镜像	IP地址	计费模式	操作
ecs-test // 589ef938-4bfb-4825-83bb-23	可用区2	😏 运行中	1vCPUs   1GB   s6.small.1 CentOS 7.6 64bit	121.36.19.170 (弹性公网) 5 M 192.168.0.81 (私有)	按需计费	远程登录 更多▼

步骤 7 打开 putty, 输入弹性 IP 地址后, 点击 open。



🔀 PuTTY Configuration		<u> </u>			
Category:					
En Session	Basic options for your PuTTY session				
	Specify the destination you want to connect to				
	Host Name (or IP address)	Port			
Bell	121.36.19.170	22			
Features	Connection type: ◎ Raw ◎ Telnet ◎ Rlogin ● SS	H 🔘 Serial			
Appearance     Behaviour     Translation     Selection     Colours     Connection     Data     Proxy     Telnet     Rlogin     SSU	Load, save or delete a stored session Saved Sessions Default Settings	Load Save Delete			
Serial	Close window on exit:	lean exit			
About Help	Open	Cancel			

步骤 8 用 root 用户名 , 和之前设置的密码登录 ECS。



登录成功后按照以下步骤进行迁移实验。



## 1.4 Porting Advisor 应用迁移操作

## 1.4.1 安装 Porting Advisor 工具

步骤 1 从华为云鲲鹏社区中获取 Porting Advisor 工具,如 Porting-advisor-x86\_64-linux-XXX.tar.gz。

https://www.huaweicloud.com/kunpeng/software/portingadvisor.html

软件下载
Kunpeng CentOS 7.6软件包下载:立即下载
X86 CentOS 7.6软件包下载: 立即下载

步骤 2 打开 WinScp, 输入之前记录的 ECS 的 IP 地址, 使用 root 用户名, 和之前设置的密码登录。

	- L	
₩ 新建站点	会话 文件协议(F) SFTP ▼ 主机名(H) 121.36.19.170 用户名(U) root 保存(S) ▼	端口号(R) 22 ♥ 密码(P) ●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●
工具(T) ▼ 管理(M) ▼	●登录 ▼	关闭 帮助

步骤 3 把 Porting Advisor 工具拖到 ECS 中的 root 下。





## 步骤 4 返回 putty, 输入命令 "gcc -v", 检查 GCC 环境是否符合要求。



#### 步骤 5 执行如下命令, 解压文件包。

tar -zxvf Porting-advisor-XXX.tar.gz(使用 tab 键补齐)



#### 步骤 6 执行如下命令,进入解压后的华为鲲鹏代码迁移工具安装包目录。

cd Porting-advisor-x86\_64-linux-XXX (使用 tab 键补齐)

[root@ecs-test ~]# ls
Porting-advisor-beta-x86\_64-linux-1.0.5.tar.gz
Porting-advisor-x86\_64-linux-1.0.5
[root@ecs-test ~]# cd Porting-advisor-x86\_64-linux-1.0.5/
[root@ecs-test Porting-advisor-x86\_64-linux-1.0.5]#

#### 步骤 7 执行以下命令,安装华为鲲鹏代码迁移工具的 WEB 模式。

sh install.sh web



- 按规划输入 Web Server 的 IP 地址,然后按 Enter 键。如果使用默认 IP 地址,不需要输入 IP,直接按 Enter 键即可。默认的 IP 地址为操作系统的 IP 地址。
- 按规划输入 https 端口, 然后按 Enter 键。如果使用默认的端口, 不需要输入端口号, 直接按 Enter 键即可。默认端口为 8084。

回显信息如下:

```
[root@ecs-test ~]# cd Porting-advisor-x86_64-linux-1.0.5
[root@ecs-test Porting-advisor-x86_64-linux-1.0.5]# sh install.sh web
please enter ip address(default is all ip addresses):
defult ip address
Cent OS
please enter https port(default: 8084):
```

注意:如果存在"/opt/portadv"目录,需执行如下命令删除该目录。

rm -rf /opt/portadv

步骤 8 等待 3~5 分钟完成安装,出现以下显示,表示安装成功。



## 1.4.2 使用代码迁移工具进行代码移植

- 步骤 1 打开 WinSCP 工具,以 root 用户登录 CentOS 操作系统。
- 步骤 2 将源代码文件传到 CentOS 操作系统的"/opt/portadv/portadmin"路径下面。配置实验任务。



📗 portadmi 🝷 🚰 👻 🔽		🖻 🗖 🎧 🖉	), 查找文件	-
📓 下载 🔹 📝 编辑 🔹 🗙		■性 🗳 新建 🗸 🛛 🕒	- 4	
/opt/portadv/portadmin/				
名字	- 大小	已改变	权限	拥有者
<u>.</u>		20.	rwxr-xr-x	root
PortingTest.zip	424 KB	2000 CT 1000	rw-rr	root

## 步骤 3 执行如下命令,进入/opt/portadv/portadmin 路径下面。

cd /opt/portadv/portadmin

#### 步骤 4 执行如下命令,解压刚上传的源代码文件压缩包。

unzip PortingTest.zip

[root@ecs-test ~]# cd /opt/portadv/portadmin
[rootgeds-test portadmin]# 11
total 424
-rw-rr 1 root root 434134 Sep 29 15:02 PortingTest.zip
[root@ecs-test portadmin]# unzip PortingTest.zip
Archive: PortingTest.zip
creating: PortingTest/
creating: PortingTest/examples/
creating: PortingTest/examples/cmdline/
inflating: PortingTest/examples/cmdline/commands.c
inflating: PortingTest/examples/cmdline/commands.h
inflating: PortingTest/examples/cmdline/main.c
inflating: PortingTest/examples/cmdline/parse_obj_list.c
inflating: PortingTest/examples/cmdline/parse_obj_list.h
creating: PortingTest/examples/distributor/

步骤 5 打开本地 PC 机的浏览器,在地址栏输入 https://所记录的 ECS 的 EIP:端口号,例如: https://121.36.19.170:8084),按 "Enter"。





步骤 6 在华为鲲鹏代码迁移工具页面, 输入用户名和密码。

系统的默认用户名为 portadmin,默认密码为 Admin@9000

- 步骤 7 单击"登录"。首次登录 Web 的用户,系统提示修改默认密码。将密码改为 XXXXX (按照密码规则自定义密码)。
- 步骤 8 成功登录后,显示首页界面,界面右上角将显示登录的用户名。



#### 步骤 9 在创建分析任务区对以下参数进行配置。

- 源代码存放路径:PortingTest/
- 编译版本:GCC4.8
- 构建工具:make
- 编译命令:make
- 目标操作系统: CentOS 7.6
- 目标系统内核版本: v4.14



海华河专分政权		
初於17月94子力又時代王	PortingTest/	]
		6
	检查项目:编译器选项、编译器宏、汇编程序、内置函数、属性 	_
编译器版本	GCC 4.8	
构建工具	make 💌	
★编译命令	make	
目标操作系统	CentOS 7.6	
目标系统内核版本	v4.14	
	分析	

步骤 10 单击"分析",生成分析报告。

弹窗页面显示任务分析进度,分析完成后,自动跳转至"移植报告"界面。

移植把	器 移植	建议					
				2019/09/2	29 19:08:44		
	源代码存放路径编译命令	준 /opt/portadv/porta make	dmin/PortingTest/	编译器版本 目标操作系统	GCC 4.8 CentOS 7.6	构建工具 目标系统内核版本	make v4.14
	分析结果						
	E	依赖库SO文件	总数:0	, 需要移植: 0			
	•	需要移植的源文件	1				
	() ()	需要移植的代码行数	C/C++和	]Makefile源代码:20行;	汇编代码:0行		
				下载报告 (.csv)	下載报告 (.html)		

步骤 11 点击下载报告,可以保存移植报告到本地。

步骤 12 单击"移植建议"页签。进入"移植建议"界面,勾选"显示源代码"。点击"确认"。



移植报告	移植建议
	您的原始源代码将显示在该页面的左侧区域,建议修改的源代码将显
	示在右侧区域。右侧区域的内容仅为修改建议,不会影响原始源代
	7 <b>3</b> .
	▶ 显示源代码
	确认

步骤 13 选择 C/C++ Source File 查看到"建议源代码"。

	原始源代码	建议源代码
文件列表	~~~~~*·	34
	33	33
	34 #ifndef _MAIN_H_	34 #ifndef _MAIN_H_
search Q	36 #derine _mALM_N_	36 #define _mkim_n_
		37 + # Suggestion: Please remove the macro 'amd64' on Kung
		38 + # Description: Enable the AMD 64 bits instructions.
	37 #defineamdb4	59 #defineamdb4
Notes		41 + # Description: Enable the AMD 64 bits instructions.
	38 #undefand64	42 #undefsmd64
C/C++ Source File		43 + # Suggestion: HLE is not supported on Kunpeng platform,
/opt/portady/portadmin/P /main.h	39 #ifdaf ATONIC HIE ACOUTER	44 + # Description: Start Lock elision on a lock variable, M 45 #:FAAF ATOMIC HIE ACOUTER
/opt/portadi/portadi/int/mainin		46 + # Suggestion: HLE is not supported on Kunpeng platform,
🗄 .S File		47 + # Description: End lock elision on a lock variable. Men
	<pre>40 #elifATOMIC_HLE_RELEASE</pre>	48 #elifATOMIC_HLE_RELEASE
	41 #else 42 #andif	49 #eise 50 #endif
	43	51
		52 + # Suggestion: Please remove the macro 'code_model_sme
		53 + # Description: Enable compiler generate the binary with

#### 步骤 14 返回 putty 登录的 ECS 界面,进入到移植建议的代码 C/C++源文件下,此处为:

```
cd /opt/portadv/portadmin/PortingTest/examples/ip_pipeline
vi main.h
```

## 步骤 15 输入 i, 即可进入 main.h 文件编辑模式, 根据步骤 13 中的建议提示逐行修改源代码。

*注:此实验仅要求掌握工具使用,不涉及代码修改具体操作,不对代码进行验证,相关实验参考综合实验。* 

步骤 16 由于此实验仅要求掌握工具使用,不涉及代码修改操作,这里不对代码进行验证,后续综合实验部分会做更进一步练习,因此执行命令:wq!,保存并退出代码文件。



## 1.4.3 结果验证

步骤 1 重复 1.4.2 中步骤 5~步骤 10,对修改后的代码进行移植分析。如果分析结果中,需要移植的依赖库、代码源文件、代码函数均为零的话,则说明没有依赖项和移植修改项,代码修改完成。

分析结果		
E	需要移植的依赖库SO文件	0
	需要移植的源文件	0
:	需要移植的代码行数	C/C++和Makefile源代码:0行;汇编代码:0行

## 1.5 思考题

我们在迁移软件时,包含编译型语言的程序,也包含解释型语言的程序,请思考:

- 1. 编译型语言和解释型语言分别有哪些?
- 2. 不同语言代码迁移流程的区别点有哪些?



# **2** 容器迁移实验

## 2.1 实验介绍

## 2.1.1 关于本实验

本实验在拥有华为云账号,并且购买了鲲鹏服务器产品的前提下,通过 Docker 安装,构建基础镜像,根据基础镜像安装 Redis,验证 Redis 镜像等 4 个模块的实验练习和实训,让学员掌握容器的迁移指导。

## 2.1.2 实验目的

- 掌握 Docker 安装。
- 掌握 Docker 构建基础镜像。
- 掌握 Docker 根据基础镜像安装 Redis。
- 掌握验证构建的 Redis 镜像。

## 2.2 登录云服务器 ECS

步骤 1 登录华为云账号,在弹性云服务器 ECS 页面下,点击"立即购买"。

https://www.huaweicloud.com/product/ecs.html



总览 价格详 	情 帮助文档	网站搭建指南
弹性云肌	6务器 EC	S 🕟
弹性云服务器(Elastic 活、高效的应用环境	Cloud Server) 是一种可随	動自助获取、可弹性伸缩的云服务器,帮助用户打造可靠、安全、灵
三年低至5折,多种配置	雪可选	
立即购买	价格计算器	

步骤 2 填写如下基础配置信息,然后点击"下一步"。

- 计费模式:按需计费
- 区域:**华北-北京四**
- 可用区:随机分配
- CPU 架构: 鲲鹏计算
- 规格:鲲鹏通用计算增强型 kc1.large.2
- 镜像:公共镜像 CentOS7.6 64bit with ARM(40GB)
- 系统盘:高IO|40G

计费模式	包年/包月 按需计费 完修	介计费 ⑦					
区域	华北北京四						
	不同区域的云服务产品之间内网互不相通;请就近选择春近急	b业务的区域,可减少网络时延,提高访问速度。					
可用区	随机分配 可用区1 可加	用区2 可用区3 ⑦					
CPU架构	X86计算 <						
规格	最新系列 ▼ vCPUs 全部	▼ 内存 全部	▼ 规格名称	Q			
	能感過用计算增强型      能助内存优化型      ()	D					
	规格名称	vCPUs   内存 \$	CPU \$	基准/最大带宽 ⑦ 💠	内网收发包 ⑦ 💠	规格参考价 ⑦	
	kc1.small.1	1vCPUs   1GB	Huawei Kunpeng 920 2.6GHz	0.5/2 Gbit/s	200,000	¥0.12/小时	^
	kc1.large.2	2vCPUs   4GB	Huawei Kunpeng 920 2.6GHz	0.8/3 Gbit/s	300,000	¥0.30/小时	
	kc1.large.4	2vCPUs   8GB	Huawei Kunpeng 920 2.60Hz	0.8/3 Gbit/s	300,000	¥0.41//j\83	
	kc1.xlarge.2	4vCPUs   8GB	Huawei Kunpeng 920 2.6GHz	1.5/5 Gbit/s	500,000	¥0.60/小时	
	kc1.xlarge.4	4vCPUs   16GB	Huawei Kunpeng 920 2.6GHz	1.5/5 Gbit/s	500,000	¥0.81/rj\85	
	kc1.2xlarge.2	8vCPUs   16GB	Huawei Kunpeng 920 2.6GHz	3/7 Gbit/s	800,000	¥1.20/小时	
	kc1.2xlarge.4	8vCPUs   32GB	Huawei Kunpeng 920 2.6GHz	3/7 Gbit/s	800,000	¥1.63/小时	
	kc1.3xlarge.2	12vCPUs   24GB	Huawei Kunpeng 920 2.6GHz	4.5/9 Gbit/s	1,100,000	¥1.80/小街	
	kc1.3xlarge.4	12vCPUs   48GB	Huawei Kunpeng 920 2.0GHz	4.5/9 Gbit/s	1,100,000	¥2.44/小街	~
	当前规格 観翩週用计算增强型   kc1.large.2   2v	vCPUs   4GB					



镜像	公共镜像	私有镜像	共享镜像	市场镜像		
	CentOS	▼ CentOS 7.0	5 64bit with ARM(400	GB) 🔻	С	
系统盘	高10	• -	40 + GB IOP	S上限1,440 , IOPS突	上限5,000 ⑦	
	+ 增加块数据盘	您还可以挂载 23 块硫	藍盘 ( 云硬盘 )			

- 步骤 3 填写如下网络配置信息(选择实验1中已创建好的 VPC 和子网),然后点击"下一步"。
  - 网络:选择已创建的网络和子网,如 myvpc和 subnet-myvpc
  - 安全组: Sys-default
  - 弹性公网 IP:现在购买
  - 规格:全动态 BGP
  - 带宽类型:独享带宽
  - 计费方式:按带宽计费
  - 带宽:5 Mbit/s

网络	myvpc(192.168.0.0/16)     ・     C     subnet-myvpc(192.168.0.0/24)     ・     C     自动分配P地址     ・     可用私有IP数量247个 ②       如需创連新的虚拟私有云,認可前往控制台创建。
扩展网卡	+ 增加一块网卡 您还可以增加1块网卡
安全组	Sys-default (入方向:TCP/22, 3389   出方向: -)     ×     ▼     C 新建安全組 ⑦       入方向:TCP/22, 3389   出方向: -
弹性公网IP	现在购买         使用已有         暂不购买         ⑦
规格	全动态BGP 静态BGP
带宽类型	独享带宽 共享带宽
计费方式	按带宽计费 按流量计费 ⑦
带宽	5 Mbit/s

#### 步骤 4 填写如下高级配置信息,然后点击"下一步"。

- 云服务器名称:worker01
- 登录凭证:密码
- 密码/确认密码:XXXXX(按照密码规则自定义密码)



• 云备份:暂不购买

云服务器名称	worker01 分许重名
	购买多台云服务器时,名称自动按序增加4位数字后缀。例如:输入ecs,从ecs-0001开始命名;若已有ecs-0010,从ecs-0011开始命名。
登录凭证	<b>密码</b> 密钥对 创建后设置
用户名	root
密码	请牢记密码,如忘记密码可登录ECS控制台重置密码。
确认密码	
二友小	
乙寅四	
高级选项	现在配置

步骤 5 在确认配置界面,勾选"我已经阅读……"后,点击"立即购买"。

步骤 6 购买完成后,在云服务器控制台可以看到正常运行中的 ECS,记录弹性 IP 地址。

名称/ID	可用区	状态	规格/镜像	IP地址	计费模式	操作
worker01 🖉 39c79914-0e4d-4672-81	可用区2	😏 运行中	2vCPUs   4GB   kc1.large.2 CentOS 7.6 64bit with ARM	119.3.162.61 (弹性公网) 192.168.0.7 (私有)	按需计费	远程登录 更多▼

步骤 7 打开 putty, 输入弹性 IP 地址后, 点击 open。



🔀 PuTTY Configuration	? X
Category:	
Session Logging Terminal Keyboard Bell	Basic options for your PuTTY session         Specify the destination you want to connect to         Host Name (or IP address)         Port         119.3.162.61         22
	Connection type: Raw Telnet Rlogin SSH Serial Load, save or delete a stored session Saved Sessions
Connection	Default Settings Load Save Delete
About Help	Close window on exit: Always Never Only on clean exit Open Cancel

步骤 8 用 root 用户名 , 和之前设置的密码登录 ECS。

P root@worker01:~	-   0	x
login as: root root@119.3.162.61's password:		^
Welcome to Huawei Cloud Service		
[root@worker01 ~]#		
		-

登录成功后按照以下步骤进行容器迁移实验。



## 2.3 容器迁移操作

## 2.3.1 Docker 安装

步骤1 检查内核版本。

uname -r



说明: Docker 支持的 Centos 版本要求

- CentOS 7, 系统为 64 位、系统内核版本为 3.10 以上
- CentOS 6.5 或更高,系统为 64 位、系统内核版本为 2.6.32-431 或者更高版本

查出的 centos7 的系统为 64 位,内核版本为 4.14,符合版本要求。

#### 步骤 2 移除旧的版本。

```
yum remove docker docker-client docker-client-latest docker-common docker-
latest docker-latest-logrotate docker-logrotate docker-selinux docker-engine-
selinux docker-engine
```

[root@worker01 ~]# yum remove docker docker-client docker-client-latest docker-common docker-latest docker-latest-lo
grotate docker-logrotate docker-selinux docker-engine-selinux docker-engine
Loaded plugins: fastestmirror
No Match for argument: docker
No Match for argument: docker-client
No Match for argument: docker-client-latest
No Match for argument; docker-common
No Match for argument: docker-latest
No Match for argument: docker-latest-logrotate
No Match for argument: docker-logrotate
No Match for argument: docker-selinux
No Match for argument: docker-engine-selinux
No Match for argument: docker-engine
No Packages marked for removal

#### 步骤 3 安装 Docker 依赖工具。

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

[root@worker01 ~]# yum install -y yum-utils device-mapper-persistent-data lvm2				
Loaded plugins: fastestmirror				
Loading mirror speeds from cached hostfile				
epel/aarch64/metalink	5.2	kВ	00:00:00	)
* base: mirror.xtom.com.hk				
* epel: hkg.mirror.rackspace.com				
* extras: mirror.xtom.com.hk				
* updates: mirror.xtom.com.hk				
pase	3.6	kВ	00:00:00	)
docker-ce-stable	3.5	kВ	00:00:00	)
epel	5.4	kВ	00:00:00	)
extras	2.9	kВ	00:00:00	)
iRedMail	2.9	kВ	00:00:00	)
updates	2.9	kВ	00:00:00	)
(1/6): epel/aarch64/updateinfo	1.0	MB	00:00:01	
epel/aarch64/primary_db FAILED ] 16 kB/s	1.6	MB	00:11:46	6 ETA
https://mirror.preqi.net/epel/7/aarch64/repodata/f44e982437a73d1eff8eb69ca0d53674c4b7c8c08670	01ccc	6f51	a67f195c1	L96-pr
imary.sqlite.bz2: [Errno 12] Timeout on https://mirror.pregi.net/epel/7/aarch64/repodata/f44e	98243	7a73	d1eff8eb6	69ca0d
53674c4b7c8c0867001ccc6f51a67f195c196-primary.sqlite.bz2: (28, 'Operation too slow. Less than	1000	byt	es/sec tr	ansfe
rred the last 30 seconds')				
Trying other mirror.				
(2/6): extras/7/aarch64/primary_db	158	kВ	00:00:33	3
(3/6): epel/aarch64/primary_db	6.6	MB	00:00:06	5
updates/7/aarch64/primary_db FAILED ] 110 kB/s	8.5	MB	00:00:41	L ETA
105, http://mirror.0x.sg/centos-altarch/7.7.1908/updates/aarch64/repodata/f81f90e2fbf945e54225c913	ab58	9cd9	bc6ca1bd0	)09d57
0173fd3512a377e34-primary.sglite.bz2: [Errno 12] Timeout on http://mirror.0x.sg/centos-altarc	h/7.7	.190	8/updates	s/aarc



#### 步骤 4 安装 Docker。



## 步骤 5 启动 Docker 后台服务。

systemctl start docker

#### 步骤 6 测试运行 hello-world。

docker run hello-world



由于本地没有 hello-world 这个镜像,所以会下载一个 hello-world 的镜像,并在容器内运行。

#### 步骤 7 查看下载的 hello-world 镜像。

doalan imagaa

docker illages			
[root@worker0	1 ~]# docker images		
REPOSITORY	TAG	IMAGE ID	CREATED
SIZE			
hello-world	latest	de6f0c40d4e5	8 months ago
9./5KB			
F	4 1 #		



#### 第 28 页

## 2.3.2 Docker 构建基础镜像

#### 步骤1 查找 Docker Hub 上的 centos 镜像。

docker search centos

[root@worker01 ~]# docker searc NAME	ch centos DESCRIPTION	STARS	OFFICIAL
centos	The official build of CentOS.	5566	[ок]
ansible/centos7-ansible	Ansible on Centos7	123	
jdeathe/centos-ssh	CentOS-6 6.10 x86_64 / CentOS-7 7.6.1810 x86	112	
consol/centos-xfce-vnc [OK]	Centos container with "headless" VNC session	99	
centos/mysql-57-centos7	MySQL 5.7 SQL database server	62	
imagine10255/centos6-lnmp-php56	6 centos6-lnmp-php56	57	
tutum/centos	Simple CentOS docker image with SSH access	45	
centos/postgresql-96-centos7	PostgreSQL is an advanced Object-Relational	39	

#### 步骤 2 拉取官方的镜像,标签为 7。

docker pull arm64v8/centos:7

```
[root@worker01 ~]# docker pull arm64v8/centos:7
7: Pulling from arm64v8/centos
4856e02b0d50: Pull complete
Digest: sha256:df89b0a0b42916b5b31b334fd52d3e396c226ad97dfe772848bdd6b00fb42bf0
Status: Downloaded newer image for arm64v8/centos:7
docker.io/arm64v8/centos:7
```

#### 步骤 3 本地镜像列表里查到 REPOSITORY 为 arm 64v8/centos, 标签为 7 的镜像。

docker images arm64v8/centos:7

[root@worker01 ~	]# docker	images arm64v8/centos:7			
REPOSITORY	TAG	IMAGE ID	CREATED	SIZE	
arm64v8/centos		0cb4fb73950e	4 weeks ago	239MB	

## 步骤 4 使用镜像 arm64v8/centos:7 以交互模式启动一个容器,在容器内执行/bin/bash 命令。

docker run -it arm64v8/centos:7 /bin/bash

root@worker01 ~]# docker run -it arm64v8/centos:7 /bin/bash root@f4603f1f918a /]#

此时进入到容器内部,图示中 "f4603f1f918a" 为容器 ID。

#### 步骤 5 容器内安装 Redis 依赖包。

yum install wget gcc make libgcc gcc-c++ glibc-devel -y



[root@f4603f1f918a /]# yum install wget gcc make libgcc gcc-c++ glibc-devel -y		
Loaded plugins: fastestmirror, ovl		
Determining fastest mirrors		
* base: mirror.xtom.com.hk		
* extras: mirror.xtom.com.hk		
* updates: mirror.xtom.com.hk		
base	3.6 kB	00:00:00
extras	2.9 kB	00:00:00
updates	2.9 kB	00:00:00
(1/4): extras/7/aarch64/primary_db	158 kB	00:00:02
(2/4): updates/7/aarch64/primary_db	234 kB	00:00:02
(3/4): base/7/aarch64/group_gz	165 kB	00:00:03
base/7/aarch64/primary_db FAILED		
http://mirror-hk.koddos.net/centos-altarch/7.7.1908/os/aarch64/repodata/99dbead306f663ac0018	323b643792	4771b436f3b3542
c22a5c2508abb03818c9-primary.sqlite.bz2: [Errno 12] Timeout on http://mirror-hk.koddos.net/c	centos-alt	arch/7.7.1908/o
s/aarch64/repodata/99dbead306f663ac001823b6437924771b436f3b3542c22a5c2508abb03818c9-primary	.sqlite.bz	2: (28, 'Operat
ion too slow. Less than 1000 bytes/sec transferred the last 30 seconds') $^{ m l}$		
Trying other mirror.		
(4/4): base/7/aarch64/primary_db	4.8 MB	00:00:07
Resolving Dependencies		
> Running transaction check		
> Package gcc.aarch64 0:4.8.5-39.el7 will be installed		
> Processing Dependency: libgomp = 4.8.5-39.el7 for package: gcc-4.8.5-39.el7.aarch64		
> Processing Dependency: cpp = 4.8.5-39.el7 for package: gcc-4.8.5-39.el7.aarch64		
> Processing Dependency: libmpfr.so.4()(64bit) for package: gcc-4.8.5-39.el7.aarch64		
> Processing Dependency: libmpc.so.3()(64bit) for package: gcc-4.8.5-39.el7.aarch64		

步骤 6 输入 exit, 退出容器, 查看容器 id。

docker ps -a					
[root@worker01 ~]#	docker ps -a				
CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
NAMES					
f4603f1f918a	arm64v8/centos:7	"/bin/bash"	14 hours ago	Exited (0) 2 minutes ago	
busy_khorana					
7c69973c7050	hello-world	"/hello"	14 hours ago	Exited (0) 14 hours ago	
mystifying_brown					

由此可知,容器 ID 为 f4603f1f918a。

步骤 7 根据容器 ID 创建一个新的镜像作为 Redis 的基础镜像。

```
docker commit -a "huawei.com" -m "redis images" f4603f1f918a t_arm64v8/centos:7
其中:
```

- huawei.com: 提交的镜像作者
- f4603f1f918a:容器 ID
- redis images: 提交时的说明文字
- t\_arm64v8/centos:7:新生成的镜像名称

```
[root@worker01 ~]# docker commit -a "huawei.com" -m "redis images" f4603f1f918a t_arm64v8/centos:7
sha256:ac4b911562b0b8ec231cb4a4e216f17b524f90e088742f557c110de7456f0c25
[root@worker01 ~]#
```

步骤 8 查看新构建的基础镜像。

docker images

[root@worker01 ~]#	docker images			
REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
t_arm64v8∕centos		ac4b911562b0	48 seconds ago	385MB
arm64v8/centos		0cb4fb73950e	4 weeks ago	239MB
hello-world	latest	de6f0c40d4e5	8 months ago	4.75kB

由此可知,新镜像 t\_arm64v8/centos:7 构建成功。



## 2.3.3 Docker 根据基础镜像安装 Redis

#### 步骤1 创建 redis 目录。

```
mkdir -p ~/redis ~/redis/data
```

data 目录将映射为 redis 容器配置的/data 目录,作为 redis 数据持久化的存储目录。

步骤 2 进入创建的 redis 目录, 创建 Dockerfile。

cd redis vi Dockerfile

#### 步骤 3 输入 i,编辑如下内容到 Dockerfile 中。

```
FROM t_arm64v8/centos:7
WORKDIR /home
RUN wget http://download.redis.io/releases/redis-5.0.5.tar.gz && \
tar -xvzf redis-5.0.5.tar.gz && \
mv redis-5.0.5/ redis && \
rm -f redis-5.0.5.tar.gz
WORKDIR /home/redis
RUN make && make install
```

EXPOSE 6379 CMD ["redis-server"]

#### 步骤 4 完成后点击 "ecs",然后输入:wq 保存退出文档。

#### 步骤 5 通过 Dockerfile 创建 redis 镜像。

docker build -t t\_arm64v8/centos\_redis:5.05 .

```
[root@worker01 redis]# docker build -t t_arm64v8/centos_redis:5.05 .
Sending build context to Docker daemon 124.6MB
Step 1/7 : FROM t_arm64v8/centos:7
---> ac4b911562b0
Step 2/7 : WORKDIR /home
---> Running in 27e0d1db3f7f
Removing intermediate container 27e0d1db3f7f
---> 6d947ee18245
Step 3/7 : RUN wget http://download.redis.io/releases/redis-5.0.5.tar.gz && tar -xvzf redis-5.0.5.tar.gz &&
v redis-5.0.5/ redis && rm -f redis-5.0.5.tar.gz
---> Running in c4380adfd537
--2019-09-21 02:58:18-- http://download.redis.io/releases/redis-5.0.5.tar.gz
Resolving download.redis.io (download.redis.io)109.74.203.151
Connecting to download.redis.io (download.redis.io)190.74.203.151|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1975750 (1.9M) [application/x-gzip]
Saving to: 'redis-5.0.5.tar.gz'
```



emoving intermediate container ac44ef6d7d9a	
> 1040c9fb40a2	
tep 6/7 : EXPOSE 6379	
> Running in 4a8ec827e80a	
emoving intermediate container 4a8ec827e80a	
> 997030afae9d	
tep 7/7 : CMD ["redis-server"]	
> Running in cac3f91e9f32	
emoving intermediate container cac3f91e9f32	
> e732d1ff296b	
uccessfully built e732d1ff296b	
uccessfully tagged t_arm64v8/centos_redis:5.05	
root@worker01 redis]#	

#### 步骤 6 查看创建的 redis 镜像。

[root@worker01 redis]# do	ocker images			
REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
t_arm64v8/centos_redis	5.05	e732d1ff296b	21 minutes ago	542MB
t_arm64v8/centos		ac4b911562b0	58 minutes ago	385MB
arm64v8/centos		0cb4fb73950e	4 weeks ago	239MB
hello-world	latest	de6f0c40d4e5	8 months ago	4.75kB

说明 Redis 镜像创建成功,镜像名称:t\_arm64v8/centos\_redis,标签 5.05,镜像 ID e732d1ff296b。

## 2.3.4 验证 Redis 镜像

步骤1运行容器,执行 redis-server。

```
docker run -p 6379:6379 -v $PWD/data:/data -d t_arm64v8/centos_redis:5.05
redis-server --appendonly yes
```

命令说明:

- -p 6379:6379: 将容器的 6379 端口映射到主机的 6379 端口。
- -v \$PWD/data:/data:将主机中当前目录下的 data 挂载到容器的/data。
- redis-server --appendonly yes: 在容器执行 redis-server 启动命令,并打开 redis 持 久化配置。

[root@worker01 redis]# docker run -p 6379:6379 -v \$PWD/data:/data -d t\_arm64v8/centos\_redis:5.05 redis-server --appendonly ye s 566f60cc5f6fedc3ac7c92c09ea36d79c1a9880459141dec492eaac3c3637ddf

步骤 2 查看容器启动状态,记录容器 ID,下一步中会通过容器 ID 进入容器。

docker ps

[root@worker01	redis]# docker ps				
CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
	NAMES				
566f60cc5f6f	t_arm64v8/centos_redis:5.05	"redis-serverappe <sup>…</sup> "	About a minute ago	Up About a minute	0.0.0.0:6
379->6379/tcp	graciou <u>s</u> _hofstadter				

## 步骤 3 执行 redis-cli 命令连接到刚启动的容器。

```
docker exec -it 566f60cc5f6f redis-cli
127.0.0.1:6379> info
```



#### 其中 566f60cc5f6f 为容器 ID。

[root@worker01 redis]# docker exec -it 566f60cc5f6f redis-cli
127.0.0.1:6379> info
# Server
redis_version:5.0.5
redis_git_sha1:0000000
redis_git_dirty:0
redis_build_id:d66412af2962c379
redis_mode:standalone
os:Linux 4.14.0-115.5.1.el7a.aarch64 aarch64
arch_bits:64
multiplexing_api:epoll
atomicvar_api:atomic-builtin
gcc_version:4.8.5
process_id:1
run_id:db158c55dcb89559465f6f823cc7f054323b9d8a
tcp_port:6379
uptime_in_seconds:252
uptime_in_days:0
hz:10
configured_hz:10
lru_clock:8757623
executable:/home/redis/redis-server
config_file:

说明连接 redis-server 成功。

#### 步骤 4 使用 redis 容器。

```
127.0.0.1:6379> ping

PONG

127.0.0.1:6379> set runkey "hello redis"

OK

127.0.0.1:6379> get runkey

"hello redis"

127.0.0.1:6379> ping
```

```
PONG
127.0.0.1:6379> set runkey "hello redis"
OK
127.0.0.1:6379> get runkey
"hello redis"
127.0.0.1:6379>
```

说明:

- ping 返回 PONG 说明检测到 redis 服务已经启动。
- set runkey "hello redis":设置 runkey 值为"hello redis",返回 OK,说明设置成功。
- get runkey:获取 runkey 的值,返回"hello redis"说明与设置的相匹配。



# **3** 资源删除

## 3.1 删除弹性云服务器及相关资源

完成实验后请务必删除华为云上的收费资源,以免造成不必要的收费。找到创建的弹性云服务器 ECS,按照如下步骤进行删除。

步骤1 打开云服务器控制台,在需要删除的云服务器后面选择"更多>删除"。

王服今韻控制台       所相正行於志 (名称 (〇) (公弦弦 (〇) (〇) (〇) (〇) (〇) (〇) (〇) (〇) (〇) (〇)			弹性云服务器 ⑦				☞ 最新言	动态   🇗 幕助引导	₩ 购买弹性云服务器
白菜       有用区       秋水       和指摘       旧地山       计提根式       操作         時た云祭祭       2 c2600affb0214fb-95a1-9600       可用区1       0 运行中       1000U 5 1.68 [ 56 small.1 Certifo 7.6 4bit       100.00 1 (86) (5.5 mall.1 Certifo 7.6 4bit       100.00 1 (86) (8.5 mall.1 Certifo 7.6 4bit       100.00 1 (8.0 mall.1 Certifo 7.6 4bit <td>云服务器控制台</td> <td>Ì</td> <td>开机 关机 重置密码 更</td> <td>ž *</td> <td></td> <td>所有运行状态 🔻</td> <td>名称 🔻</td> <td>Q 标签搜索:</td> <td>× C Ľ I</td>	云服务器控制台	Ì	开机 关机 重置密码 更	ž *		所有运行状态 🔻	名称 🔻	Q 标签搜索:	× C Ľ I
P#社会報告報 F基注机 F基注机 F基注机 F基注机 F基金程450 F基金属450	总览	_	名称/ID	可用区	状态	规格/镜像	IP地址	计费模式	操作
中重正約         受更効格         受更効格           非金屋服务器         ご         型面でのののののののののののののののののののののののののののののののののののの	弹性云服务器	_	2c8dbadf-b021-4f8b-95a1-99fc0	可用区1	😏 运行中	1vCPUs   1GB   s6.small.1 CentOS 7.6 64bit	(弹性公网)	按需计费	远程登录 更多▼
#全電路名	专属主机	- 1							变更规格
安理血     第       专品分布式存在     1       专品分布式存在     1       特金房子     1       特金房子     1       特性伸缩     1       支配分器指     1       子電台     1       方面分子     1       方面分素合     1       方面分素合     1	裸金属服务器	- 1							创建镜像
中最分布式存储         単         単純年編         1 <t< td=""><td>云硬盘</td><td>~</td><td></td><td></td><td></td><td></td><td></td><td></td><td>重置密码</td></t<>	云硬盘	~							重置密码
	を届公布式友祥	~							重装系统
報金統分 弾性伸縮 寄閉対 示服分器相 ご當分 の 、 、 、 、 、 、 、 、 、	STREET FOR COMPANY	- 1							切换操作系统
理性伸缩     700       密明对     第       立部分器组     第       二部分器组     第       二部分器40     第       正報务器40     第       正報务器40     第       正報务器40     第	镜像服务	- 1							创建备份
密明対         第2           五銀券器組         第2           五銀券器組         第4回期           大銀券器         第2           五銀券器組         第2	弹性伸缩	~							71110
三部分報組     一     一     一     一     一     一     一     一     一     一     日     月     月     月     月     月     月     月     月     日 <t< td=""><td>密钥对</td><td>- 1</td><td></td><td></td><td></td><td></td><td></td><td></td><td>天川</td></t<>	密钥对	- 1							天川
二部份         ク         特包周期         购买用间配置         购买用间配置         更改实金组	云服务器组								删除
	云备份	9							转包周期
更改会会組	- 00 40 99 40 /0	0							购买相同配置
	乙胺旁羅首份	9							更改安全组

步骤 2 在弹出对话框中勾选"释放云服务器绑定的弹性公网 IP 地址"和"删除云服务器挂载的数据盘",然后点击"是"。



删除				×							
<ul> <li>删除云服务器会同时删除系统盘对应快照,确定要进行删除操作吗?</li> <li>删除的云服务器和磁盘无法恢复。云服务器删除完成后,对应的磁盘需要1分钟左右才能完成删除。此时不要对磁盘有任何操作,否则可能导致云服务器故障或磁盘删除失败,需要重新执行删除操作。</li> <li>删除云服务器时保留关联的云服务器备份,该备份继续收费,可在云服务器备份页面执行删除操作。</li> <li>未删除(释放)的弹性公网IP和数据盘会继续计费。</li> <li>删除数据每公同时删除数据每对应快路</li> </ul>											
■ 释放云服务器绑定的弹性公网IP地址											
名称		状态	备注								
ecs	-	😏 运行中									
		是	否								

## 步骤 3 查看到列表中已没有资源时,表示弹性云服务器已删除。

弹性云服务器 ⑦						4	最新动态 📋 🖻 帮助司	目导 │ □ 购买弹性云服务器		
开机 关机 重置密码	更多 ▼			所有运行状态	▼ 名称	•	Q 标签搜	素 ≫ C Ľ ⊡ 😑		
名称/ID	可用区	状态	规格/镜像		IP地址		计费模式	操作		
				<₽						
			暂无	表格数据						
# **HCIA-** Kunpeng Application

# Developer

# 应用部署与发布

# 实验指导手册

版本:1.0



华为技术有限公司

### 版权所有 © 华为技术有限公司 2019。 保留一切权利。

非经本公司书面许可,任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部,并不得以任何形式传播。

### 商标声明

### 

huawe和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标,由各自的所有人拥有。

### 注意

您购买的产品、服务或特性等应受华为公司商业合同和条款的约束,本文档中描述的全部或部分产品、服务 或特性可能不在您的购买或使用范围之内。除非合同另有约定,华为公司对本文档内容不做任何明示或暗示 的声明或保证。

由于产品版本升级或其他原因,本文档内容会不定期进行更新。除非另有约定,本文档仅作为使用指导,本 文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

# 华为技术有限公司

地址: 深圳市龙岗区坂田华为总部办公楼 邮编:518129

网址: http://e.huawei.com



# 华为认证体系介绍

华为认证是华为公司基于"平台+生态"战略,围绕"云-管-端"协同的新ICT技术架构,打造的ICT技术架构认证、平台与服务认证、行业ICT认证三类认证,是业界唯一覆盖 ICT (Information and Communications Technology 信息通信技术)全技术领域的认证 体系。

根据ICT从业者的学习和进阶需求,华为认证分为工程师级别、高级工程师级别和专家 级别三个认证等级。华为认证覆盖ICT全领域,符合ICT融合的技术趋势,致力于提供领先的 人才培养体系和认证标准,培养数字化时代新型ICT人才,构建良性ICT人才生态。

华为认证HCIA-Kunpeng Application Developer V1.0定位于培养与认证能够应用华为鲲鹏计算平台进行业务部署与迁移,性能测试与调优,常见解决方案设计与规划的工程师。

通过HCIA-Kunpeng Application Developer V1.0认证,您将掌握对鲲鹏计算平台的使用与维护方法,具备对鲲鹏计算平台上的应用进行全生命周期管理的能力,能够胜任鲲鹏平台的应用开发和运维岗位。



# **Huawei Certification**







# 简介

本书为 HCIA-Kunpeng Application Developer 认证培训教程,适用于准备参加 HCIA-Kunpeng Application Developer 考试的学员或者希望了解华为鲲鹏架构下的应用部署与发 布相关操作的读者。

# 内容描述

本实验指导书书共包含 3 个实验,从虚拟机创建开始,逐步介绍了在华为鲲鹏架构下的应用 部署与发布步骤。

- 实验一为开发环境搭建,通过华为公有云申请 ECS,搭建并验证 x86 编译服务器交叉编译环境。
- 实验二为 rpm 打包实验,本书以 Redis 打包 rpm 为例,介绍通过 Redis 源码进行 rpm 打包。
- 实验三云服务私有镜像的制作,通过前述的步骤,将已经创建的 ECS 制作成私有镜像。

# 读者知识背景

本课程为华为认证基础课程,为了更好地掌握本书内容,阅读本书的读者应首先具备以下基本条件:

- 具有基本的 Linux 操作命令,同时了解 rpm 概念。
- 了解 Redis 相关概念,并熟悉基本的 SQL 语句。

# 实验环境说明

组网说明

本实验环境部署在华为公有云上,需要创建 VPC 及子网,实验所涉及到的 ECS 在一个子网内。



### 环境需求介绍

### 为了满足实验需要,建议每套实验环境采用以下配置:

名称	版本	规格	用途
鲲鹏ECS	CentOS 7.6	kc1.large.2   2 vCPUs   4GB	用于验证可执行程序
x86 ECS	CentOS 7.6	s6.large.2   2 vCPUs   4GB	用于搭建编译环境
Putty	/	/	ssh登录服务器

实验拓扑



图3-1 实验拓扑



# 目录

前	言	3
简介		3
内容	描述	3
读者	知识背景	3
实验	环境说明	3
1 开	F发环境搭建	6
1.1 🗄	实验介绍	6
1.1.1	关于本实验	6
1.1.2	实验目的	6
1.2 🌶	购买云服务器	6
1.2.1	购买云服务器	6
1.2.2	测试验证1	0
1.3 L	inaro 交叉环境搭建1	0
1.3.1	Linaro 交叉环境搭建1	0
1.3.2	测试验证1	2
2 R	edis rpm 打包14	4
2.1	实验介绍1	4
2.1.1	关于本实验1	4
2.1.2	实验目的1	4
2.2 F	Redis rpm 打包	4
3 Li	nux 私有镜像制作1	7
3.1 🗄	实验介绍1	7
3.1.1	关于本实验1	7
3.1.2	实验目的1	7
3.2 3	环境数据准备1	7
3.3 🕯	意像制作1	9



# ▲ 开发环境搭建

# 1.1 实验介绍

### 1.1.1 关于本实验

本实验通过拥有华为公有云账号的前提下,选择一款适合自己的云服务器进行购买,在购买的 x86 云服务器上搭建交叉编译环境,并且编译生成 ARM 平台上的可执行代码。

### 1.1.2 实验目的

- 掌握云服务器购买流程。
- 掌握交叉编译的基本原理。
- 掌握 x86 服务器上交叉编译器安装。
- 验证交叉编译是否成功。

### 1.2 购买云服务器

### 1.2.1 购买云服务器

- 步骤 1 打开华为公有云 www.huaweicloud.com 网页,点击右上角"登录",在登录窗口中 输入账号密码登录华为公有云。
- 步骤 2 选择"产品" → "基础服务" → "虚拟私有云 VPC"。
- 步骤 3 点击"访问控制台",进入网络控制台 VPC 页签。在网络控制台 VPC 页签点击右上角 "创建虚拟私有云"选项。

Ø	虚拟私有云 ③				▶ 使用指南 + 创建虚拟私有云
网络控制台		所有项目 🔻	名称	Ŧ	Q 标签搜索 ≫ C □
总览	名称	IPv4网段	状态	子网个数 企业项目	操作
虚拟私有云	vpcd92d	172.16.0.0/16	正常	2个 default	修改删除



步骤 4 按照如下表格配置 VPC 属性,然后点击右下角"立即创建"。

参数	配置
区域	华北-北京四
名称	vpc-test
网段	192.168.1.0/24
子网可用区	可用区 1
子网名称	subnet-test
子网网段	192.168.1.0/24

步骤 5 展开网络控制台左侧列表的访问控制,选择"安全组",进入安全组页签。点击右上角 "创建安全组"。

(E)	安全组 ⑦				🖻 使用指南	+ 创建安全组
网络控制台				名称 🔻		QC
总览	名称	安全组规则	关联实例	描述	操作	
虚拟私有云	sg-775e	4	0	-	配置规则 关联	实例 更多▼
访问控制 ^	Sys-default	4	0	default	配置规则 关联	实例 克隆
・ 网络ACL						

步骤 6 在"创建安全组"窗口中配置模板选择为"开放全部端口",名称设置为"sg-test", 然后点击"确定",完成安全组的创建。

创建安全组		×
* 模板	开放全部端□ ▼	
<b>★</b> 名称	sg-test	
描述	开放全部端口有一定安全风险,请谨慎选择。	
查看模板规则	0/255	
	确定 取消	



步骤 7 展开网络控制台左侧导航栏的"弹性公网 IP 和共享带宽",选择"共享带宽",进入共享带宽页签。

	▼ 控制台 服务列表 ▼	收藏	Q	费用 资源	工单 企业	备案	支持 HC	IE-Cloud   🔤 😶
$\bigcirc$	共享带宽 ⑦							₩ 购买共享带宽
网络控制台	续费 转包年/包月				名称	Ŧ		QCĽ
总览	名称	带宽 (Mbi 计	V	计费方式 🏹	公网IP地域	Ł	操作	
虚拟私有云								
访问控制								
弹性公网IP和带宽								
・弾性公网IP				暂无数据				
<ul> <li>・ 共享帯宽</li> <li>・ 共享流量包</li> </ul>								

步骤 8 点击右上角"购买共享带宽",按照如下参数配置共享带宽参数,然后点击"立即购买",在订单详情界面点击"提交"。

参数	配置
计费模式	按需计费
区域	华北-北京四
计费方式	按带宽计费
带宽大小	5 Mbit/s
带宽名称	bandwidth-test

步骤 9 选择"服务列表" → "计算" → "弹性云服务器 ECS" ,进入云服务器控制台的弹性云服务器页签。

₩ 华为云	♀ 北京四	▼   控制	影台服务	列表▲ 收調	ق (	Q
最近访问的服务:	弹性云服务器 ECS	虚拟私有云 VPC	弹性公网IP	统一身份认证	服务 IAM 对象存储服	务对
计算		存储			网络	
弹性云服务器 E	CS	云硬盘			虚拟私有云 VPC	
云容器引擎 CCE		专属分布了	式存储		弹性负载均衡 ELB	
云容器实例 CCI		存储容灾服	<b>股务</b>		云专线 DC	



<b>姚</b> 华为云 ♀ 北京四	▼   控制台 SB务列表 ▼ 收藏 Q 费用 资源 工单 企业 备案 支持… HCIE-Cloud…   2 <sup>999</sup> ⊙
	弹性云服务器 ②
云服务器控制台	开机 关机 重置密码 更多 ▼
总览	新有运行状态 🔹 名称 🔹 🛛 Q 标签搜索 ※ C 🖸 🗐 🗮
弹性云服务器	名称/ID         可用区         状态         规格/镜像         IP地址         计费模式         操作
专属主机	
裸金属服务器	

步骤 10 点击"购买弹性云服务器",按如下参数分别购买 x86\_test 和 kp\_test 两台弹性云服务器。

参数	kp_test 配置	x86_test 配置	
计费模式	按需计费	按需计费	
区域	华北-北京四	华北-北京四	
CPU 架构	鲲鹏计算	X86 计算	
规格	kc1.large.2   2vCPUs   4GB	s6.large.2   2vCPUs   4GB	
公共镜像	Centos7.6 64bit with ARM	Centos7.6 64bit	
系统盘	高 I/O,40GB	高 I/0,40GB	
网络	vpc-test   subnet-test   手动分 配 IP 地址   192.168.1.20	vpc-test   subnet-test   手动分 配 IP 地址   192.168.1.10	
安全组	sg-test	sg-test	
弹性公网 IP	现在购买	现在购买	
弹性公网 IP 规格	全动态 BGP	全动态 BGP	
宽带类型	共享	共享	
宽带名称	bandwidth-test	bandwidth-test	
云服务器名称	kp_test	x86_test	
登录凭证	密码	密码	
用户名	root	root	
密码/确认密码	XXXXX(按照密码规则自定义密 码)	XXXXX(按照密码规则自定义密 码)	



云备份	暂不购买	暂不购买
-----	------	------

#### 步骤 11 购买完成后,点击"返回云服务器列表",查看购买的服务器状态信息。

	弹性云服务器 ③	◆最新动态   ●報助引导     中 购买弹性云酸券器
云服务器控制台	开机     美机     重置密码     更多 ▼	
总览	所有运行状态 ▼ 名称 ▼	Q 标签搜索 ≫ C ビ 回 ≔
弹性云服务器	名称/ID 可用区 状态 规格/镜像	IP地址 计费模式 操作
专属主机 裸金属服务器	kp_test b3a15cfc-cf98-452 可用区2 ● 运 2vCPUs   4GB   kc1.l CentOS 7.6 64bit wit	. 119.3.168.20 (弹性 . 192.168.1.20 (私有) 按需计费 远程登录 更多▼
云硬盘	×86.test ○ 5 运 2vCPUs   4GB   s6.la → b41a9df2-d7d4-4b 可用区1 ④ 运 CentOS 7.6 64bit	. 119.3.217.57 (弹性 192.168.1.10 (私有) 按需计费 远程登录 更多 ▼

### 1.2.2 测试验证

步骤 1 点击 kp\_test 弹性云服务器的"远程登录"。

ſ	名称/ID	可用区	状态	规格/镜像	IP地址	计费模式	操作
	kp_test b3a15cfc-cf98-452	可用区2	⊖ 运	2vCPUs   4GB   kc1.l CentOS 7.6 64bit wit	119.3.168.20 (弹性 192.168.1.20 (私有)	按需计费	远程登录 更多▼
	x86_test b41a9df2-d7d4-4b	可用区1	⊖ 运	2vCPUs   4GB   s6.la CentOS 7.6 64bit	119.3.217.57 (弹性 192.168.1.10 (私有)	按需计费	远程登录 更多 ▼

# 步骤 2 在弹出的 VNC 网页窗口中输入用户名 root 以及密码,弹出如下提示时,表示 ECS 正常。



步骤 3 按照上述两个步骤验证 x86\_test 服务器是否正常。

# 1.3 Linaro 交叉环境搭建

### 1.3.1 Linaro 交叉环境搭建

步骤1 参考2.2.2步骤1~步骤2,登录x86\_test 服务器。





### 步骤 2 执行如下命令,安装开发环境:

yum -y groupinstall Development Tools

### 步骤 3 在/usr/local/ 目录下创建 ARM-toolchain 目录。

mkdir /usr/local/ARM-toolchain

### 步骤 4 进入/usr/local/ARM-toolchain/目录,使用 wget 下载 gcc-linaro-5.5.0-2017.10x86\_64\_aarch64-linux-gnu.tar.xz。

cd /usr/local/ARM-toolchain/ wget https://releases.linaro.org/components/toolchain/binaries/latest-5/aarch64-linux-gnu/gcc-linaro-5.5.0-2017.10-x86\_64\_aarch64-linux-gnu.tar.xz Iroot0x86\_test ARM-toolchain1# wget https://releases.linaro.org/components/toolchain/binaries/latest aro-5.5.8-2017.10-x86\_64\_aarch64-linux-gnu.tar.xz --2019-10-12 11:27:39-- https://releases.linaro.org/components/toolchain/binaries/latest-5/aarch64-017.10-x86\_64\_aarch64-linux-gnu.tar.xz Resolving releases.linaro.org (releases.linaro.org)... 13.228.101.2041 Commecting to releases.linaro.org (releases.linaro.org)13.228.101.2041;443... connected. HTTP request sent, awaiting response... 302 Found Location: https://publishing-ap-linaro-org.s3.amazonaws.com/releases/components/toolchain/binaries/l c-linaro-5.5.0-2017.10-x86\_64\_aarch64-linux-gnu.tar.xz?Signature=696sZxdxCV202hkFiRwIbkdZ4V8x3D&Expi d=AKIAIELXV2RYMAHFUP7A ftollowing] --2019-10-12 11:27:42-- https://publishing-ap-linaro-org.s3.amazonaws.com/releases/components/toolc d=AKIAIELXV2RYMAHFUP7A ftollowing] --2809-10-12 11:27:42-- https://publishing-ap-linaro-org.s3.amazonaws.com/releases/components/toolc d=AKIAIELXV2RYMAHFUP7A Resolving publishing-ap-linaro-org.s3.amazonaws.com (publishing-ap-linaro-org.s3.amazonaws.com)... 5 Commecting to 'gcc-linaro-5.0.8-2017.10-x86\_64\_aarch64-linux-gnu.tar.xz' 3% [==> 13,419,

### 步骤 5 等待下载 100%完成后, 解压压缩包。

tar -Jxvf gcc-linaro-5.5.0-2017.10-x86\_64\_aarch64-linux-gnu.tar.xz # 修改目录名称

mv gcc-linaro-5.5.0-2017.10-x86\_64\_aarch64-linux-gnu linaro

#### 步骤 6 修改环境变量

vim /etc/profile

### 按 i 键进入插入模式,按上下左右键将光标移动到最后一行的最末尾,按回车添加新的一行,在新的一 行输入如下内容,然后按 esc 键退出插入模式,按 冒号+wq 键保存退出。

export PATH=\$PATH:/usr/local/ARM-toolchain/linaro/bin/

#### 步骤 7 更新环境变量



#### source /etc/profile

### 步骤 8 输入如下命令查看 aarch64-gcc 版本信息 , 显示如图所示信息表示配置正确。

aarch64-linux-gnu-gcc -v

[root@x86_test ARM-toolchain]# aarch64-linux-gnu-gcc -v
Using built-in specs.
COLLECT_GCC=aarch64-linux-gnu-gcc
COLLECT_LTO_WRAPPER=/usr/local/ARM-toolchain/linaro/bin//libexec/gcc/aarch64-linux-gnu/5.5.0/lto-wrapper
Target: aarch64-linux-gnu
Configured with: '/home/tcwg-buildslave/workspace/tcwg-make-release/builder_arch/amd64/label/tcwg-x86_64-build/target/aarch64-li
nux-gnu/snapshots/gcc.git~linaro-5.5-2017.10/configure' SHELL=/bin/bashwith-mpc=/home/tcwg-buildslave/workspace/tcwg-make-rel
ease/builder_arch/amd64/label/tcwg-x86_64-build/target/aarch64-linux-gnu/_build/builds/destdir/x86_64-unknown-linux-gnuwith-m
pfr=/home/tcwg-buildslave/workspace/tcwg-make-release/builder_arch/amd64/label/tcwg-x86_64-build/target/aarch64-linux-gnu/_build
/builds/destdir/x86_64-unknown-linux-gnuwith-gmp=/home/tcwg-buildslave/workspace/tcwg-make-release/builder_arch/amd64/label/t
cwg-x86_64-build/target/aarch64-linux-gnu/_build/builds/destdir/x86_64-unknown-linux-gnuwith-gnu-aswith-gnu-lddisable-l
ibmudflapenable-ltoenable-sharedwithout-included-gettextenable-nlsdisable-sjlj-exceptionsenable-gnu-unique-obj
ectenable-linker-build-iddisable-libstdcxx-pchenable-c99enable-clocale=gnuenable-libstdcxx-debugenable-long-lo
ngwith-cloog=nowith-ppl=nowith-isl=nodisable-multilibenable-fix-cortex-a53-835769enable-fix-cortex-a53-843419
with-arch=armv8-aenable-threads=posixenable-multiarchenable-libstdcxx-time=yeswith-build=sysroot=/home/tcwg-builds
lave/workspace/tcwg-make-release/builder_arch/amd64/label/tcwg-x86_64-build/target/aarch64-linux-gnu/_build/sysroots/aarch64-lin
ux-gnuwith-sysroot=/home/tcwg-buildslave/workspace/tcwg-make-release/builder_arch/amd64/label/tcwg-x86_64-build/target/aarch6
4-linux-gnu/_build/builds/destdir/x86_64-unknown-linux-gnu/aarch64-linux-gnu/libcenable-checking=releasedisable-bootstrap
enable-languages=c,c++,fortran,ltobuild=x8b_b4-unknown-linux-gnuhost=x8b_b4-unknown-linux-gnutarget=aarch64-linux-gnu
pref1x=/home/tcwg-bu1ldslave/workspace/tcwg-make-release/bu1lder_arch/amdb4/label/tcwg-x8b_b4-bu1ld/target/aarchb4-l1nux-gnu/
_build/builds/destdir/x86_b4-unknown-linux-gnu
Thread model: posix
gcc version 5.5.0 (Linaro GC 5.5-2017.10)

## 1.3.2 测试验证

### 步骤 1 在/tmp/目录下创建目录用于保存代码。

mkdir /tmp/test/

#### 步骤 2 进入/tmp/test/目录,创建 hello.c 文件,并按如下所示编写该文件。

```
cd /tmp/test/
vim hello.c
```

# 按 i 键进入插入模式,按如下所示输入信息,然后按 esc 键退出插入模式,按 冒号+wq 键保存退出。

```
#include <stdio.h>
int main(void)
{
printf("hello linux\n");
return 0;
}
```

#### 步骤 3 执行如下命令, 交叉编译。

aarch64-linux-gnu-gcc -o kp-hello hello.c

#### 步骤 4 执行编译后的脚本。

- ./kp-hello
- # 此时会有报如下错误

[root@x86\_test test]# sh kp-hello kp-hello: kp-hello: cannot execute binary file

#### 步骤 5 执行如下命令,将脚本拷贝至 kp\_test 服务器。

scp kp-hello 192.168.1.20:/tmp



# 依次输入 "yes" 以及 kp\_test 服务器 root 密码 , 完成拷贝。

### 步骤 6 参考 1.2.2 步骤 1~步骤 2, 登录 kp\_test 服务器。



### 步骤 7 执行如下命令,验证脚本执行。

/tmp/kp-hello

# 当出现如图所示信息时,表示执行成功。





# **2** Redis rpm 打包

# 2.1 实验介绍

### 2.1.1 关于本实验

本实验主要介绍利用 Redis 源码进行 rpm 包的打包操作方法。

- 2.1.2 实验目的
  - 能将 Redis 的源码包打包成 rpm 包。

# 2.2 Redis rpm 打包

步骤 1 参考 2.2.2 步骤 1~步骤 2, 登录 kp\_test 服务器。



步骤 2 安装 rpmbuild 工具。

yum -y install rpm-build

步骤 3 安装 wget 工具。

yum -y install wget

步骤 4 使用 wget 下载 Redis 源代码。

wget http://download.redis.io/releases/redis-4.0.9.tar.gz

步骤 5 等待进度达到 100%下载完成后将下载的压缩包拷贝到~/rpmbuild/SOURCES/目录下。



```
# 创建一个 rpmbuild 目录。此步骤会报错,属于正常现象。
rpmbuild -ba test
# 移动源码包到 SOURCES 目录。
mv redis-4.0.9.tar.gz ~/rpmbuild/SOURCES/
```

#### 步骤 6 进入~/rpmbuild/SPECS/目录,新建并按要求编辑 redis.spec 文件。

```
cd ~/rpmbuild/SPECS/
 vim redis.spec
 # 按 i 键进入插入模式, 删除原有内容, 再输入如下内容后, 按 esc 键退出插入模式, 再按 冒号+wq 保存
 退出。
Name: redis
Version: 4.0.9
Release: 1%{?dist}
Summary: This is a RedisDB
License: GPL
URL: https://redis.io
Source0: redis-4.0.9.tar.gz
BuildRequires: gcc
#Requires: zlib-devel,readline-devel
%description
RedisDB
%prep
%setup -q
%build
make %{?_smp_mflags}
%install
make install PREFIX=%{buildroot}%{ prefix}
install -p -D -m 644 %{name}.conf %{buildroot}%{_sysconfdir}/%{name}.conf
chmod 755 %{buildroot}%{_bindir}/%{name}-*
mkdir -p %{buildroot}%{ sbindir}
mv %{buildroot}%{ bindir}/%{name}-server %{buildroot}%{ sbindir}/%{name}-server
%clean
rm -rf %{buildroot}
%files
%defattr(-,root,root,-)
%{_bindir}/%{name}-*
%{ sbindir}/%{name}-*
%config(noreplace)%{ sysconfdir}/%{name}.conf
%changelog
```

#### 步骤 7 输入如下命令生成二进制版本的 rpm 包。

rpmbuild -bb redis.spec

步骤 8 输入 ls ~/rpmbuild/RPMS/aarch64 查看刚生成的 rpm 包, 若显示如图所示, 表示正常。



```
ls ~/rpmbuild/RPMS/aarch64
```

# 回显

[root@kp-test SPECS]# ls ~/rpmbuild/RPMS/aarch64/ redis-4.0.9-1.el7.aarch64.rpm redis-debuginfo-4.0.9-1.el7.aarch64.rpm

### 步骤 9 安装 Redis。

rpm -ivh ~/rpmbuild/RPMS/aarch64/redis-4.0.9-1.el7.aarch64.rpm

### 步骤 10 查看已安装的 Redis 客户端版本

redis-cli -v # 回显

[root@kp-test aarch64]# redis-cli -v redis-cli 4.0.9



# **3** Linux 私有镜像制作

# 3.1 实验介绍

### 3.1.1 关于本实验

本指导书以"通过云服务器创建 Linux 系统盘镜像"为例,指导用户如何制作 Linux 私有镜像。

### 3.1.2 实验目的

- 熟悉 Redis 数据库简单操作
- 熟悉私有镜像制作流程

# 3.2 环境数据准备

步骤 1 参考 2.2.2 步骤 1~步骤 2, 登录 kp\_test 服务器。



步骤 2 执行如下命令, 查看 Redis 服务端版本。

```
redis-server -v
# 回显
[root0kp-test aarch64]# redis-server -v
Redis server v=4.0.9 sha=00000000:0 malloc=jemalloc-4.0.3 bits=64 build=6f57f8d6fada4ec4
```

步骤 3 执行如下命令 , 启动 Redis。

redis-server # 此界面不要关闭



步骤 4 在云服务器控制台的弹性云服务器页签查看 kp\_test 服务器弹性 IP 地址。

	弹性云服务器 ②			<table-cell-columns> 最新动态 📔 🗗 帮助</table-cell-columns>	防日导 只购买弹性云服务器
云服务器控制台	开机 关机 重置	鸾码 更多 ▼			
总览		所有运行状态	▼ 名称 ▼	Q 标签携	☆ × C Ľ @ ≔
弹性云服务器	名称/ID	可用区 状态	规格/镜像	IP地址	计费模式 操作
专属主机 裡金属服务器	b3a15cfc-cf98-452	可用区2 🧿 运	2vCPUs   4GB   kc1.l CentOS 7.6 64bit wit	119.3.168.20 (弹性 192.168.1.20 (私有)	按需计费 远程登录 更多 ▼
云硬盘	x86_test b41a9df2-d7d4-4b	可用区1 😏 运	2vCPUs   4GB   s6.la CentOS 7.6 64bit	119.3.217.57 (弹性 192.168.1.10 (私有)	按需计费 远程登录 更多 ▼

步骤 5 打开自己电脑上的 putty 工具,将获取到的弹性 IP 地址写入"Host Name (or IP address)处",然后点击"Open"若有提示框,选择"yes",然后输入用户名 root 以及密码,登录 kp\_test 服务器。

	N PuTTY Configuration	? ×	
	Category:		
	Session	Basic options for your PuTTY session	
	Logging	Specify the destination you want to connect to	
	Keyboard	Host <u>N</u> ame (or IP address) <u>P</u> ort	
	Bell	22	
	Features	Connection type:	
	- Window	○ Ra <u>w</u> ○ <u>T</u> elnet ○ Rlogin	
	- Appearance	Load, save or delete a stored session	
	Translation	Sav <u>e</u> d Sessions	
	Selection		
	Colours	Default Settings	
	Connection		
	Data	Sa <u>v</u> e	
	Telnet	Delete	
	Riogin		
	Serial	Close window on exit:	
		○ Always ○ Never	
	About Help	<u>Open</u>	
	<u>A</u> bout <u>H</u> elp	<u>Open</u>	
🧬 root@kp	About Help		×
P root@kp	About Help	Open Cancel	×
Proot@kp login as: root@119.3	About Help -test:~ root .168.20's password: . Set Oct 12.2142151		×
Proot@kp login as: root@119.3 Last login	About Help -test~ root .168.20's password: : Sat Oct 12 21:21:51	Open         Cancel           -         -           2019 from 119.3.119.14         -	×
Proot@kp login as: root@ll9.3 Last login We	About Help -test:~ root .168.20's password: : Sat Oct 12 21:21:51 lcome to Huawei Cloud	Qpen Cancel - Cancel 2019 from 119.3.119.14 Service	×
Proot@kp login as: root@l19.3 Last login We	About Help -test:~ root 1.68.20's password: : Sat Oct 12 21:21:51 lcome to Huawei Cloud	Open         Cancel           -         -           2019 from 119.3.119.14         -           Service         -	×
Proot@kp login as: root@l19.3 Last login We [root@kp-t	About Help -test:~ root 1.68.20's password: : Sat Oct 12 21:21:51 lcome to Huawei Cloud est ~]#	Qpen Qancel — () 2019 from 119.3.119.14 Service	×
Proot@kp login as: root@l19.3 Last login We [root@kp-t	About Help -test:~ root .168.20's password: : Sat Oct 12 21:21:51 lcome to Huawei Cloud est ~]#	Open     Cancel       -     -       2019 from 119.3.119.14     -       Service     -	×
Proot@kp login as: root@119.3 Last login We [root@kp-t	About Help -test~ root .168.20's password: : Sat Oct 12 21:21:51 lcome to Huawei Cloud est ~]#	Qpen Qancel — — — — — — — — — — — — — — — — — — —	×
Proot@kp login as: root@ll9.3 Last login We [root@kp-t	About Help -test~ root .168.20's password: : Sat Oct 12 21:21:51 lcome to Huawei Cloud est ~]#	Qpen Gancel — 2019 from 119.3.119.14 Service	×
Proot@kp login as: root@l19.3 Last login We [root@kp-t	About Help -test:~ root .168.20's password: : Sat Oct 12 21:21:51 lcome to Huawei Cloud est ~]#	Open     Cancel       -     -       2019 from 119.3.119.14     -       Service     -	×
Proot@kp login as: root@119.3 Last login We [root@kp-t	About Help -test:~ root .168.20's password: : Sat Oct 12 21:21:51 lcome to Huawei Cloud est ~]#	Open         Cancel           -         -           2019 from 119.3.119.14         -           Service         -	×
Proot@kp login as: root@119.3 Last login We {root@kp-t	About Help -test:~ root 1.68.20's password: : Sat Oct 12 21:21:51 lcome to Huawei Cloud est ~]#	Qpen Gancel — 2019 from 119.3.119.14 Service Service	×
Proot@kp login as: root@ll9.3 Last login We [root@kp-t	About Help -test~ root .168.20's password: : Sat Oct 12 21:21:51 lcome to Huawei Cloud est ~]#	Qpen Qancel — — — — — — — — — — — — — — — — — — —	×
Proot@kp login as: root@ll9.3 Last login We [root@kp-t	About Help -test~ root .168.20's password: : Sat Oct 12 21:21:51 lcome to Huawei Cloud est ~]#	Qpen Gancel -  2019 from 119.3.119.14 Service	×
Proot@kp login as: root@l19.3 Last login We [root@kp-t	About Help -test~ root .168.20's password: : Sat Oct 12 21:21:51 lcome to Huawei Cloud est ~]#	Qpen Cancel -  2019 from 119.3.119.14 Service	×
Proot@kp login as: root@119.3 Last login We [root@kp-t	About Help -test:~ root .168.20's password: : Sat Oct 12 21:21:51 lcome to Huawei Cloud est ~]\$	Open       Cancel         -       -         2019 from 119.3.119.14       -         Service       -	×
Proot@kp login as: root@l19.3 Last login We [root@kp-t	About Help -test:~ root .168.20's password: : Sat Oct 12 21:21:51 lcome to Huawei Cloud est ~]#	Open         Cancel	×
Proot@kp login as: root@ll9.3 Last login We [root@kp-t	About Help -test~ root .168.20's password: : Sat Oct 12 21:21:51 lcome to Huawei Cloud est ~]#	Open       Cancel         -       -         2019 from 119.3.119.14       -         Service       -	×
Proot@kp login as: root@l19.3 Last login We [root@kp-t	About Help -test~ root .168.20's password: : Sat Oct 12 21:21:51 lcome to Huawei Cloud est ~]#	Open       Cancel         -       -         2019 from 119.3.119.14       -         Service       -	×

步骤 6 输入 redis-cli 连接数据库服务器。





### 步骤 7 输入 set huawei arm , 写入数据库。

```
127.0.0.1:6379> set huawei arm
ok
```

127.0.0.1:6379> set huawei arm OK

步骤 8 输入 get huawei, 获取 K=huawei 的值。

```
127.0.0.1:6379> get huawei
"arm"
```

127.0.0.1:6379> get huawei "arm"

步骤 9 按 Ctrl+D 退出 Redis 连接,同时,在 kp\_test 的 VNC 窗口中按 Ctrl+C,关闭 Redis 数据库。关闭 putty 窗口,关闭 kp\_test 服务器 VNC 窗口。

# 3.3 镜像制作

步骤1 在云服务器控制台界面的左侧导航栏,点击"镜像服务器",进入镜像服务页签。

	镜像服务 ⑦ + 创建私有额象
云服务器控制台	目前镜像服务已进入商业化阶段,私有镜像会收取一定的存储费用,删除创建的镜像后将不再计费。详细计费标准可参考镜像的计费标准
总览	
弹性云服务器	公共镜像 私有镜像 共享镜像 到云市场发布镜像。了解发布镜像流程,请单击帮助
专属主机	您还可以创建100个私有镜像。
裸金属服务器	删除 共学
云硬盘 ~	所有機像     ▼     名称     ▼     Q     标签搜索 ≫     C     □
专属分布式存储 >	名称 ◆ 状态 操作系 操作系统 镜像类型 磁盘容量 加密 创建时间 7 ◆ 操作
镜像服务	

步骤 2 点击"创建私有镜像",按照下表信息配置私有镜像参数,配置好后点击"立即创建"。在信息汇总页面确认信息无误后继续完成镜像创建,然后返回镜像列表。

参数	配置
区域	华北-北京四
创建方式	系统盘镜像
镜像源	云服务器   kp_test
镜像名称	image_kp_redis



### 步骤 3 等待镜像创建完成后,可以在镜像服务页签的私有镜像处看到已经创建的镜像。

	镜像服务 ② + 创建和有镜像
云服务器控制台	目前镜像服务已进入商业化阶段,私有镜像会收取一定的存储费用,删除创建的镜像后将不再计费。详细计费标准可参考镜像的计费标准
总览	
弹性云服务器	公共镜像   私有镜像   大享镜像
专属主机	删除
裸金属服务器	所有損像 ▼ 所有操作系统 ▼ 名称 ▼ Q 标签搜索 ※ C ビ
云硬盘 >	名称 \$ 状态 操作系 操作系统 镜像类型 磁盘容量 加密 创建时间 \7 \$ 操作
专属分布式存储 >	iamge_k  ◎正常 Linux CentOS ECS系统 40 否 2019/10/13 14:56 申请服务器 修改 更多 -
镜像服务	

### 步骤 4 进入弹性云服务器页签,点击"购买弹性云服务器"。

		弹性云服务器 ③				🗹 最新动态   🖻 帮問	胡导 📮	购买弹性云服务器
云服务器控制台		开机 关机 重置密	码具	18 ×				
总览			所有	有运行状态	▼ 名称 ▼	Q 标签搜	ģ ≫ C	[] [] []
弹性云服务器		名称/ID	可用区	状态	规格/镜像	IP地址	计费模式	操作
专属主机 裸金属服务器		b3a15cfc-cf98-45	可用区2	⊖ 运	2vCPUs   4GB   kc1.l CentOS 7.6 64bit wit	119.3.168.20 (弹性 192.168.1.20 (私有)	按需计费	远程登录 更多 ▼
云硬盘	~	x86_test b41a9df2-d7d4-4b	可用区1	⊖ 运	2vCPUs   4GB   s6.la CentOS 7.6 64bit	119.3.217.57 (弹性 192.168.1.10 (私有)	按需计费	远程登录 更多 ▼
オロハーサナが								

### 步骤 5 按照下表配置弹性云服务器参数,完成弹性云服务器的创建。

参数	kp_redis 配置		
计费模式	按需计费		
区域 华北-北京四			
可用区随机分配			
CPU 架构	鲲鹏计算		
规格	kc1.large.2		
镜像	私有镜像   image_kp_redis		
网络	vpc-test   subnet-test   自动分配 IP 地址		
安全组	sg-test		
弹性公网 IP 现在购买			



弹性公网 IP 规格	全动态 GGP
带宽类型	共享带宽
带宽名称	bandwidth-test
云服务器名称	kp_redis
登录凭证	密码
密码/确认密码	XXXXX(按照密码规则自定义密码)
云备份	暂不购买
高级选择	不勾选

步骤 6 返回弹性云服务器列表,等待 kp\_redis 创建完成后,点击"远程登录",输入用户名 root 及密码登录 kp\_redis 云服务器。输入 redis-server 启动 Redis。

kp-redis login: root Password: Last failed login: Mon Oct 14 08:46:20 CST 2019 on tty1 There was 1 failed login attempt since the last successful login. Last login: Sat Oct 12 22:48:30 from 119.3.119.14
Welcome to Huawei Cloud Service
[root@kp-redis ~]# redis-server 4199:C 14 Oct 08:46:29.443 # 000000000000 Redis is starting o0000000000000 4199:C 14 Oct 08:46:29.443 # Redis version=4.8.9, bits=64, commit=00000000, modified=0, pid=4199, just started 4199:C 14 Oct 08:46:29.443 # Marning: no config file specified, using the default config. In order to specify a config file use redis-server /path/to/redis.conf
Redis 4.8.9 (80000000/0) 64 bit         Running in standalone mode         Port: 6379         PID: 4199         Homory of the standalone mode         Pid: 100 / 10
4199:M 14 Oct 08:46:29.444 # Server initialized 4199:M 14 Oct 08:46:29.444 # WARRING overcommit_memory is set to 0! Background save may fail under low memory condition. To fix this issue add 'vm.overcommit_memory = 1' to /etc/sysctl.conf and then reboot or run the command 'sysctl vm.overcommit_memory=1' for this to take effect. 4199:M 14 Oct 08:46:29.445 # WARNING you have Transparent Huge Pages (THP) support enabled in your kernel. This will create late ney and memory usage issues with Redis. To fix this issue run the command 'echo never > /sys/kernel/mm/transparent_hugepage/enal led' as root, and add it to your /etc/rc.local in order to retain the setting after a reboot. Redis must be restarted after THP is disabled. 4199:M 14 Oct 08:46:29.445 * Ready to accept connections

步骤 7 参考 4.2 步骤 5,使用自己电脑上的 putty 远程登录到 kp\_redis 云服务器。

步骤 8 输入如下命令,连接 Redis。

redis-cli

步骤 9 输入 set huawei arm, 然后再输入 get huawei, 查询 key=huawei 的值, 验证 Redis 服务是否正常。



127.0.0.1:6379> set huawei arm
ok
127.0.0.1:6379> get huawei
 "arm"
127.0.0.1:6379> set huawei arm
oK

127.0.0.1:6379> get huawei "arm"

# HCIA-Kunpeng Application Developer

# 华为鲲鹏平台应用软件移植调优

# 综合实验指导手册

版本:1.0



华为技术有限公司

### 版权所有 © 华为技术有限公司 2019。 保留一切权利。

非经本公司书面许可,任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部,并不得以任何形式传播。

### 商标声明

### 注意

您购买的产品、服务或特性等应受华为公司商业合同和条款的约束,本文档中描述的全部或部分产品、服务 或特性可能不在您的购买或使用范围之内。除非合同另有约定,华为公司对本文档内容不做任何明示或暗示 的声明或保证。

由于产品版本升级或其他原因,本文档内容会不定期进行更新。除非另有约定,本文档仅作为使用指导,本 文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

# 华为技术有限公司

地址: 深圳市龙岗区坂田华为总部办公楼 邮编:518129

网址: http://e.huawei.com



# 华为认证体系介绍

华为认证是华为公司基于"平台+生态"战略,围绕"云-管-端"协同的新ICT技术架构,打造的ICT技术架构认证、平台与服务认证、行业ICT认证三类认证,是业界唯一覆盖ICT (Information and Communications Technology 信息通信技术)全技术领域的认证体系。

根据ICT从业者的学习和进阶需求,华为认证分为工程师级别、高级工程师级别和专家 级别三个认证等级。华为认证覆盖ICT全领域,符合ICT融合的技术趋势,致力于提供领先的 人才培养体系和认证标准,培养数字化时代新型ICT人才,构建良性ICT人才生态。

华为认证HCIA-Kunpeng Application Developer V1.0定位于培养与认证能够应用华为鲲鹏计算平台进行业务部署与迁移,性能测试与调优,常见解决方案设计与规划的工程师。

通过HCIA-Kunpeng Application Developer V1.0认证,您将掌握对鲲鹏计算平台的使用与维护方法,具备对鲲鹏计算平台上的应用进行全生命周期管理的能力,能够胜任鲲鹏平台的应用开发和运维岗位。









# 简介

本书为 HCIA-Kunpeng Application Developer 认证培训教程,适用于准备参加 HCIA-Kunpeng Application Developer 考试的学员或者希望了解华为鲲鹏平台 PostgreSQL 相关性能测试、性能调优以及软件打包和安装等相关技术的读者。

# 内容描述

本实验以数据库 PostgreSQL 为例,介绍在华为鲲鹏平台上对应用软件进行软件移植、性能测试、性能调优等全流程的综合实验。



本实验指导书共包含 8 个实验,从搭建华为鲲鹏平台实验环境开始,逐一介绍了 PostgreSQL 通过源码移植到华为鲲鹏平台的配置与实现。

- 实验一为通过购买 3 台华为云 ECS 云服务器搭建华为鲲鹏平台实验环境。
- 实验二为通过 Porting Advisor 工具对 PostgreSQL 源码包进行移植分析,并按修改建议 对源码进行修改实现代码移植。
- 实验三为在华为鲲鹏平台上通过源码安装部署 BenchmarkSQL。
- 实验四为在华为鲲鹏平台上安装性能测试工具 BenchmarkSQL。
- 实验五为通过性能测试工具 BenchmarkSQL 对 PostgreSQL 进行性能测试。
- 实验六为通过分析性能测试结果对 PostgreSQL 进行性能调优。
- 实验七为通过 rpmbuild 工具在华为鲲鹏平台上制作 PostgreSQL 的 RPM 安装包。



实验八为对制作完成的 PostgreSQL 的 RPM 安装包进行安装测试,完成 PostgreSQL 的 快速安装部署。

# 读者知识背景

本课程为华为认证基础课程,为了更好地掌握本书内容,阅读本书的读者应首先具备以下基本 条件:

• 具有基本的云服务基础知识以及 Linux 操作系统基础知识。

# 实验环境说明

组网说明

本实验环境面向准备 HCIA-Kunpeng 考试的鲲鹏应用开发工程师。每套实验环境包括 3 台华 为云 ECS 云服务器。每套实验环境适用于 1 名学员同时上机操作。

设备介绍

为了满足 HCIA-Kunpeng 实验需要,建议每套实验环境采用以下配置:

3 台华为云 ECS 云服务器及其配置、OS 版本的对应关系如下:

表3-1	设备参数表
------	-------

设备名称	云主机名称	配置	OS版本	备注
x86云服务器	ecs_postgre_s can_test	c6.large.2 2vCPUs   4GB	CentOS 7.6 64bit	x86架构 , 用于安装部署 Porting Advisor源码移植工 具 , BenchmarkSQL性能测试 工具。
鲲鹏云服务器	ecs_postgre_t est	kc1.2xlarge.2 8vCPUs   16GB	CentOS 7.6 64bit	鲲鹏架构,用于部署数据库 PostgreSQL,并进行性能测试 和性能调优。
鲲鹏云服务器	ecs_postgre_r pm_test	kc1.large.2 2vCPUs   4GB	CentOS 7.6 64bit	鲲鹏架构,用于安装部署 rpmbuild工具,制作 PostgreSQL的RPM安装包, 并对安装包进行安装测试。



# 实验拓扑



实验拓扑说明:

在华为云上购买的 3 台 ECS 云服务器之间网络互通 , 且与 PC 之间也网络互通。

# 准备实验环境

在实验开始之前,请每组学员确认是否已获取华为云账号和密码。



# 目录

前	言	3
简介		3
内容	描述	3
读者	知识背景	4
实验	环境说明	4
实验	拓扑	5
准备	实验环境	5
1 挠	<b>建华为鲲鹏平台实验环境</b>	9
1.1 🗄	实验介绍	9
1.2 ī	前提条件	9
1.3	登录华为云	9
1.3.1	操作步骤	9
1.4 🌶	购买华为云 ECS 云服务器1	1
1.4.1	操作步骤1	1
1.5 (	修改安全组规则1	17
1.5.1	操作步骤1	.7
2 ∄	丧装 Porting Advisor 代码移植工具	9
2.1 🗄	实验介绍1	9
2.2 ī	前提条件1	9
2.3 🗄	安装 Porting Advisor1	9
2.3.1	操作步骤1	9
3 Po	orting Advisor 移植 PostgreSQL2	4
3.1	实验介绍	24
3.2 ī	前提条件	24
3.3 🔅	分析 PostgreSQL 源码2	24
3.3.1	操作步骤2	24
3.4 🗧	升级 GCC 编译器版本	29



3.4.1 操作步骤	
3.5 移植 PostgreSQL	
3.5.1 操作步骤	
4 安装 BenchmarkSQL 性能测试工具	
实验介绍	
前提条件	
4.1 安装 jdk	
4.1.1 操作步骤	
4.2 安装 BenchmarkSQL	
4.2.1 操作步骤	
5 PostgreSQL 性能测试	
5.1 实验介绍	40
5.2 前提条件	40
5.3 创建数据库 tpcc	40
5.3.1 操作步骤	40
5.4 PostgreSQL 性能测试	41
5.4.1 操作步骤	41
6 PostgreSQL 性能调优	
6.1 实验介绍	
6.2 前提条件	46
6.3 PostgreSQL 性能调优	46
6.3.1 操作步骤	46
7 PostgreSQL 软件打包	
7.1 实验介绍	
7.2 前提条件	
7.3 制作 RPM 包	
7.3.1 操作步骤	
8 PostgreSQL 软件包安装测试	54
8.1 实验介绍	54
8.2 前提条件	54
8.2.1 RPM 包安装测试	54





# 搭建华为鲲鹏平台实验环境

## 1.1 实验介绍

本实验通过购买华为云 ECS 云服务器搭建华为鲲鹏平台实验环境,主要包含三台华为云 ECS 云服务器:

- 一台是基于 x86 架构,用于安装部署 Porting Advisor 源码移植工具,BenchmarkSQL性能测试工具。
- 一台是基于鲲鹏架构,用于部署数据库 PostgreSQL,并进行性能测试和性能调优。
- 一台是基于鲲鹏架构,用于安装部署 rpmbuild 工具,制作 PostgreSQL 的 RPM 安装包, 并对安装包进行安装测试。

# 1.2 前提条件

• 已获取登录华为云的账号、用户名和密码。

# 1.3 登录华为云

### 1.3.1 操作步骤

步骤 1 使用 PC 上的浏览器访问华为云官网:https://www.huaweicloud.com/?locale=zh-cn, 单击页面右上角的"登录",进入华为云账号登录页面。





步骤 2 单击右下角的"IAM 用户登录 (中国站)",进入华为云 IAM 用户登录页面。

账号登录 (「	中国站)
账号名/邮箱	
密码	<i>م</i> يرخ
手机号登录	
记住登录名	忘记密码?
	登录
免费注册	IAM用户登录 (中国站) ⑦
	使用其他账号登录 ~

步骤 3 输入账号名,用户名和密码,单击下方的"登录",登录华为云官网。

IAM用户登录 (中国站)			
账号名			
用户名/邮箱/手机			
密码	~~~~~		
记住登录名	忘记密码?		
登录			
免费注册	账号登录 (中国站)		
——————————————————————————————————————	V		

--结束


# 1.4 购买华为云 ECS 云服务器

## 1.4.1 操作步骤

步骤1 在华为云首页,单击右上角的"控制台",进入控制台操作页面。



步骤 2 在页面左上角,选择区域"北京四",单击"服务列表",选择"计算 > 弹性云服务器 ECS", 进入弹性云服务器列表页面。

<b>柴</b> 为云	♀ 北京四	▼   控制台	服务列表▲	收藏		Q	费用	资源	工单	企业	备案	支持与服务	
最近访问的服务: 弹性	主云服务器 ECS	统一身份认证服务 IAM	镜像服务 IMS	对象存储服务	Anti-DDoS流量清洗	虚拟私有云 VPC					请输入名称	网诸功能查找服	傍
计算		存储			网络		5	数据库				安全	
弹性云服务器 ECS		云硬盘			虚拟私有云 VPC			云数据库 RD	IS			Anti-DDoS8	托量清洗
云容器引擎 CCE		专属分布式得	字储		弹性负载均衡 ELB			文档数据库服	阪务 DDS			DDoS高防朋	铸
云容器实例 CCI		存储容灾服务	5		云专线 DC		:	分布式数据的	车中间件 DE	M		Web应用防	火墙 WAF
裸金属服务器 BMS		云服务器备	÷		虚拟专用网络 VPN		į	数据复制服务	5 DRS			漏洞扫描服	务
云手机, CPH		云备份			云解析服务 DNS		1	数据管理服务	号 DAS			企业主机安	全
镜像服务 IMS		云硬盘备份			NAT网关							容器安全服	务
批处理服务		对象存储服务	5		弹性公网IP		1	应用服务				数据库安全	服务 DBSS
函數工作流 Function	Graph	数据快递服务	cia		云连接 CC		1	应用管理与道	Ξ维平台 Se	rviceStag	e	数据加密服	务
弹性伸缩 AS		弹性文件服务	ni		VPC 终端节点		1	应用编排服务	e Aos			安全专家服	务
专属云		CDN					1	容器镜像服务	号 SWR			态势感知	
专属主机		专属企业存储	諸服务		管理与部署		4	微服务引擎(	CSE			SSL证书管理	里

步骤 3 单击页面右上角的"购买弹性云服务器"。

	弹性云服务器 ②				✓ 最新动态	▶ 帮助引导	<b>只</b> 购买弹性云服务器
云服务器控制台	开机 关机 重置密码	更多 👻	所有运	⊇行状态 ▼ 名称	- Q	标签搜索 ≫	C [] 🗉 🗮
总览	名称/ID	可用区 物	状态 规	见格/镜像	IP地址	计费模式	操作
弹性云服务器							

步骤 4 进入弹性云服务器的基础配置页面,基础配置(X86 计算,通用计算增强型, c6.large.2 2vCPUs | 4GB, CentOS 7.6 64bit ) 如下图所示:



性云服务器 く 🤉	<b>国云服务器列表</b> 自定义购买	快速购买 New!				
1 基础配置		2 网络配置		3 高级配置		4 确认面
计费模式 区域	包年/包月 <b>华北北京四</b> 本国区域的示影者产品之间达超互			行动事度		
可用区	随机分配	1用区1 可用区2	可用区3	0		
CPU架构	X86计算 鲲鹏计算	0				
规格	最新系列	▼ vCPUs 全部	▼ 内存	全部	▼ 规格名称	Q
	通用计算型 通用计算	增强型内存优化型	超大内存型 磁盘增	强型 超高1/0型	GPU加速型 通用	月入门型 ②
	规格名称	vCPUs   内存 💲	CPU ‡	基准 / 最大带宽 ②	\$ 内网收发包 ⑦ \$	规格参考价 ②
	c6.large.2	2vCPUs   4GB	Intel Cascade Lake 3.0GHz	1.2/4 Gbit/s	400,000	¥0.46//]\85
	C6.large,4	2vCPUs   8GB	Intel Cascade Lake 3.0GHz	1.2/4 Gbit/s	400,000	¥0.71/小时
	C6.xlarge.2	4vCPUs   8GB	Intel Cascade Lake 3.0GHz	2.4/8 Gbit/s	800,000	¥0.91//J\85
	C6.xlarge.4	4vCPUs   16GB	Intel Cascade Lake 3.0GHz	2.4/8 Gbit/s	800,000	¥1.42/小时
	C6.2xlarge.2	8vCPUs   16GB	Intel Cascade Lake 3.0GHz	4.5/15 Gbit/s	1,500,000	¥1.83/小时
	0 06 2xlarco 4	SUCDI IN LOOG	Intel Cascade Lake	4 5/15 Chit/o	1 500 000	NO 85//hB+
镜像	当前规格 <b>通用计算增强</b> 公共编像 私	<b>理   c6.large.2   2vCPUs   4</b> 有镜像	IGB 市场镜像			
	CentOS 👻	CentOS 7.6 64bit(40GB)	•	С		
	🗌 主机安全基础版 🕲 (基)	出版本限时免费6个月)				
系统盘	高IO + 増加一块数据盘 忽还可		3 IOPS上限1,440, IOPS突	发上限5,000 ②		
买量 — 1	+ 台 配置费用 ¥0.4790	5 /小时 新弟为准 了解计费详结				下一步:网络

步骤 5 单击"下一步:网络配置"。

步骤 6 进入弹性云服务器的网络配置页面,网络配置如下图所示:



弹性云服务器(	返回云服务器列表 自定义购买 快速购买 New!		
✓ 基础配置	2 网络配置	3 高级配置	4 确认配置
虚拟私有云	default_vpc C ⑦ 您猫一次使用华为云服务,系统将自动为您创建一个虚拟私有云,如果您希	還定制,您也可以到虛拟私有云影务中配置资源。	
网卡	default-subnet 💌		
安全组	default_securitygroup     ×     ●       请勾选要开通的协议/编口     60编口     443鐃口     ICMP协	议 🗹 22. 3389皖口 🕐	
弹性公网IP	現在购买使用已有 暂不购买	0	
规格	全动态BGP 静态BGP		
带宽类型	独享带宽 共享带宽		
计费方式	按带宽计费 技流量计费 ⑦		
带宽	5 Mbit/s	200 300 5 Mbi	t/s
9h157-84		80,68	
购买量 — 1	+ 台	ÖU/GB	上一步 下一步: 高级配置

步骤7 单击"下一步:高级配置"。

SM LL - PR As DD

步骤 8 进入弹性云服务器的高级配置页面,高级配置如下图所示:

Property 11/ Day 377 Los 100-Day 377

- 云服务器名称: ecs\_postgre\_scan\_test
- 密码:XXXXX(按照密码规则自定义密码),并再次确认密码

✓ 基础配置 ──	▶ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■	3 高级配置	4 确认配置
云服务器名称	ecs_postgre_scan_test		
登录凭证	密码 密钥对 创建后设置		
用户名	root		
密码	请半记密码,如忘记密码可登录ECS控制台重置密码。		
确认密码			
云备份	使用云备份服务,需购买备份存储库,存储库是存放服务器产生的备份副本的容器。		
	現在购买 使用已有 暂不购买 ②		
高级洗项	现在配置		
	V0 1700		
137 <b>= 1</b>	▲ 配置费用 ¥U.4/90/小时 十 弾性公网P流量费用 ¥U.80/GB		L 15 15 15 15 16 10 1 10 10 10 10 10 10 10 10 10 10 10 1

步骤9 单击"下一步:确认配置"。



步骤 10 进入弹性云服务器的确认配置页面,核对信息无误后,勾选下方的"我已经阅读并同意《华为镜像免责声明》",单击"立即购买",完成用于部署 Porting Advisor 源码移植工具的 x86 云服务器的购买。

弹性云服务器 < 返回	云服务器列表 自知	定义购买 快速购买 New!				
✔ 基础配置		─────────────────────────────────────		高级配置		4 确认配置
配置	基础配置					
	计费模式 规格	按需计费 通用计算增强型   c6.large.2   2vCP	地域 鏡像	北京四 CentOS 7.6 64bit	可用区 系统盘	可用区1 高I0,40GB
	网络配置 🖉		安全组	Sys-default	弹性公网IP	全动态BGP   计费方式: 按流量计费
	高级配置 🖉	ecs_postgre_scan_test	登录凭证	密码		
购买数量	- 1 +	您还可以创建200台云服务器。申请更多云服9	多器配额清单击申请扩大置	£7.		
配置费用 ¥0.4796/小8 参考价格, 具体扣费请以账单为	时 十 弹性公网IP流量费 p准。 <u>了解计费详情</u>	明 ¥0.80/gB				✓ 我已经阅读并同意《华为镇像免责声明》 上一步 立即购买

# 步骤 11 返回弹性云服务器列表页面,查看刚刚购买的弹性云服务器 ecs\_postgre\_scan\_test,等待 其状态变为"运行中"。

	弹性云服务器 ⑦				✓ 最新动态	🕞 帮助引导	购买弹性云服务器
云服务器控制台	开机 关机 重置密码	更多 ▼	所	有运行状态 🔻 名称	- Q	标签搜索 ≫ ℃	C 🗐 🗮
总党	名称/ID	可用区	状态	规格/镜像	IP地址	计费模式	操作
弹性云服务器	ecs_postgre_scan_test	可用区1	● 运行中	2vCPUs   4GB   c6.large.2	121.36.1.167 (弹性公网) 5	按需计费	远程登录 更多•
专属主机	50502070-0030-4512-80			Centos 7.0 04bit	192.100.1.04 ((21))		

#### 步骤 12 参考步骤 3~11,购买用于部署 PostgreSQL 的鲲鹏云服务器。

基础配置(鲲鹏计算,鲲鹏通用计算增强型,kc1.2xlarge.2 8vCPUs | 16GB, CentOS 7.6 64bit)如下所示(其他配置同云服务器 ecs\_postgre\_scan\_test):



CPU架构	X86计算 经膨计算 ②	)				
规格	最新系列	vCPUs 全部	▼ 内存	全部	▼ 规格名称	Q
	鲲鹏通用计算增强型 鲲鹏	内存优化型				
	规格名称	vCPUs   内存 💲	CPU \$	基准/最大带宽 ② 💲	内网收发包 ② 💲	规格参考价 ②
	kc1.large.4	2vCPUs   8GB	Huawei Kunpeng 920 2.6GHz	0.8/3 Gbit/s	300,000	¥0.41/小时
	kc1.xlarge.2	4vCPUs   8GB	Huawei Kunpeng 920 2.6GHz	1.5/5 Gbit/s	500,000	¥0.60/小时
	kc1.xlarge.4	4vCPUs   16GB	Huawei Kunpeng 920 2.6GHz	1.5/5 Gbit/s	500,000	¥0.81/小时
	kc1.2xlarge.2	8vCPUs   16GB	Huawei Kunpeng 920 2.6GHz	3/7 Gbit/s	800,000	¥1.20/小时
	kc1.2xlarge.4	8vCPUs   32GB	Huawei Kunpeng 920 2.6GHz	3/7 Gbit/s	800,000	¥1.63/小时
	liat Sylama S	10000UAL0400	Huawei Kunpeng 920	4.E/O Chit/a	1 100 000	V1 00 //kp+
	当前规格 鲲鹏 <b>通用计算增强</b>	型   kc1.2xlarge.2   8vCF	PUs   16GB			
镜像	公共镜像 私有镜	像	市场镜像			
	CentOS 👻 Ce	ntOS 7.6 64bit with ARM(	40GB) 👻	C		

网络配置如下所示 (其他配置同云服务器 ecs\_postgre\_scan\_test):

弹性云服务器	(返回云服务器列表 自定义购买 快速购买 New!		
✓ 基础配置 —	2 网络耐雨	3 高级配置	4 确认配置
网络 扩展网卡	vpc-e236(192.168.0.0/16)     ▼	4) - C 自动分配P地址	▼ 可用私有PP数量249个 ⑦
安全组	sg-f53a (入方向:TCP/3389, 22   出方向: -) × → C 新建会会 入方向:TCP/3389, 22  出方向: -	19 () 19	

## 🛄 <sub>说明:</sub>

上图中的 vpc-e236(192.168.0.0/16)、subnet-e236(192.168.1.0/24)和 sg-f53a (入方 向:TCP/3389, 22 | 出方向: - ),为购买云服务器 ecs\_postgre\_scan\_test 时创建,此处与云 服务器 ecs\_postgre\_scan\_test 使用相同的网络和安全组。

高级配置如下所示 (其他配置同云服务器 ecs\_postgre\_scan\_test):

• 云服务器名称: ecs\_postgre\_test

弹性云服务器 < 返回云	服务器列表 自定义购买	快速购买 New!		
✓ 基础配置		─ ─ 网络配置 ──	3 高级配置	4 确认配置
云服务器名称	ecs_postgre_test 购买多台云服务器时,名称自动按归	□ 允许重名 增加4位数字后缀。例如:输入ecs,从ecs4	0001开始命名;若已有ecs-0010,从ecs-0011开始命名。	

步骤 13 参考步骤 3~11,购买用于制作 PostgreSQL 的 RPM 安装包,并对安装包进行安装测试的 鲲鹏云服务器。



基础配置(鲲鹏计算,鲲鹏通用计算增强型,kc1.large.2 2vCPUs | 4GB, CentOS 7.6 64bit)如下所示(其他配置同云服务器 ecs\_postgre\_scan\_test):

CPU架构	X86计算					
规格	最新系列	vCPUs 全部	▼ 内存	全部	▼ 规格名称	Q
	鲲鹏通用计算增强型 鲲鹏内	內存优化型 ⑦				
	规格名称	vCPUs   内存 💲	CPU \$	基准 / 最大带宽 🕐 💲	内网收发包 ② 💲	规格参考价 ②
	kc1.small.1	1vCPUs   1GB	Huawei Kunpeng 920 2.6GHz	0.5/2 Gbit/s	200,000	¥0.12/小时
	kc1.large.2	2vCPUs   4GB	Huawei Kunpeng 920 2.6GHz	0.8/3 Gbit/s	300,000	¥0.30/小时
	kc1.large.4	2vCPUs   8GB	Huawei Kunpeng 920 2.6GHz	0.8/3 Gbit/s	300,000	¥0.41/小时
	kc1.xlarge.2	4vCPUs   8GB	Huawei Kunpeng 920 2.6GHz	1.5/5 Gbit/s	500,000	¥0.60/小时
	C kc1.xlarge.4	4vCPUs   16GB	Huawei Kunpeng 920 2.6GHz	1.5/5 Gbit/s	500,000	¥0.81/小时
	lie1 2vlarea 2	Sucolie 116CD	Huawei Kunpeng 920	2/7 Chit/e	800.000	x1 20///vet
	当前规格 銀鹏通用计算增强西	실   kc1.large.2   2vCPUs	4GB			
镜像	公共镜像 私有镜像	共享镜像	市场镜像			
	CentOS - Cent	tOS 7.6 64bit with ARM(4	0GB) -	С		

网络配置如下所示 (其他配置同云服务器 ecs\_postgre\_scan\_test):

弹性云服务器	< 返回云服务器列表 自定义购买 快速购买 New!		
✓ 基础配置 —	2 网络融票	3 高级配置	4 确认配置
网络	vpc-e236(192.168.0.0/16)         で         Subnet-e236(192.168.1.0/24           加需创建新创造以私有云、您可能注意制合创建。           1         増加一块例卡         認知可以提加3 块陽卡	り • C 自动分配P地址	▼ 可用私有IP数量249个 ⑦
安全组	sg-f53a (入方向:TCP/3389, 22   出方向: -) × - C 新建安全 入方向:TCP/3389, 22   出方向: -	題 ⑦	

## 🛄 说明:

上图中的 vpc-e236(192.168.0.0/16)、subnet-e236(192.168.1.0/24)和 sg-f53a (入方向:TCP/3389, 22 | 出方向: - ),为购买云服务器 ecs\_postgre\_scan\_test 时创建,此处与云服务器 ecs\_postgre\_scan\_test 使用相同的网络和安全组。

高级配置如下所示 (其他配置同云服务器 ecs\_postgre\_scan\_test):

● 云服务器名称:ecs\_postgre\_rpm\_test

弹性云服务器 < 返	回云服务器列表 自定义购买 快速购买 New!	
✓ 基础配置	● 网络配置	4 确认配置
云服务器名称	ecs_postgre_rpm_test         介许重名           购买多台云圆务器时,名称自动按序增加4位数字后端,例如: 输入ecs,从ecs-0001开始命名;若已有ecs-0010,从ecs-0011开始命名,	



步骤 14 返回弹性云服务器列表页面 ,查看购买的 3 台弹性云服务器 ,等待其状态均变为"运行中"。

		弹性云服务器 ⑦									
云服务器控制台		开机 关机 重置密码	更多 💌	月	有运行状态 • 名	称 • Q	标签搜索 ≫ ℃	C 0 .			
总览		名称/ID	可用区	状态	规格/镜像	IP地址	计费模式	操作			
弹性云服务器	4	ecs_postgre_rpm_test 1fc5e17f-08ee-4672-b74	可用区1	● 运行中	2vCPUs   4GB   kc1.lar CentOS 7.6 64bit with	ge.2 121.36.24.250 (弹性公网). ARM 192.168.1.158 (私有)	按需计费	远程登录 更多,			
<b>禄金属服务器</b>		ecs_postgre_test 05f34234-40fa-41ee-a19	可用区1	😏 运行中	8vCPUs   16GB   kc1.2 CentOS 7.6 64bit with	xlar 121.36.22.80 (弹性公网) 5 ARM 192.168.1.184 (私有)	i 按需计费	远程登录 更多,			
云硬盘 安属分布式存储	ž	ecs_postgre_scan_test 9b582b78-bc58-45f2-ad	可用区1	😒 运行中	2vCPUs   4GB   c6.larg CentOS 7.6 64bit	e.2 121.36.1.167 (弹性公网) 5 192.168.1.54 (私有)	i	远程登录 更多,			

--结束

## 1.5 修改安全组规则

## 1.5.1 操作步骤

步骤1 在弹性云服务器列表页面的左侧导航栏,选择"安全组",进入安全组列表页面。

专属主机 裸金属服务器		弹性	云服务器 ⑦					✓ 最新动态	▶ 帮助引导	Ħ	购买弹性z	服务器
云硬盘	~											
专属分布式存储	~	开机	1 关机 重置密码	更多 💌		所有运行状态 💌	名称	• Q	标签搜索 ≫	С	Ľ	0
镜像服务			名称/ID	可用区	状态	规格/镜像		IP地址	计费模式		操作	
弹性伸缩	~		ecs_postgre_rpm_test 1fc5e17f-08ee-4672-b74	可用区1	<ul> <li>运行中</li> </ul>	2vCPUs   4GB   k CentOS 7.6 64bit	c1.large.2 with ARM	121.36.24.250 (弹性公网) 192.168.1.158 (私有)	按需计费		远程登录	更多。
密钥对云服务器组	1		ecs_postgre_test 05f34234-40fa-41ee-a19	可用区1	3 运行中	8vCPUs   16GB   CentOS 7.6 64bit	kc1.2xlar with ARM	121.36.22.80 (弹性公网) 5 192.168.1.184 (私有)	按需计费		远程登录	更多。
云备份	1		ecs_postgre_scan_test 9b582b78-bc58-45f2-ad	可用区1	👌 运行中	2vCPUs   4GB   c CentOS 7.6 64bit	6.large.2	121.36.1.167 (弹性公网) 5 192.168.1.54 (私有)	按需计费		远程登录	更多。
云服务器备份	1											
云硬盘备份	1											
弹性负载均衡	1											
弹性公网IP	1											
安全组	0											

步骤 2 在弹性云服务器所关联的安全组所在行,单击"配置规则",进入安全组规则列表页面。

$\bigcirc$		13	安全组 ②						F 使用指南 🛛 🕂	创建安全	è组
网络控制台							名称	•		Q	C
总览			名称	安全组规则	关联实例	描述			操作		
虚拟私有云			Sys-default	4	0	defau	lt		配置规则 关联实例	克隆	
访问控制	^		sg-f53a	4	3				配置规则 关联实例	更多▼	
<ul> <li>安全组</li> </ul>											

步骤 3 在如下安全组规则所在行,单击"修改"。



基本信息 入方向规则 出方向规则	关联实例				
添加规则快速添加规则删除	一键放通	入方向规则: 3 教我设置			С
协议端口 🍸 💿	类型	源地址 ⑦	描述	操作	
全部	IPv4	sg-f53a 🕐		修改复制删除	
TCP : 22	IPv4	0.0.0.0/0 ③	Permit default Linux SSH port.	修改复制删除	
TCP : 3389	IPv4	0.0.0.0/0 ②	Permit default Windows remote de	修改复制删除	

## 步骤 4 将源地址修改为 "IP 地址: 0.0.0.0/0" ,并单击 "确认"。

基本信息 入方向规则 出方向规则 并	关联实例			
添加规则快速添加规则删除	一键放通	入方向规则: 3 教我设置		C
协议端口 🍞 💿	类型	源地址 ⑦	描述	操作
全部放 ▼	IPv4 💌	IP地址     ▼       0 · 0 · 0 · 0 / 0     ⑦		确认 取消

步骤 5 若弹出如下信息提示框,勾选"我已知晓安全风险,暂不开启",并单击"确定"。

提示	×
为保障您的账号及资源的安全,建议您开启敏感操作保护。	
<ul> <li>免费开启</li> <li>在控制台进行敏感操作时,需通过二次认证再次确认您的身份。</li> <li>我已知晓安全风险,暂不开启</li> </ul>	
您可以随时在统一身份认证服务-账号设置开启/关闭操作保护。 下次不再提示	
确定	

#### 步骤 6 返回安全组规则列表,查看源地址是否修改成功。

基本信息 入方向规则 出方向规则	关联实例						
添加规则 快速添加规则 删除	一键放通	入方向规则: 3 教我设置					С
协议端口 🍞 ③	类型	源地址 ②	描述	操作			
全部	IPv4	0.0.0.0/0 ⑦		修改	复制	删除	
TCP : 22	IPv4	0.0.0.0/0 ②	Permit default Linux SSH port.	修改	复制	删除	
TCP : 3389	IPv4	0.0.0.0/0 ③	Permit default Windows remote de	修改	复制	删除	

--结束



# くとして 安装 Porting Advisor 代码移植工具

## 2.1 实验介绍

本实验通过在 x86 云服务器上安装 Porting Advisor 代码移植工具,为后续移植数据库 PostgreSQL 做准备。

## 2.2 前提条件

- x86 云服务器已上电。
- x86 云服务器与 PC 之间网络互通 , 且能访问外网。
- x86 云服务器上已安装 CentOS 7.6 操作系统,且已获取 root 用户帐号和密码。
- PC 上已安装 SSH 远程登录工具 putty 和 WinSCP,下载地址分别为:https://hciakunpeng-application-developer.obs.cn-north-1.myhwclouds.com/putty.exe , https://hcia-kunpeng-application-developer.obs.cn-north-1.myhwclouds.com/WinSCP.zip。
- 已获取华为鲲鹏代码迁移工具安装包 Porting-advisor-x86\_64-linux-1.0.5.tar.gz,下载地址:https://www.huaweicloud.com/kunpeng/software/portingadvisor.html。

# 2.3 安装 Porting Advisor

## 2.3.1 操作步骤

步骤 1 在弹性云服务器列表中,记录 x86 云服务器 ecs\_postgre\_scan\_test 的弹性公网 IP 地址, 如下图中的"121.36.1.167"。



	~ 一 一 一 一 一 一 一 一 一 一 一 一 一 一 一 一 一 一 一		云服务器 ⑦						✓最新动态	▶ 帮助引导	Ŕ	购买弹性法	、服务器
云服务器控制台	з С	开机	关机	重置密码	更多 ▼		所有运行状态 🔻	名称	• Q	标签搜索 ≫	C	Ľ	0
总览	- 1		名称/ID		可用区	状态	规格/镜像		IP地址	计费模式		操作	
弹性云服务器	_		ecs_postgre_rp	m_test	可用区1	😏 运行中	2vCPUs   4GB   k	c1.large.2	121.36.24.250 (砷性公网)	按需计费		远程登录	更多
专属主机			Truber	4072-074			Centos 7.0 04bi	WIT ANW	132.100.1.130 (HAH)				
裸金属服务器			ecs_postgre_te 05f34234-40fa	est -41ee-a19	可用区1	运行中	e 8vCPUs   16GB   CentOS 7.6 64bit	kc1.2xlar with ARM	121.36.22.80 (弹性公网) 5 192.168.1.184 (私有)	按需计费		远程登录	更多
云硬盘	~		ecs_postgre_so 9b582b78-bc58	can_test 3-45f2-ad	可用区1	〇 运行中	2vCPUs   4GB   c CentOS 7.6 64bit	6.large.2	121.36.1.167 (弹性公网) 5 192.168.1.54 (私有)	按需计费		远程登录	更多

步骤 2 打开 PC 上的 WinSCP, 输入 x86 云服务器 ecs\_postgre\_scan\_test 的弹性公网 IP 地址, 用户名 "root"和密码, 单击下方的 "Login"。

🔩 WinSCP Login	– 🗆 X
Vew Site	Session File protocol: SFTP Host name: 121.36.1.167 User name: Password: root Save Advanced
Tools   Manage	▼ Login ▼ Close Help

步骤 3 首次访问会弹出如下信息提示框,单击 "Yes"。



步骤 4 进入如下页面,在左侧选择华为鲲鹏代码迁移工具安装包 Porting-advisor-x86\_64-linux-1.0.5.tar.gz 所在路径,并将安装包拖至右侧 x86 云服务器 ecs\_postgre\_scan\_test 的 /root 目录下。

若弹出如下信息提示框,单击"OK",等待上传完成。



Upload			?	×			
Upload file '	Porting-advisor-x86_64-lin	ux-1.0.6.tar.gz' to remote dire	ectory:				
/rooy				· · · ·			
Transfer settings Transfer type: Bina	ıry						
Transfer on bac	kground (add to transfer q	iene)					
Transfer setting	gs 💌	OK Ca	ncel He	elp			
Do not show thi	s dialog box again						
D:\ - root@121.36.1.1	167 - WinSCP						
Local Mark Files Comm	nands Session Options Remote	e Help					
🖽 🚟 🗟 Synchronize 🔳 🖬	🦻 📰 🎯 💷 🖗 Queue 🔹 Transf	er Settings Default 🔹 🖉 •					
📮 root@121.36.1.167 🧉	New Session						
🥪 D: Data	• 🖀 🔯 📾 📾 🏠 🍘 🗣 • • • •		📜 root	• 🖀 🔽 🗈	😰 🍙 🌌 🐐 Find Files 🐁 🗇 -	40 ×	
🗊 Upload 🗊 🕼 Edit 🗙 🖬	🖞 🖳 Properties 😫 🖻 🖅 🗉 🕅		Download 🚇	🛛 Edit 🗙 🛃 🖏	Properties 😂 🖻 🗏 🖻		
D:\			/root				
Name Ext	Size Type	Changed	Name Ext		Size Changed	Rights	Owner
\$RECYCLE.BIN	文件夹	2018/10/30 18:55:45			2019/10/14 15:17:03	г-хг-хг-х	root
D-Box	文件夹	21% Uploading		? ×	2019/2/27 15:16:51	rwx	root
Email	文件夹			Connel	2019/2/27 15:17:03	rwxr-xr-x	root
HedExDownload	文件夹	0-7		Cancer	2019/2/27 15:12:13	rwxr	root
HedExInstall	文件夹	K.	2	Minimize	2019/10/14 15:17:03	rwx	root
LenovoDrivers	文件夹	Files Dulbertine educes of (	d Revert 0.6 her en		0 B 2019/9/29 21:59:14	rw-rr	root
<b>WRECYCLER</b>	文件夹	Target: /root/	94-11nux-1.0.0.tar.gz	Once finished:	8 B 2013/12/29 10:26:31	rw-rr	root
System Volume Infor	文件夹	ingst proof		Stav idle V	6 B 2013/12/29 10:26:31	rw-rr	root
temp	文件夹				6 B 2013/12/29 10:26:31	ΓW-ΓΓ	root
WORK	文件夹	Time left: 0:00:20 Tir	ne elapsed: 0:00:05	Course ( Arms)	0 B 2013/12/29 10:26:31	ΓW-ΓΓ	root
Porting-advisor-x86	91,065 KiB WinRAR archive	bytes transferred: 19,127 KiB Sp	eeu: 3,51/ KIB/s	≥peed (KIB/s):	0 B 2019/2/27 15:17:47	rw	root
				Unimited	9 B 2013/12/29 10:26:31	rw-rr	root

步骤 5 打开 PC 上的 putty, 输入 x86 云服务器 ecs\_postgre\_scan\_test 的弹性公网 IP 地址,单 击下方的"Open"。

Category:		
<ul> <li>Session</li> <li>Logging</li> <li>Terminal</li> <li>Keyboard</li> <li>Bell</li> <li>Features</li> <li>Window</li> <li>Appearance</li> <li>Behaviour</li> <li>Translation</li> <li>Selection</li> <li>Colours</li> <li>Connection</li> <li>Data</li> <li>Proxy</li> <li>Telnet</li> <li>Rlogin</li> <li>SSH</li> <li>Serial</li> </ul>	Basic options for your PuTTY ses Specify the destination you want to connect Host Name (or IP address) 121.36.1.167 Connection type: Raw Telnet Rlogin SSH Load, save or delete a stored session Saved Sessions	sion to Port 22 Serial
	Default Settings	Load Save Delete
About	Close window on exit: Always Never Only on cle	ean exit

步骤 6 首次登录会弹出如下信息提示框,单击"是(Y)"。



PuTTY See	curity Alert			×
	The server's have no guar think it is. The server's ssh-rsa 2048 If you trust th PuTTY's cach If you want tr adding the k If you do not connection.	host key is not c antee that the s rsa2 key fingerp 2d:33:cd:1c:96:5 nis host, hit Yes e and carry on c o carry on conne ey to the cache, trust this host, l	ached in the regis erver is the comp rint is: i9:3f:0c:d8:93:22:1 to add the key to onnecting. ecting just once, w hit No. hit Cancel to aban	stry. You uter you 3:d1:0c:91:13 vithout don the
		是(Y)	否(N)	取消

步骤 7 输入 x86 云服务器 ecs\_postgre\_scan\_test 的用户名 "root"和密码,进入如下命令行界面,表示登录成功。

🖻 root@ecs-postergre-scan-test:~	—	$\times$
login as: root root@121.36.1.167's password: Last login: Tue Oct 15 20:24:34 2019 from 119.3.119.14		~
Welcome to Huawei Cloud Service		
[root@ecs-postergre-scan-test ~]#		

步骤 8 执行以下命令,查看是否存在华为鲲鹏代码迁移工具安装包 Porting-advisor-x86\_64-linux-1.0.5.tar.gz,若存在则直接解压缩,否则重复步骤 2~4 重新上传。

```
tar -zxvf Porting-advisor-beta-x86 64-linux-1.0.5.tar.gz
```

[root@ecs-postgre-scan-test ~]# ls Porting-advisor-beta-x86_64-linux-1.0.5.tar.gz
[root@ccs-postgre-scan-test ~] # tar -zxvf Porting-advisor-beta-x86 64-linux-1.0.
5.tar.gz
Porting-advisor-x86 64-linux-1.0.5/
Porting-advisor-x86_64-linux-1.0.5/install.sh
Porting-advisor-x86_64-linux-1.0.5/Porting-advisor-x86_64-linux-webui-1.0.5.tar.
gz
Porting-advisor-x86_64-linux-1.0.5/Porting-advisor-x86_64-linux-cmd-1.0.5.tar.gz
Porting-advisor-x86_64-linux-1.0.5/Porting-advisor-x86_64-linux-portal-1.0.5.tar
.gz

步骤 9 执行以下命令,进入解压后的华为鲲鹏代码迁移工具安装包目录,安装其 Web 模式。

```
ls
cd Porting-advisor-bete-x86_64-linux-1.0.5
sh install.sh web
```

- 配置 Web Server 的 IP 地址:按 Enter 键,默认为操作系统上所有 IP 地址。
- 配置 https 端口:按 Enter 键,默认为 8084。

ls





步骤 10 等待 3~5 分钟完成安装,直到出现如下回显信息,表示安装成功。

Running mig	grations:
Applying	contenttypes.0001_initial oĸ
Applying	admin.0001_initial OK
Applying	admin.0002_logentry_remove_auto_add OK
Applying	admin.0003_logentry_add_action_flag_choices OK
Applying	contenttypes.0002_remove_content_type_name ok
Applying	auth.0001_initial OK
Applying	auth.0002_alter_permission_name_max_length or
Applying	auth.0003_alter_user_email_max_length OK
Applying	auth.0004_alter_user_username_opts oĸ
Applying	auth.0005_alter_user_last_login_null oĸ
Applying	auth.0006_require_contenttypes_0002 ox
Applying	<pre>auth.0007_alter_validators_add_error_messages OK</pre>
Applying	auth.0008_alter_user_username_max_length OK
Applying	auth.0009_alter_user_last_name_max_length OK
Applying	auth.0010_alter_group_name_max_length OK
Applying	auth.0011_update_proxy_permissions OK
Applying	operationlog.0001_initial OK
Applying	sessions.0001_initial OK
Applying	taskmanager.0001_initial oĸ
Successful	ly installed porting advisor in /opt/portadv/tools

--结束



# 3 Porting Advisor 移植 PostgreSQL

## 3.1 实验介绍

本实验以 PostgreSQL 的源码为例,使用 Porting Advisor 代码移植工具对 PostgreSQL 进行移植分析,并根据分析报告和修改建议,修改源码,实现代码的移植。

## 3.2 前提条件

- x86 云服务器和鲲鹏云服务器已上电。
- x86 云服务器、鲲鹏云服务器与 PC 之间网络互通 , 且能访问外网。
- 云服务器上已安装 CentOS 7.6 操作系统,且已获取 root 用户帐号和密码。
- x86 云服务器上已安装 Porting Advisor 工具。
- PC 上已安装 SSH 远程登录工具 putty 和 WinSCP。

# 3.3 分析 PostgreSQL 源码

## 3.3.1 操作步骤

步骤1 使用 PC 上的浏览器访问:https://www.postgresql.org/ftp/source/v11.3/,下载 PostgreSQL 11.3 版本源代码包 postgresql-11.3.tar.gz。



Quick	File Browser 🖿		
LINKS	Top $\rightarrow$ source $\rightarrow$ v11.3		
<ul> <li>Downloads</li> <li>Binary</li> <li>Source</li> </ul>	Directories		
<ul> <li>Software Catalogue</li> </ul>	Parent Directory]		
• File Browser	Files		
	postgresql-11.3.tar.bz2	2019-05-06 21:00:06	18.8 MB
	🗋 postgresql-11.3.tar.bz2.md5	2019-05-06 21:00:06	58 bytes
	postgresql-11.3.tar.bz2.sha256	2019-05-06 21:00:06	90 bytes
	postgresql-11.3.tar.gz	2019-05-06 21:00:06	24.7 MB
	postgresql-11.3.tar.gz.md5	2019-05-06 21:00:06	57 bytes
	🗋 postgresql-11.3.tar.gz.sha256	2019-05-06 21:00:06	89 bytes

## 步骤 2 参考 2.3.1 中的步骤 2~4, 打开 WinSCP, 将下载的 PostgreSQL 源代码包 postgresql-11.3.tar.gz 上传至 x86 云服务器 ecs\_postgre\_scan\_test 的/opt/portadv/portadmin/ 目录下。

🖺 D:\ - root@121.36.1.10	67 - WinSCP							×
Local Mark Files Comma	ands Session Options Remote	e Help						
🎛 🗃 💐 Synchronize 🔳 🖉	🛛 💽 🗶 🗊 Queue · 🛛 Transfe	er Settings Default 🔹 🔗 •						
📮 root@121.36.1.167 🧉	New Session							
D: Data	• 🔊 🖸 🖿 🕜 🎜 💊 📥 • • • •		portadmin	- 🖅 💽 😥 🕫	🏫 🖉 🍓 Find Files 🗞 📥 🔹	ap a		
Dpload 🗊 🔐 Edit 🗙 🏑	🗣 Properties 🖆 🕞 🖅 🖂 🕅		Download 📾	a Edit 🗙 🛃 🗟 Pro	operties 🖆 🕞 🗐 🖂 🕅			
D:\			/opt/portadv/por	tadmin				
Name Ext \$RECYCLE.BIN	Size Type 文件夹	Changed 2018/10/30_18:55:45	Name Ext	Si	ize Changed 2019/10/16 15:43:53	Rights rwxr-xr-x	Owner root	
D-Box Email HedExDownload HedExInstall LenovoDrivers RECYCLER System Volume Infor temp	文件夹 文件夹 文件夹 文件夹 文件夹 文件夹 文件夹 文件夹 文件夹	File: D:/postgresql=11.3.tar.gz Target: /opt/portadv/portadmin/	Þ	Cancel Minimize Once finished: Stay idle				
WORK Porting-advisor-x86 postgresql-11.3.tar.gz	文件夹 91,065 KiB WinRAR archive 25,262 KiB WinRAR archive	Bytes transferred: 14,044 KiB Spe	eed: 3,594 KiB/s	Speed (KiB/s): Unlimited ~				

步骤 3 返回 putty ,登录 x86 云服务器 ecs\_postgre\_scan\_test ,执行以下命令 ,进入 PostgreSQL 源代码包 postgresql-11.3.tar.gz 所在路径 ,查看是否存在源代码包 ,若存在则直接解压 缩 , 否则需要重新上传。

cd /opt/portadv/portadmin
ls
tar -zxvf postgresql-11.3.tar.gz
[root@ecs-postergre-scan-test ~]# cd /opt/portadv/portadmin/ [root@ecs-postergre-scan-test portadmin]# ls postgresql-11.3.tar.gz
<pre>[root@ecs-postergre-scan-test portadmin]# tar -zxvf postgresql-11.3.tar.gz</pre>

步骤 4 使用 PC 上的浏览器访问:https://x86 云服务器的弹性公网 IP 地址:8084,如 https://121.36.1.167:8084,进入华为鲲鹏代码移植工具的 Web 页面。



● 学う組織代码迁移工具	双距照近 <b>华为鲲鹏代码迁移工具</b> V1.0.5 加产名 密码	<ul> <li>圖傳中文 •</li> </ul>

🛄 说明:

如果弹出"此网站的安全证书有问题"的告警提示,请选择"继续浏览此网站(不推荐)",进入登录界面。

步骤 5 输入用户名和密码,单击"登录",进入登录界面。

🛄 <sub>说明:</sub>

系统的默认用户名为 portadmin, 默认密码为 Admin@9000。

首次登录 Web 的用户,系统提示修改默认密码。请按提示修改密码,密码需要满足如下复 杂度要求:

- 密码长度为6~32个字符。
- 必须包含大写字母、小写字母、数字、特殊字符(`~!@#\$%^&\*()-\_=+\|[{}];:'",<.>/?)中
   的至少两种及以上的字符。

修改初始密码	<b>马</b>		Х
*旧密码			
<b>★</b> 密码			
*确认密码			
	确认	取消	

按要求修改初始密码后单击"确认",或者直接单击"取消",后续再自行修改密码。



#### 步骤 6 配置源代码存放路径,其他参数保持默认值。

□ 说明:

这里的源代码存放路径参数填写 "postgresql-11.3/",其中

"/opt/portadv/portadmin/"为固定路径,源码代码全路径为

"/opt/portadv/portadmin/postgresql-11.3/" 。

界面上的编译器版本 GCC 4.8 与后台 CentOS 上安装的编译器 GCC 版本相互独立,两者没有关联关系。

源代码存放路径	/opt/portadv/portadmin/	
	postgresql-11.3/	
	检查项目:编译器选项、编译器宏、汇编程序、内置函数、属性	
编译器版本	GCC 4.8	
构建工具	make 💌	
*编译命令	make	
目标操作系统	CentOS 7.6	
目标系统内核版本	v4.14	]
	分析	

#### 步骤 7 单击"分析",生成分析报告。

弹窗页面显示任务分析进度,分析完成后,自动跳转至"移植报告"界面。

🛄 <sub>说明:</sub>

在分析进行中,请勿关闭当前页面,任务分析完成后可直接跳转至"移植报告"界面,也可以在历史报告的列表中,单击指定分析任务的报告日期。





# 步骤 8 待分析完成后,选择左上角的"移植报告"和"移植建议"页签,查看移植报告和移植建议。

<	2019/1	10/17	10:06:35								
移植	报告 利	随建议									
						2019/10/17 10:	06:35				
	源代码存放 编译命令	踏径		/opt/portadv/portadmin/postgresql-11. make	.3/	编译器版本 目标場作系统	GCC 4.8 CentOS 7.6	构建工	工具	ma v4.	ake
		E	依赖库SO文件	总数	ģ: 0,	需要移植: 0					-
		<b>—</b>	需要移植的源文件	0							
		 (7)	需要移植的代码行数	α ς/c	(+ +和)	Makefile源代码:0行; 汇编代码:0行					
						下载报告 (.csv) 下载	或报告 (.html)				



<b>&lt;</b> 1	2019/10/17 10:06:	35		
移植报告	移植建议			
			您的原始源代码将显示在该页面的左侧区域	成,建议修改的源代码将显示在右侧区
			域。右侧区域的内容仅为修改建	议,不会影响原始源代码。
			□ 显示	原代码
			确认	×

从移植报告中可以看出,需要移植的依赖库 SO 文件、源文件、代码行数均为0,说明 postgresql-11.3 源码不需要修改,可以直接进行源码编译安装。

--结束

# 3.4 升级 GCC 编译器版本

## 3.4.1 操作步骤

步骤 1 在弹性云服务器列表中,记录鲲鹏云服务器 ecs\_postgre\_test 的弹性公网 IP 地址,如下 图中的 "121.36.22.80"。

云服务器控制台	1机 关机	<b>田宮</b> 慶初								
			更多 🔻	Ē	所有运行状态 🔹	名称	* Q	标签搜索 ≫ ℃	C d	J #
总览	名称/ID		可用区	状态	规格/镜像		IP地址	计费模式	操作	
弹性云服务器	ecs_postgre_	rpm_test e-4672-b74	可用区1	🕤 运行中	2vCPUs   4GB   k CentOS 7 6 64bit	c1.large.2	121.36.24.250 (弹性公网)	按需计费	远程登录	更多。
专属主机										
裸金属服务器	05f34234-40f	a-41ee-a19	可用区1	😏 运行中	CentOS 7.6 64bit	with ARM	192.168.1.184 (私有)	按需计费	远程登录	更多。
云硬盘	ecs_postgre_ 9b582b78-bc	scan_test 58-45f2-ad	可用区1	🕤 运行中	2vCPUs   4GB   c CentOS 7.6 64bit	6.large.2	121.36.1.167 (弹性公网) 5 192.168.1.54 (私有)	按需计费	远程登录	更多。

步骤 2 参考 2.3.1 中的步骤 5~7, 打开 putty, 登录鲲鹏云服务器 ecs\_postgre\_test。





#### 步骤 3 执行以下命令,配置 yum 源,并安装 PostgreSQL 依赖包。

yum -y install gcc gcc-c++ automake zlib zlib-devel bzip2 bzip2-devel bzip2libs readline readline-devel bison ncurses ncurses-devel libaio-devel openssl openssl-devel gmp gmp-devel mpfr mpfr-devel libmpc libmpc-devel

-			-	-	
[root@ecs-pos zip2 bzip2-de libaio-devel evel	stgre-test ~]# yw evel bzip2-libs n openssl openssl-	um -y install ceadline read devel gmp gmp	gcc gcc-c++ aut line-devel bison p-devel mpfr mp:	tomake zlib n ncurses nc fr-devel lib	zlib-devel k urses-devel mpc libmpc-c
Loaded plugin Determining f	ns: fastestmirron Tastest mirrors	-			
epel/aarch64/ * base: mirr * epel: mirr * extras: mi * updates: m	metalink cors.huaweicloud cors.yun-idc.com rrors.huaweiclou dirror-hk.koddos	.com nd.com .net		7.2 kB	00:00
		<u></u>			



Updated: gcc.aarch64 0:4.8.5-39.el7 openssl.aarch64 1:1.0.2k-19.el7	gcc-c++.aarch64 0:4.8.5-39.el7 readline.aarch64 0:6.2-11.el7
Dependency Updated: cpp.aarch64 0:4.8.5-39.el7 e2fsprogs-libs.aarch64 0:1.42.9-16.el7 libcom_err.aarch64 0:1.42.9-16.el7 libgomp.aarch64 0:4.8.5-39.el7 libstdc++.aarch64 0:4.8.5-39.el7 openssl-libs.aarch64 1:1.0.2k-19.el7	e2fsprogs.aarch64 0:1.42.9-16.el7 krb5-libs.aarch64 0:1.15.1-37.el7_7.2 libgcc.aarch64 0:4.8.5-39.el7 libss.aarch64 0:1.42.9-16.el7 libstdc++-devel.aarch64 0:4.8.5-39.el7

#### 步骤 4 执行以下命令,检查当前环境中的 GCC 编译器是否符合版本要求。

gcc -v
<pre>[root@ecs-postergre-test ~]# gcc -v Using built-in specs. COLLECT_GCC=gcc COLLECT_LTO_WRAPPER=/usr/libexec/gcc/aarch64-redhat-linux/4.8.5/lto-wrapper Target: aarch64-redhat-linux Configured with:/configureprefix=/usrmandir=/usr/share/maninfodir=/usr hare/infowith-bugurl=http://bugzilla.redhat.com/bugzillaenable-bootstrape ble-sharedenable-threads=posixenable-checking=releasewith-system-zlibe blecxa_atexitdisable-libunwind-exceptionsenable-gnu-unique-objectenabl linker-build-idwith-linker-hash-style=gnuenable-languages=c,c++,objc,obj-c++ ava,fortran,ada,ltoenable-pluginenable-initfini-arraydisable-libgcjwit isl=/builddir/build/BUILD/gcc-4.8.5-20150702/obj-aarch64-redhat-linux/isl-install with-cloog=/builddir/build/BUILD/gcc-4.8.5-20150702/obj-aarch64-redhat-linux/cloog nstallenable-gnu-indirect-functionbuild=aarch64-redhat-linux</pre>
gcc version 4.8.5 20150623 (Red Hat 4.8.5-39) (GCC)

- 若版本为 5.3 及以上,则表示符合版本要求,结束。
- 否则,请执行步骤5,升级GCC版本(本实验以升级到7.3.0版本为例进行介绍)。

#### 步骤 5 执行以下命令,安装 wget 工具。

yum -y install wget

[ro	ot@ecs-postgre-test ~]# yum -y install wget
Loa	ded plugins: fastestmirror
Loa	ding mirror speeds from cached hostfile
*	base: mirrors.huaweicloud.com
*	epel: mirrors.yun-idc.com
*	extras: mirrors.huaweicloud.com
*	updates: mirror-hk.koddos.net



#### 步骤 6 执行以下命令,下载 7.3.0 版本的 GCC 软件包。

wget https://mirrors.tuna.tsinghua.edu.cn/gnu/gcc/gcc-7.3.0/gcc-7.3.0.tar.gz



#### 步骤 7 执行以下命令, 对软件包进行解压缩。

ls
tar -xvf gcc-7.3.0.tar.gz
[root@ecs-postergre-test ~]# ls
gcc-7.3.0.tar.gz
[root@ecs-postergre-test ~]# tar -xvf gcc-7.3.0.tar.gz

#### 步骤 8 执行以下命令,编译安装 GCC。

```
ls
```

```
cd gcc-7.3.0
```

```
./configure --prefix=/usr --mandir=/usr/share/man --infodir=/usr/share/info --
enable-bootstrap --enable-shared --enable-threads=posix --enable-
checking=release --with-system-zlib --enable-___cxa_atexit --disable-libunwind-
exceptions --enable-gnu-unique-object --enable-linker-build-id --with-linker-
hash-style=gnu --enable-languages=c,c++,objc,obj-c++,fortran,lto --enable-
plugin --enable-initfini-array --disable-libgcj
make -j 96
```

make install

此过程需要等待较长时间,直到出现如下回显信息,表示安装完成。

See any operating	system documentation about shared libraries for
more information,	such as the ld(1) and ld.so(8) manual pages.
make[4]: Nothing t	o be done for `install-data-am'.
make[4]: Leaving d	irectory `/root/gcc-7.3.0/aarch64-unknown-linux-gnu/libatomic'
make[3]: Leaving d	<pre>irectory `/root/gcc-7.3.0/aarch64-unknown-linux-gnu/libatomic'</pre>
make[2]: Leaving d	irectory `/root/gcc-7.3.0/aarch64-unknown-linux-gnu/libatomic'
make[1]: Leaving d	irectory `/root/gcc-7.3.0'

#### 步骤 9 执行以下命令,再次查看 GCC 的版本,验证是否升级成功。

gcc -v



pot@ecs-postergre-test gcc-7.3.0]# gcc -v
ing built-in specs.
LLECT_GCC=gcc
LLECT_LTO_WRAPPER=/usr/libexec/gcc/aarch64-unknown-linux-gnu/7.3.0/lto-wrapper
rget: aarch64-unknown-linux-gnu
nfigured with: ./configureprefix=/usrmandir=/usr/share/maninfodir=/usr/sh
e/infoenable-bootstrapenable-sharedenable-threads=posixenable-checking
eleasewith-system-zlibenable cxa atexitdisable-libunwind-exceptionse
ple-gnu-unique-objectenable-linker-build-idwith-linker-hash-style=gnuenab
-languages=c,c++,objc,obj-c++,fortran,ltoenable-pluginenable-initfini-array
lisable-libgcj
read model: posix
c version 7.3.0 (GCC)

--结束

# 3.5 移植 PostgreSQL

## 3.5.1 操作步骤

步骤 1 返回 putty,登录鲲鹏云服务器 ecs\_postgre\_test,执行以下命令,将 x86 云服务器 ecs\_postgre\_scan\_test 上的 PostgreSQL 源码包拷贝至 "/home" 目录下。

scp root@121.36.1.167:/opt/portadv/portadmin/postgresql-11.3.tar.gz /home/

# 此处 IP 地址请参照自己的实验环境 ecs\_postgre\_scan\_test 的弹性公网 IP。

首次连接会出现如下提示,输入"yes",按 Enter。

按提示输入 x86 云服务器 ecs\_postgre\_scan\_test 的 root 用户的密码。

等待拷贝完成。

[root@ecs-postergre-test ~]# scp root@121.36.1.167:	/opt/por	tadv/p	ortadmin/p	ostgresql-
11.3.tar.gz /home/				
The authenticity of host '121.36.1.167 (121.36.1.16	57)' can'	t be e	stablished	l.
ECDSA key fingerprint is SHA256:x/T12QhE2eDq+J161I4	12QaE/ltq	[vFzJ8W	10Y8DmoN1k	ς.
ECDSA key fingerprint is MD5:74:45:77:7f:8f:a5:0f:a	al:bb:2d:	2e:b8:	2b:4a:a7:9	b.
Are you sure you want to continue connecting (yes/r	ıo)? yes			
Warning: Permanently added '1 <u>21.36.1.16</u> 7' (ECDSA) t	to the li	st of	known host	s.
root@121.36.1.167's password:				
postgresql-11.3.tar.gz	100% 2	5MB	6.5MB/s	00:03

步骤 2 执行以下命令,进入/home 目录,查看是否存在 PostgreSQL 源码包,若存在则直接解压缩,否则重复步骤1重新拷贝。

```
cd /home/
ls
tar -zxvf postgresql-11.3.tar.gz
[root@ecs-postergre-test ~]# cd /home/
[root@ecs-postergre-test home]# ls
postgresql-11.3.tar.gz
[root@ecs-postergre-test home]# tar -zxvf postgresql-11.3.tar.gz
```

步骤 3 执行以下命令,进入解压后的源码目录,创建安装目录/home/pgsql。

cd postgresql-11.3



mkdir /home/pgsql

[root@ecs-postergre-test home]# cd postgresql-11.3
[root@ecs-postergre-test postgresql-11.3]# mkdir /home/pgsql

#### 步骤 4 执行以下命令 , 生成 Makefile 文件。

./configure -prefix=/home/pgsql

```
直到出现如下回显信息,表示生成成功。
```

config.status:	creating GNUmakefile
config.status:	creating src/Makefile.global
config.status:	creating src/include/pg_config.h
config.status:	creating src/include/pg_config_ext.h
config.status:	creating src/interfaces/ecpg/include/ecpg_config.h
config.status:	linking src/backend/port/tas/dummy.s to src/backend/port/tas.s
config.status:	linking src/backend/port/dynloader/linux.c to src/backend/port/dy
nloader.c	
config.status:	linking src/backend/port/posix_sema.c to src/backend/port/pg_sema
. C	
config.status:	linking src/backend/port/sysv_shmem.c to src/backend/port/pg_shme
m.c	
config.status:	linking src/backend/port/dynloader/linux.h to src/include/dynload
er.h	
config.status:	linking src/include/port/linux.h to src/include/pg_config_os.h
config.status:	linking src/makefiles/Makefile.linux to src/Makefile.port

#### 步骤 5 执行以下命令,编译并安装 PostgreSQL。

```
make -j 64
make install
```

直到出现如下回显信息,表示安装完成。

```
make[2]: Entering directory `/home/postgresql-11.3/src/test/perl'
make[2]: Nothing to be done for `install'.
make[2]: Leaving directory `/home/postgresql-11.3/src/test/perl'
/usr/bin/mkdir -p '/home/pgsql/lib/pgxs/src'
/usr/bin/install -c -m 644 Makefile.global '/home/pgsql/lib/pgxs/src/Makefile.gl
obal'
/usr/bin/install -c -m 644 Makefile.port '/home/pgsql/lib/pgxs/src/Makefile.port
'/usr/bin/install -c -m 644 ./Makefile.shlib '/home/pgsql/lib/pgxs/src/Makefile.s
hlib'
/usr/bin/install -c -m 644 ./Makefile.shlib '/home/pgsql/lib/pgxs/src/Makefile.s
hlib'
/usr/bin/install -c -m 644 ./nls-global.mk '/home/pgsql/lib/pgxs/src/nls-global.
mk'
make[1]: Leaving directory `/home/postgresql-11.3/src'
make -C config install
make[1]: Entering directory `/home/postgresql-11.3/config'
/usr/bin/install -c -m 755 ./install-sh '/home/pgsql/lib/pgxs/config/install-sh'
/usr/bin/install -c -m 755 ./missing '/home/pgsql/lib/pgxs/config/missing'
make[1]: Leaving directory `/home/postgresql-11.3/config'
PostgreSQL installation complete.
```

#### 步骤 6 执行以下命令 , 创建 postgres 用户和用户组。

```
/usr/sbin/groupadd -g 1001 postgres
/usr/sbin/useradd -u 1012 -m -g postgres postgres
[root@ecs-postergre-test postgresql-11.3]# /usr/sbin/groupadd -g 1001 postgres
[root@ecs-postergre-test postgresql-11.3]# /usr/sbin/useradd -u 1012 -m -g postg
res postgres
```

步骤 7 执行以下命令,设置 postgres 用户密码。



passwd postgres

按提示输入 postgres 用户的密码,并再次确认。



#### 步骤 8 执行以下命令, 切换到 postgres 用户, 并初始化数据库。

```
su - postgres
```

/home/pgsql/bin/initdb -D pgsql/

```
[root@ecs-postergre-test postgresql-11.3]# su - postgres
[postgres@ecs-postergre-test ~]$ /home/pgsql/bin/initdb -D pgsql/
The files belonging to this database system will be owned by user "postgres".
This user must also own the server process.
The database cluster will be initialized with locale "en_US.UTF-8".
The default database encoding has accordingly been set to "UTF8
The default text search configuration will be set to "english".
Data page checksums are disabled.
creating directory pgsql ... ok
creating subdirectories ... ok
selecting default max_connections ... 100
selecting default shared_buffers ... 128MB
selecting dynamic shared memory implementation ... posix
creating configuration files ... ok
running bootstrap script ... ok
performing post-bootstrap initialization ... ok
syncing data to disk ... ok
WARNING: enabling "trust" authentication for local connections
You can change this by editing pg_hba.conf or using the option -A, or --auth-local and --auth-host, the next time you run initdb.
Success. You can now start the database server using:
     /home/pgsql/bin/pg_ctl -D pgsql/ -l logfile start
```

#### 步骤 9 执行以下命令, 启动 PostgreSQL 数据库。

/home/pgsql/bin/pg\_ctl -D pgsql/ -l logfile start

```
[postgres@ecs-postergre-test ~]$ /home/pgsql/bin/pg_ctl -D pgsql/ -l logfile start
waiting for server to start.... done
server started
```

#### 步骤 10 执行以下命令,确认 PostgreSQL 数据库进程是否正常启动。

ps -ef | grep postgres

	1.6 C I	
[postgres@ecs-postgre-test	~]\$ ps -er   grep po	ostgres
root 26852 10617 0 20:	14 pts/0 00:00:00	) su - <b>postgres</b>
postgres 26853 26852 0 20:	14 pts/0 00:00:00	) -bash
postgres 26890 1 0 20:	15 pts/0 00:00:00	) /home/pgsql/bin/ <mark>postgres</mark> -D pgsql
postgres 26892 26890 0 20:	15 ? 00:00:00	) <b>postgres:</b> checkpointer
postgres 26893 26890 0 20:	15 ? 00:00:00	) <b>postgres:</b> background writer
postgres 26894 26890 0 20:	15 ? 00:00:00	) <b>postgres:</b> walwriter
postgres 26895 26890 0 20:	15 ? 00:00:00	) <mark>postgres:</mark> autovacuum launcher
postgres 26896 26890 0 20:	15 ? 00:00:00	) <b>postgres:</b> stats collector
postgres 26897 26890 0 20:	15 ? 00:00:00	) <b>postgres:</b> logical replication launcher
postgres 26898 26853 0 20:	15 pts/0 00:00:00	) ps -ef
postgres 26899 26853 0 20:	15 pts/0 00:00:00	) grepcolor=auto <b>postgres</b>



如上图所示, PostgreSQL相关进程已经正常启动了。

步骤 11 执行以下命令,验证是否可以正常登录 PostgreSQL 数据库。

□ 说明:

初次登录数据库,无需密码。

/home/pgsql/bin/psql -U postgres

<pre>[postgres@ecs-postergre-test ~]\$ /home/pgsql/bin/psql -U postgres psql (11.3)</pre>
Type "help" for help.
postgres=#
如上图所示,可以正常登录 PostgreSQL 数据库。
输入"\q",退出数据库。
postgres=# \q [postgres@ecs-postergre-test ~]\$

- 步骤 12 执行以下命令,编辑配置文件 pg\_hba.conf。
  - cd /home/postgres/pgsql/
  - vi pg\_hba.conf

[postgres@ecs-postergre-test ~]\$ cd /home/postgres/pgsql/ [postgres@ecs-postergre-test pgsql]\$ vi pg\_hba.conf

按"i"进入编辑模式,输入以下内容,按"Esc"退出编辑模式,输入":wq!",保存退出。

# "loca local	l" is for all	r Unix domain socket all		trust
nost	all	all	127.0.0.1/32	trust
host	all	all	■.■.■.■/■ 🔺	trust

<u>
 〇</u> (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1)
 〇 (1

该行内容用以允许非本地访问 PostgreSQL 数据库, IP 地址段表示允许访问 PostgreSQL 数据库的地址段,具体请按实际情况配置。例如,此处配置为 172.168.1.1,表示只允许 172.168.1.1 这一台主机访问。

步骤 13 执行以下命令,编辑配置文件 postgresql.conf。

vi postgresql.conf

[postgres@ecs-postergre-test pgsql]\$ vi postgresql.conf

修改配置连接信息,按"i"进入编辑模式,将"listen\_addresses"后的"localhost"修改为"\*",并删除前面的"#",按"Esc"退出编辑模式,输入":wq!",保存退出。



	_

# CONNECTIONS AND AUTHENTICATION	
#	
# - Connection Settings -	
listen addresses = '*' # what I	
#port = 5432	
<pre>max_connections = 100</pre>	
<pre>#superuser_reserved_connections = 3</pre>	
<pre>#unix socket directories = '/tmp'</pre>	
<pre>#unix socket group = ''</pre>	
<pre>#unix socket permissions = 0777</pre>	
<pre>#bonjour = off</pre>	

🛄 说明 :

"listen\_addresses = '\*' ",表示监听所有 IP 地址。

步骤 14 配置完成后,执行以下命令,返回上级目录,重启数据库服务,使配置文件生效。

cd ..

/home/pgsql/bin/pg\_ctl -D pgsql/ stop

/home/pgsql/bin/pg\_ctl -D pgsql/ start

```
[postgres@ecs-postergre-test pgsql]$ cd ..
[postgres@ecs-postergre-test ~]$ /home/pgsql/bin/pg_ctl -D pgsql/ stop
waiting for server to shut down.... done
server stopped
[postgres@ecs-postergre-test ~]$ /home/pgsql/bin/pg_ctl -D pgsql/ start
waiting for server to start...2019-10-18 15:15:54.619 CST [7156] LOG: listening on
IPv4 address "0.0.0.0", port 5432
2019-10-18 15:15:54.619 CST [7156] LOG: listening on IPv6 address "::", port 5432
2019-10-18 15:15:54.626 CST [7156] LOG: listening on Unix socket "/tmp/.s.PGSQL.5432
"
2019-10-18 15:15:54.639 CST [7157] LOG: database system was shut down at 2019-10-18
15:15:49 CST
2019-10-18 15:15:54.643 CST [7156] LOG: database system is ready to accept connectio
ns
done
server started
```

--结束



# 4 安装 BenchmarkSQL 性能测试工具

## 实验介绍

本实验介绍在 x86 云服务器 ecs\_postgre\_scan\_test 上安装性能测试工具 BenchmarkSQL,为后续对数据库 PostgreSQL 进行性能测试做准备。

## 前提条件

- x86 云服务器已上电。
- x86 云服务器与 PC 之间网络互通, 且能访问外网。
- 云服务器上已安装 CentOS 7.6 操作系统, 且已获取 root 用户帐号和密码。
- PC 上已安装 SSH 远程登录工具 putty 和 WinSCP。
- 已获取性能测试工具源码文件 BenchmarkSQL.zip,下载地址:https://hcia-kunpengapplication-developer.obs.cn-north-1.myhwclouds.com/BenchmarkSQL.zip。

# 4.1 安装 jdk

## 4.1.1 操作步骤

步骤 1 打开 putty,登录 x86 云服务器 ecs\_postgre\_scan\_test,执行以下命令,查看是否有安装 jdk。

```
java -version
```

[root@	ecs-pos	stergre-:	scan-	-test	~]#	java	-version
-bash:	java:	command	not	found			

如上图所示,表示未安装jdk。

- 若已安装,则表示符合环境要求,结束。
- 否则,请执行步骤2,开始安装 jdk (本实验以安装 java-1.8.0-openjdk 为例进行介绍)。



```
步骤 2 执行以下命令, 安装 java-1.8.0-open jdk。
```

yum -y install java-1.8.0-openjdk

[root@ecs-postgre-scan-test ~]# yum -y install java-1.8.0-openjdk

步骤 3 安装完成后,执行以下命令,设置环境变量。

vi /etc/profile

[root@ecs-postergre-scan-test ~]# vi /etc/profile

🛄 说明 :

安装完之后,默认的安装目录在:/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.222.b10-1.el7\_7.x86\_64。

按"i"进入编辑模式,输入以下内容,按"Esc"退出编辑模式,输入":wq",保存退出。

```
# ---Set Java Environment---
JAVA_HOME=/usr/local/java/jdk1.8.0_181
JRE_HOME=/usr/local/java/jdk1.8.0_181/jre
CLASS_PATH=.:$JAVA_HOME/lib/dt.jar:$JAVA_HOME/lib/tools.jar:$JRE_HOME/lib
PATH=$PATH:$JAVA_HOME/bin:$JRE_HOME/bin
export JAVA_HOME JRE_HOME CLASS_PATH PATH
```

```
# ---Set Java Environment---
# ---Set Java Environment---
JAVA_HOME=/usr/local/java/jdk1.8.0_181
JRE_HOME=/usr/local/java/jdk1.8.0_181/jre
CLASS_PATH=.:$JAVA_HOME/lib/dt.jar:$JAVA_HOME/lib/tools.jar:$JRE_HOME/lib
PATH=$PATH:$JAVA_HOME/bin:$JRE_HOME/bin
export JAVA_HOME JRE_HOME CLASS_PATH PATH
# ---Set Java Environment---
```

步骤 4 执行以下命令, 使修改的配置生效。

source /etc/profile

[root@ecs-postergre-scan-test ~]# source /etc/profile

步骤 5 执行以下命令,再次查看 jdk 的版本。

java -version

[root@ed	cs-postergre-scan-test ~]# java -version	
openjdk	version "1.8.0 222"	
OpenJDK	Runtime Environment (build 1.8.0_222-b10)	
OpenJDK	64-Bit Server VM (build 25.222-b10, mixed mode)	

--结束



# 4.2 安装 BenchmarkSQL

## 4.2.1 操作步骤

步骤 1 参考 2.3.1 中的步骤 2~4, 打开 WinSCP, 将性能测试工具源码文件 BenchmarkSQL.zip 上传至 x86 云服务器 ecs\_postgre\_scan\_test 的/home 目录下。

월 D:\ - 121.36.1.167 - V	VinSCP												>
Local Mark Files Comm	nands Sessio	n Options Remote	Help										
🎛 😂 📚 Synchronize 🔳 🖬	P 🖬 🕸 🖻		Settings D	efault	<i>8</i> •								
📮 121.36.1.167 😅 New 🗄	Session												
D: Data	- 🖆 🔽 🖻 🖻	🗈 🔁 🗞 🔶				📙 hom	ie	- 👛 🛙	2 🖻 🗖 🕯	🕈 🖉 🝓 Find Files 🐁	<b>4</b> • -> -		
🗊 Upload 🕼 📝 Edit 🗙 🛛	🖌 🗣 Properti	es 📫 🖻 🛨 🖃 🗹				Dow	nload 🔬 🛙	≥ Edit ×	ad 🗣 Proj	oerties 🖆 🖻 া 🗉 🗹			
D:\						/home							
Name Ext	Size	Туре	Changed			Name	Ext		Siz	e Changed	Rights	Owne	r
\$RECYCLE.BIN		文件夹	2018/10/3	30 18:55:45						2019/10/14 15:17:0	03 r-xr-xr-x	root	
D-Box		文件夹	30% Uploa	iding				?	×				
📜 Email		文件夹						-					
HedExDownload		文件夹		<b>~</b>				Cano	el				
HedExInstall		文件夹						Minim	nize				
LenovoDrivers		文件夹											
<b>WRECYCLER</b>		文件夹	File:	D:\BenchmarkSQ	L.zip			Onco fini	chod				
System Volume Infor		文件夹	rarget.	/nome/				Chuce Inn	sileu.				
📜 temp		文件夹						Stay fulle					
WORK		文件夹	Time left:	0:00	:04 Time	e elapsed:	0:00:02						
BenchmarkSQL.zip	22,775 KiB	WinRAR ZIP archi	Bytes trans	ferred: 6,850	KiB Spee	ed:	3,620 KiB/s	Speed (K	iB/s):				
Porting-advisor-x86	91,065 KiB	WinRAR archive						Unlimited	1 ~				
postgresql-11.3.tar.gz	25,262 KiB	WinRAR archive	2019/10/	16 17:16:14									

### 步骤 2 执行以下命令,查看下载的 BenchmarkSQL 源码文件,并解压。

cd /home/ ls unzip BenchmarkSQL.zip [root@ecs-postergre-scan-test ~]# cd /home/ [root@ecs-postergre-scan-test home]# ls BenchmarkSQL.zip [root@ecs-postergre-scan-test home]# unzip BenchmarkSQL.zip

🛄 说明 :

BenchmarkSQL 不需要再编译安装,可以在 run 目录下,直接通过修改配置文件,连接 PostgreSQL 数据库。

--结束



# 5 PostgreSQL 性能测试

# 5.1 实验介绍

本实验介绍使用 BenchmarkSQL 性能测试工具对 PostgreSQL 数据库进行性能测试的方法。

## 5.2 前提条件

- x86 云服务器已上电。
- x86 云服务器与 PC 之间网络互通 , 且能访问外网。
- 云服务器上已安装 CentOS 7.6 操作系统,且已获取 root 用户帐号和密码。
- x86 云服务器上已安装 BenchmarkSQL 性能测试工具。
- 鲲鹏云服务器上已安装 PostgreSQL 数据库,并已创建数据库用户 postgres。
- PC 上已安装 SSH 远程登录工具 putty 和 WinSCP。

# 5.3 创建数据库 tpcc

## 5.3.1 操作步骤

步骤 1 返回 putty,登录鲲鹏云服务器 ecs\_postgre\_test,执行以下命令,切换到 postgres 用户,登录数据库。

```
su - postgres
/home/pgsql/bin/psql -U postgres
[root@ecs-postergre-test ~]# su - postgres
Last login: Fri Oct 18 16:46:48 CST 2019 on pts/1
[postgres@ecs-postergre-test ~]$ /home/pgsql/bin/psql -U postgres
psql (11.3)
Type "help" for help.
postgres=#
```



#### 步骤 2 执行以下命令, 查看已创建的数据库列表。

1				
		List of databa	ases	
Owner	Encoding	Collate	Ctype	Access privileges
postgres	UTF8	en US.UTF-8	en US.UTF-8	
postgres	UTF8 	en_US.UTF-8	en_US.UTF-8	=c/postgres   postgres=CTc/postgres
postgres	UTF8 	en_US.UTF-8	en_US.UTF-8	=c/postgres   postgres=CTc/postgres
	l Owner postgres postgres postgres	1 Owner   Encoding postgres   UTF8 postgres   UTF8   postgres   UTF8	l List of databa Owner   Encoding   Collate postgres   UTF8   en_US.UTF-8 postgres   UTF8   en_US.UTF-8     postgres   UTF8   en_US.UTF-8	l List of databases Owner   Encoding   Collate   Ctype postgres   UTF8   en_US.UTF-8   en_US.UTF-8 postgres   UTF8   en_US.UTF-8   en_US.UTF-8 postgres   UTF8   en_US.UTF-8   en_US.UTF-8

#### 步骤 3 执行以下命令, 创建数据库 tpcc, 并查看是否创建成功(先不要退出数据库)。

```
create database tpcc;
\1
  postgres=# create database tpcc;
CREATE DATABASE
  postgres=# \l
                                          List of databases
Collate |
               | Owner
                            | Encoding |
                                                               Ctype
                                                                              Access privileges
     Name
                                           en_US.UTF-8 | en_US.UTF-8
en_US.UTF-8 | en_US.UTF-8
                              UTF8
   template0
                 postgres
                              UTF8
                                                                            =c/postgres
   template1
                                           en_US.UTF-8
                 postgres
                                                                            =c/postgres
                                                                            postgres=CTc/postgres
                                           en_US.UTF-8
                                                           en_US.UTF-8
                              UTF8
  tpcc
  (4 rows)
```

--结束

# 5.4 PostgreSQL 性能测试

## 5.4.1 操作步骤

步骤1 返回 putty,登录 x86 云服务器 ecs\_postgre\_scan\_test,执行以下命令,添加 BenchmarkSQL 目录的执行权限。



步骤 2 执行以下命令,进入 BenchmarkSQL 的 run 目录,按实际情况修改配置文件。



修改以下内容,其余内容保持默认:



## 表5-1 数据库参数说明

参数	说明
	"121.36.22.80" 表示数据库服务器地址,按实际填写。
conn	"5432"表示PostgreSQL数据库端口,按实际填写。
	"tpcc"是数据库名称,按实际填写。
user	修改为创建数据库用户帐号。
password	修改为创建数据库用户密码。
warehouses	参数是初始化加载数据时,需要创建多少仓库的数据。例如输入10,则 创建10个仓库数据,每一个数据仓库的数据量大概是76823.04KB,可 能有少量的上下浮动,因为测试过程中将会插入或删除现有记录。
loadworkers	参数表示加载数据时,每次提交的进程数。
terminals	终端数量,指同时有多少终端并发执行,表示并发程度。
runTxnsPerTerminal	每分钟每个终端执行的事务数。
runMins	执行多少分钟。
limitTnxsPermin	每分钟执行的事务总数。
	用于指定终端和仓库的绑定模式,设置为true时可以运行4.x兼容模式, 意思为每个终端都有一个固定的仓库。设置为false时可以均匀的使用数 据库整体配置。
terminalWarehouseFixed	注意:
	runMins和runTxnsPerTerminal这两个参数指定了两种运行方式,前 者是按照指定运行时间执行,以时间为标准;后者以指定每个终端的事 务数为标准执行。两者不能同时生效,必须有一个设定为0。

按"i"进入编辑模式,如下图所示进行修改,按"Esc"退出编辑模式,输入":wq",保存退出。



db=postgres driver=orq.postgresql.Driver conn=jdbc:postgresql://121.36.22.80:5432/tpcc	7
user=postgres	
password=H	
warehouses=10	
loadWorkers=100	
terminals=1	
//To run specified transactions per terminal-	runMins must equal zero
runTxnsPerTerminal=100	
//To run for specified minutes- runTxnsPerTer	minal must equal zero
runMins=0	
//Number of total transactions per minute	
limitTxnsPerMin=300	
//Set to true to run in 4.x compatible mode.	Set to false to use the
//entire configured database evenly.	
terminalWarehouseFixed=true	

[root@ecs-postergre-scan-test run]# ./runDatabaseBuild.sh arm\_postgres.properties

#### 步骤 3 执行以下命令,初始化数据。

#### 1. 在/run/目录下,执行 runDatabaseBuild.sh 脚本,加载数据。

./runDatabaseBuild.sh arm\_postgres.properties

如下图所示的过程,需要等待较长时间。

Startin	ng Bei	nchmarkS(	QL LoadData			
driver=	=org.	postgres	ql.Driver			
conn=jo	dbc:p	ostgresq	l://121.36.	22.80:	5432/t	tpcc
user=po	ostgre	es				-
passwoi	rd=**:	*******	<del>k</del>			
- warehou	ises=	10				
loadWo	rkers	=100				
fileLoo	catio	n (not de	efined)			
csvNull	lValu	e (not de	efined - us	ing def	fault	'NULL')
Worker	000:	Loading	ттем			
Worker	001:	Loading	Warehouse	1		
Worker	002:	Loading	Warehouse	2		
Worker	003:	Loading	Warehouse	3		
Worker	004:	Loading	Warehouse	4		
Worker	005:	Loading	Warehouse	5		
Worker	006:	Loading	Warehouse	6		
Worker	007:	Loading	Warehouse	7		
Worker	008:	Loading	Warehouse			
Worker	010:	Loading	Warehouse	10		
Worker	009:	Loading	Warehouse	9		
Worker	000:	Loading	ITEM done			
Worker	008:	Loading	Warehouse	8	done	
Worker	007:	Loading	Warehouse	7	done	
Worker	001:	Loading	Warehouse	1	done	
Worker	002:	Loading	Warehouse	2	done	
Worker	004:	Loading	Warehouse	4	done	
Worker	009:	Loading	Warehouse	9	done	
Worker	006:	Loading	Warehouse	6	done	
Worker	005:	Loading	Warehouse	5	done	
Worker	010:	Loading	Warehouse	10	done	
Worker	003.	Loading	Warehouse	٦	done	

直到出现如下回显信息,表示数据加载成功。





2. 等待加载完成后,返回 putty,登录鲲鹏云服务器 ecs\_postgre\_test,执行以下命令,连接 tpcc 数据库,并查看数据库表。

\c tpcc

\dt								
<pre>postgres=# \c tpcc You are now connected to database "tpcc" as user "postgres". tpcc=# \dt List of relations</pre>								
Schema	Name	Type   Owner						
public   public   public   public   public   public   public   public   public   public   (10 rows)	bmsql_config bmsql_customer bmsql_district bmsql_history bmsql_item bmsql_new_order bmsql_oorder bmsql_order_line bmsql_stock bmsql_warehouse	<pre>  table   postgres   table   postgres</pre>						

tpcc=#

如上图所示,创建并初始化的表包括:9张表(warehouse, stock, item, order-line, new-order, history, distirct, customer, oorder)和1张配置表。

3. 执行以下命令,退出数据库 tpcc。



步骤 4 返回 putty,登录 x86 云服务器 ecs\_postgre\_scan\_test,执行以下命令,运行 BenchmarkSQL 程序,对 PostgreSQL 数据库进行压力测试。

./runBenchmark.sh	arm	_postgres.properties
-------------------	-----	----------------------

<pre>[root@ecs-postgre-scan-test run]# ./runBenchmark.sh arm_postgres</pre>	.properties
22:11:30,207 [main] INFO jTFCC: Term-00, C value for C LAST during load: 17	
22:11:30,207 [main] INFO jFECC: Term-00, C Value FOR C LASSI this run: 127 22:11:30,207 [main] INFO jFECC: Term-00, Term-00, Running Average tpmTOTAL: 305.84 Current tpmTOTAL: 696	Memory Usage: 15MB / 57MB
22:11:50,242 [Thread-1] INFO ]TFCC: Term-00, 22:11:50,242 [Thread-1] INFO ]TFCC: Term-00,	
22:11:50,242 [Thread-1] INFO ]TFCC: Term-00, Measured tpmTOTAL = 302.62	
22:11:50,242 [Thread-1] INFO jTFCC: Term-00, Session End = 2019-10-19 22:11:50 22:12:50,242 [Thread-1] INFO jTFCC: Term-00, Session End = 2019-10-19 22:11:50 20:11:60 242 [Thread-1] INFO jTFCC: Term-00 Termenation Count = 100	

记录 Running Average tpmTOTAL 值 "305.84"和 Current tpmTOTAL 值 "696"。

#### 步骤 5 执行以下命令,删除数据库和数据。

 $./{\tt runDatabaseDestroy.sh\ arm\_postgres.properties}$ 



<pre>[root@ecs-postgre-scan-test run]# ./runDatabaseDestroy.sh arm_postgres.properties</pre>	
#	
# Loading SQL file ./sql.common/tableDrops.sql	
#	
drop table bmsql_config;	
drop table bmsql_new_order;	
drop table bmsql_order_line;	
drop table bmsql_oorder;	
drop table bmsql_history;	
drop table bmsql_customer;	
drop table bmsql_stock;	
drop table bmsql_item;	
drop table bmsql_district;	
drop table bmsql_warehouse;	
drop sequence bmsql_hist_id_seq;	

--结束





PostgreSQL 性能调优

# 6.1 实验介绍

本实验介绍对 PostgreSQL 数据库进行性能调优的方法。

## 6.2 前提条件

- 鲲鹏云服务器已上电。
- 鲲鹏云服务器与 PC 之间网络互通, 且能访问外网。
- 云服务器上已安装 CentOS 7.6 操作系统,且已获取 root 用户帐号和密码。
- x86 云服务器上已安装 BenchmarkSQL 性能测试工具。
- 鲲鹏云服务器上已安装 PostgreSQL 数据库,并已创建数据库用户 postgres 和数据库 tpcc。
- PC 上已安装 SSH 远程登录工具 putty 和 WinSCP。

# 6.3 PostgreSQL 性能调优

## 6.3.1 操作步骤

步骤 1 返回 putty,登录鲲鹏云服务器 ecs\_postgre\_test,编译参数调优。

重新移植 PostgreSQL 数据,使用更高效率的 ARMv8 原子指令。

1. 执行以下命令,退出 postgres 用户,重新切换到 root 用户,进入/home 目录,删除以下目录。

```
exit
cd /home/
ls
rm -rf /home/postgresql-11.3
rm -rf /home/pgsql/
rm -rf /home/postgres/logfile
```


rm -rf /home/postgres/pgsql/

[root@ecs-postgre-test ~]# exit
logout
<pre>[root@ecs-postgre-test postgresql-11.3]# cd /home/</pre>
[root@ecs-postgre-test home]# ls
pgsql postgres postgresql-11.3 postgresql-11.3.tar.gz
<pre>[root@ecs-postgre-test home] # rm -rf /home/postgresql-11.3</pre>
[root@ecs-postgre-test home]# rm -rf /home/pgsql/
<pre>[root@ecs-postgre-test home]# rm -rf /home/postgres/logfile</pre>
<pre>[root@ecs-postgre-test home]# rm -rf /home/postgres/pgsql/</pre>

2. 参考 3.5.1 中的步骤 2~4, 重新解压 PostgreSQL 源码包, 生成 Makefile 文件。

3. 执行以下命令,修改 "src/Makefile.global" 文件。

vi src/Makefile.global

[root@ecs-postgre-test postgresql-11.3]# vi src/Makefile.global

输入":set nu"显示行号,在261行CFLAGS中,按"i"进入编辑模式,增加"march=armv8-a+crc+lse"(性能优化选项 enable lse),按"Esc"退出编辑模式,输入 ":wq!",保存退出。



4. 修改完成后,执行以下命令,安装 PostgreSQL 数据库。

make && make install

[root@ecs-postgre-test postgresql-11.3]# make && make install

步骤 2 参考 3.5.1 中的步骤 8~13,对 PostgreSQL 数据库进行初始化并启动,修改配置文件 pg\_hba.conf。

步骤 3 数据库参数调优。

对于不同业务场景,通过在调整数据库的参数配置,可以有效提升服务器性能。

1. 执行以下命令,打开配置文件 postgresql.conf。

```
vi postgresql.conf
```

[postgres@ecs-postgre-test pgsql]\$ vi postgresql.conf

2. 根据实际情况,按"i"进入编辑模式,如下图所示修改配置文件,并删除前面的"#",按 "Esc"退出编辑模式,输入":wq!",保存退出。

```
listen_addresses = '*'
bgwriter_delay = 10ms
bgwriter_lru_maxpages = 800
max_wal_size = 20GB
min_wal_size = 1GB
checkpoint_completion_target = 0.9
max_connections = 1000
checkpoint_timeout = 60min
full_page_writes = off
```



```
max_files_per_process = 100000
max_prepared_transactions = 2048
shared_buffers = 9GB
wal_buffers = 1GB
work_mem = 1GB
log_min_messages = FATAL
synchronous_commit = on
fsync = on
maintenance_work_mem = 2GB
vacuum_cost_limit = 10000
autovacuum = on
autovacuum_max_workers = 5
autovacuum_naptime = 20s
autovacuum_vacuum_scale_factor = 0.002
autovacuum_analyze_scale_factor = 0.001
```

### 表6-1 数据库参数说明

参数名称	参数含义	优化建议		
max_connections	允许客户端的最大并发连接数 目。	可根据实际情况设置。		
bgwriter_delay	后台写数据库进程的睡眠时 间。	设置为最小10ms。		
bgwriter_lru_maxpages	后台写数据库进程每次写脏数 据块时,写到外部文件中的脏 数据块的最大个数。			
max_wal_size	最大WAL x_log文件大小。	设置稍大以免频繁checkpoint。		
checkpoint_completion_target	表示checkpoint的完成时间要 在两个checkpoint间隔时间的 N%内完成。	设置长一点,避免对性能测试的影 响。		
checkpoint_timeout	检查点周期。			
full_page_writes	在checkpoint之后在对页面的 第一次写时将整个页面写到 wal里面。			
shared_buffers	可以被PostgreSQL用于缓存 数据的内存大小。	建议不超过物理内存60%,可根据 实际情况调整。		
wal_buffers	日志缓存区的大小。			
work_mem	数据库的排序操作和哈希表使 用的内存缓冲区的大小。			



log_min_messages	控制写到数据库日志文件中的 消息的级别。	
fsync	数据同步更新到磁盘。	默认打开 , 关闭会影响数据可靠 性。
synchronous_commit	同步提交日志。	默认打开 , 关闭会影响数据可靠 性。
maintenance_work_mem	数据库的维护操作使用的内存 空间的大小 , 如VACUUM、 CREATE INDEX和ALTER TABLE ADD FOREIGN KEY 等。	
autovacuum_max_workers	设置系统自动清理工作进程的 最大数量。	
autovacuum_naptime	设置两次系统自动清理操作之 间的间隔时间。	

- 步骤 4 参考 3.5.1 中的步骤 14, 重启数据库服务, 使配置文件生效。
- 步骤 5 参考 5.3.1 中的步骤 1~3 和 5.4.1 中的步骤 3~4, 返回 putty, 登录 x86 云服务器 ecs\_postgre\_scan\_test ,对 PostgreSQL 数据库重新进行性能测试。

40:10,360 [main] INFO jTPCC : Term-00,	
40:10,370 [Thread-0] ERROR OSCollector\$CollectData : OSCollector, unexpected EOF while reading from external helper process	
40:10,412 [main] INFO jTPCC : Term-00, C value for C LAST during load: 196	
40:10,412 [main] INFO jTPCC : Term-00, C value for C LAST this run: 83	
40:10,412 [main] INFO jTPCC : Term-00, Term-00, Running Average tpmTOTAL: 305.92 Current tpmTOTAL: 720 Memory Usage: 4MB / 57M	B
40:30,433 [Thread-1] INFO jTPCC : Term-00,	
40:30,433 [Thread-1] INFO jTPCC : Term-00,	
40:30,433 [Thread-1] INFO jTPCC : Term-00, Measured tpmC (NewOrders) = 116.91	
40:30,433 [Thread-1] INFO jTPCC : Term-00, Measured tpmTOTAL = 302.78	
40:30,433 [Thread-1] INFO jTPCC : Term-00, Session Start = 2019-10-21 15:40:10	
40:30,433 [Thread-1] INFO jTFCC: Term-00, Session End = 2019-10-21 15:40:30	
40:30,433 [Thread-1] INFO jTPCC · Term-00 Transaction Count = 100	

从调优后的性能测试数据可以看到, Running Average tpmTOTAL 值"305.92"和 Current tpmTOTAL 值"720"与调优前相比均有提升,已达到调优的效果。若要达到最优性能,需要 反复进行性能调优和测试,找到性能最优点。

--结束



# PostgreSQL 软件打包

# 7.1 实验介绍

通过将 PostgreSQL 源码制作成 RPM 包或 DEB 包的方式,可以实现 PostgreSQL 数据库在新鲲鹏云服务器上的快速安装部署。本实验介绍了制作 PostgreSQL 源码 RPM 包的方法。

🛄 <sub>说明:</sub>

制作 PostergreSQL 的 RPM 包是很复杂的过程,以下是精简的示例,不是完整的 RPM 包示例。

# 7.2 前提条件

- 鲲鹏云服务器已上电。
- 鲲鹏云服务器与 PC 之间网络互通 , 且能访问外网。
- 云服务器上已安装 CentOS 7.6 操作系统,且已获取 root 用户帐号和密码。
- PC 上已安装 SSH 远程登录工具 putty 和 WinSCP。

# 7.3 制作 RPM 包

### 7.3.1 操作步骤

步骤 1 参考 3.5.1 中的步骤 1, 返回 putty,登录鲲鹏云服务器 ecs\_postgre\_rpm\_test,执行以下命令,将鲲鹏云服务器 ecs\_postgre\_test 上的 PostgreSQL 源代码拷贝到鲲鹏云服务器 ecs\_postgre\_rpm\_test。

cd /usr/local/src/ scp root@121.36.22.80:/home/postgresql-11.3.tar.gz ./ # 此处 IP 地址请参照自己的实验环境 ecs postgre test 的弹性公网 IP。





步骤 2 执行以下命令,安装 RPM 包制作工具 rpmbuild。

```
yum -y install rpm-build
[root@ecs-postgre-rpm-test src]# yum -y install rpm-build
```

步骤 3 执行以下命令, 生成制作 RPM 包的相关目录。

rpmbuild -ba pgsql.spec

[root@ecs-postgre-rpm-test src]# rpmbuild -ba pgsql.spec error: failed to stat /usr/local/src/pgsql.spec: No such file or directory

🛄 说明:

若报错提示找不到 pgsql.spec,可忽略。执行以下命令,查看 root 目录,可以看到生成的 rpmbuild 目录以及相关的子目录。

ll /root/rpmbuild/

[root@ecs-	00	stgre-	-rpm-t	test :	src]#	ŧ 11	/root	/rpmbuild/
total 24								
drwxr-xr-x	2	root	root	4096	Oct	20	22:34	BUILD
drwxr-xr-x	2	root	root	4096	Oct	20	22:34	BUILDROOT
drwxr-xr-x	2	root	root	4096	Oct	20	22:34	RPMS
drwxr-xr-x	2	root	root	4096	Oct	20	22:34	SOURCES
drwxr-xr-x	2	root	root	4096	Oct	20	22:34	SPECS
drwxr-xr-x	2	root	root	4096	Oct	20	22:34	SRPMS

步骤 4 执行以下命令,将 PostgreSQL 源码包拷贝至 rpmbuild 的 SOURCE 目录下。

cp /usr/local/src/postgresql-11.3.tar.gz /root/rpmbuild/SOURCES/
[root@ecs-postgre-rpm-test src]# cp /usr/local/src/postgresql-11.3.tar.gz /root
rpmbuild/SOURCES/

步骤 5 执行以下命令,安装依赖包。

yum -y install readline-devel zlib-devel

[root@ecs-postgre-rpm-test src]# yum -y install readline-devel zlib-devel

步骤 6 执行以下命令,进入 rpmbuild 的 SPECS 目录,创建并打开编写制作 postgreSQL RPM 包的 spec 文件。

cd /root/rpmbuild/SPECS/ vi pgsql.spec [root@ecs-postgre-rpm-test src]# cd /root/rpmbuild/SPECS/ [root@ecs-postgre-rpm-test SPECS]# vi pgsql.spec

按 "i"进入编辑模式,输入以下内容,按 "Esc"退出编辑模式,输入 ":wq!",保存退出。 Name: \_\_\_\_postgresgl



```
Version: 11.3
Release: 1%{?dist}
Summary: This is a DB
License: PostgreSQL
URL:
        https://ftp.postgresql.org/
Source0: postgresql-11.3.tar.gz
BuildRequires: gcc
Requires: zlib-devel, readline-devel
%description
postgresql 11.3.
%prep
%setup -q
%build
%configure
make %{?_smp_mflags}
%install
make install DESTDIR=%{buildroot}
%files
%defattr(-,root,root,-)
%{ prefix}
%changelog
```

```
Jame:
          postgresql
          11.3
Version:
Release:
         1%{?dist}
Summary:
         This is a DB
License:
          PostgreSQL
          https://ftp.postgresql.org/
URL:
Source0: postgresql-11.3.tar.gz
BuildRequires:
                gcc
Requires: zlib-devel, readline-devel
%description
postgresql 11.3.
%prep
%setup -q
%build
%configure
make %{?_smp_mflags}
%install
make install DESTDIR=%{buildroot}
%files
%defattr(-,root,root,-)
{ prefix}
%changelog
```

### 步骤 7 执行以下命令,制作 RPM 包。

```
rpmbuild -ba pgsql.spec
[root@ecs-postgre-rpm-test SPECS]# rpmbuild -ba pgsql.spec
```

### 步骤 8 执行以下命令, 查看生成的 RPM 包。

ll /root/rpmbuild/RPMS/aarch64/



[root@ecs-postgre-rpm-test SPECS]# 11 /root/rpmbuild/RPMS/aarch64/ total 39488 -rw-r--r-- 1 root root 23036804 Oct 21 09:04 postgresql-11.3-1.el7.aarch64.rpm -rw-r--r-- 1 root root 17392204 Oct 21 09:04 postgresql-debuginfo-11.3-1.el7.aarch64.rpm

--结束



# **8** PostgreSQL 软件包安装测试

## 8.1 实验介绍

本实验介绍了在新鲲鹏云服务器上通过制作的 PostgreSQL 源码 RPM 包,实现 PostgreSQL 数据库的快速安装部署。

## 8.2 前提条件

- 鲲鹏云服务器已上电。
- 鲲鹏云服务器与 PC 之间网络互通, 且能访问外网。
- 云服务器上已安装 CentOS 7.6 操作系统,且已获取 root 用户帐号和密码。
- 鲲鹏云服务器上已完成 PostgreSQL 源码 RPM 包的制作。
- PC 上已安装 SSH 远程登录工具 putty 和 WinSCP。

## 8.2.1 RPM 包安装测试

### 8.2.2 操作步骤

步骤 1 返回 putty,登录鲲鹏云服务器 ecs\_postgre\_rpm\_test,执行以下命令,将制作的 RPM 包拷贝到/home 目录下。

```
cp /root/rpmbuild/RPMS/aarch64/* /home
cd /home/
ls
[root@ecs-postgre-rpm-test SPECS]# cp /root/rpmbuild/RPMS/aarch64/* /home
[root@ecs-postgre-rpm-test SPECS]# cd /home/
[root@ecs-postgre-rpm-test home]# ls
postgresql-11.3-1.el7.aarch64.rpm postgresql-debuginfo-11.3-1.el7.aarch64.rpm
```

步骤 2 执行以下命令,使用 RPM 包部署 PostgreSQL 数据库。

rpm -ivh postgresql-11.3-1.el7.aarch64.rpm --nodeps --force





### 步骤 3 执行以下命令,测试 postgreSQL 数据库。

创建 pgsql 运行用户和目录,并修改目录所属组。

```
useradd pgsql
cd /root/
mkdir -p /data/pgsql/data
chown -R pgsql:pgsql /data
```

[root@ecs-postgre-rpm-test home]# useradd pgsql [root@ecs-postgre-rpm-test home]# cd /root/ [root@ecs-postgre-rpm-test ~]# mkdir -p /data/pgsql/data [root@ecs-postgre-rpm-test ~]# chown -R pgsql:pgsql /data

切换至 pgsql 用户。

su - pgsql

[root@ecs-postgre-rpm-test ~]# su - pgsql

#### 初始化数据库。

initdb -D /data/pgsql/data

[pgsql@ecs-postgre-rpm-test ~]\$ initdb -D /data/pgsql/data The files belonging to this database system will be owned by user "pgsql". This user must also own the server process.

The database cluster will be initialized with locale "en\_US.UTF-8". The default database encoding has accordingly been set to "UTF8". The default text search configuration will be set to "english".

Data page checksums are disabled.

fixing permissions on existing directory /data/pgsql/data ... ok creating subdirectories ... ok selecting default max\_connections ... 100 selecting default shared\_buffers ... 128MB selecting dynamic shared memory implementation ... posix creating configuration files ... ok running bootstrap script ... ok performing post-bootstrap initialization ... ok syncing data to disk ... ok

WARNING: enabling "trust" authentication for local connections You can change this by editing pg\_hba.conf or using the option -A, or --auth-local and --auth-host, the next time you run initdb.

Success. You can now start the database server using:

pg\_ctl -D /data/pgsql/data -l logfile start

启动数据库。

pg ctl -D /data/pgsql/data -l logfile start



创建数据库。



createdb test [pgsql@ecs-postgre-rpm-test ~]\$ createdb test 进入数据库。 psql test [pgsql@ecs-postgre-rpm-test ~]\$ psql test psql (11.3) Type "help" for help. test=# 创建表。 CREATE TABLE COMPANY(ID INT PRIMARY KEY NOT NULL, NAME TEXT NOT NULL); cest=# CREATE TABLE COMPANY(ID INT PRIMARY KEY NOT NULL, NAME TEXT NOT NULL); CREATE TABLE test=# 插入记录 INSERT INTO COMPANY VALUES (1, 'Hello'); test=# INSERT INTO COMPANY VALUES (1, 'Hello'); INSERT 0 1 test=# 查询数据记录。 select \* from COMPANY; test=# select \* from COMPANY; id | name 1 | Hello (1 row) test=# 删除数据记录。 DELETE FROM COMPANY WHERE ID=1; test=# DELETE FROM COMPANY WHERE ID=1; DELETE 1 test=# 测试完成。 --结束