

1. 实验一 鲲鹏云 ECS 的构建及内核编译

Hollow Man

1. 实验介绍

本实验通过构建鲲鹏云 ECS、编译安装 openEuler 操作系统新内核以及简单的内核模块编程任务操作带领大家了解操作系统以及内核编程。

1. 任务描述

- 构建鲲鹏云 ECS
- 编译安装 openEuler 操作系统新内核
- 简单的内核模块编程实验，在内核模块中打印 “Hello, world!”

2. 实验目的

- 学习掌握如何安装构建 ECS
- 学习掌握如何编译操作系统内核
- 了解内核模块编程。

1.1 构建云实验环境

1.1.1 创建 VPC

在浏览器地址栏输入华为云控制台网址 `console.huaweicloud.com` 并按回车键，这时页面将跳转至登录页。

账号登录

账号名/邮箱

密码

手机号登录

记住登录名

登录

免费注册

忘记密码

IAM用户登录

HUAWEI ID登录


使用其他账号登录

按要求输入账号密码，进行登录。

注意：在此之前您需要在华为云主页注册华为云账号。

登录成功后会自动进入控制台页面，这时将区域选在 “华北-北京四”。



将鼠标悬停于左侧导航栏图标处展开服务列表，然后在服务列表中点击“虚拟私有云 VPC”项。



点击“虚拟私有云”控制台页面右上角的“创建虚拟私有云”按钮。



在创建虚拟私有云的页面中按照下表内容配置虚拟私有云参数。

参数	配置
区域	华北-北京四
名称	vpc-test
网段	192.168.1.0/24
企业项目	default
默认子网可用区	可用区1
默认子网名称	subnet-test
子网网段	如192.168.1.0/24

配置完成后，点击“立即创建”，创建完成后会自动回到 VPC 控制台。

点击 VPC 控制台左侧导航栏的“访问控制”→“安全组”，进入安全组控制台。



点击右上角的“创建安全组”。



在弹出的对话框中按“通用 Web 服务器”配置安全组参数，然后点击“确定”。

创建安全组

名称

sg-test

模板

通用Web服务器

描述


通用Web服务器，默认放通22、3389、80、443端口和ICMP协议。适用于需要远程登录、公网ping及用于网站服务的云服务器场景。

查看模板规则

确定

取消

1.1.1.2 购买 ECS

将鼠标悬停于左侧导航栏图标处展开服务列表。然后在服务列表中点击“弹性云服务器 ECS”项。



点击弹性云服务器 ECS 控制台页面右上角的“购买弹性云服务器 ECS”按钮进入购买页面。



按照下表内容配置弹性云服务器 ECS 的参数。

参数	配置
----	----

计费模式	按需计费
区域	华北-北京四
可用区	可用区1
CPU架构	鲲鹏计算
规格	鲲鹏通用计算增强型 kc1.xlarge.2 4vCPUs 8GB
镜像	公共镜像 openEuler openEuler 20.03 64bit with ARM(40GB)
系统盘	通用型SSD 40GB

注意：这里“区域”的配置是和 VPC 的区域配置保持一致的。

配置完成后点击 “下一步：网络配置” ， 进入网络配置，按下表配置网络参数。

参数	配置
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
线路	全动态BGP
公网带宽	按流量计费
带宽大小	5Mbit/s

配置完成后， 点击 “下一步：高级配置” ， 按下表配置 ECS 高级配置参数。

参数	配置
云服务器名称	openEuler （输入符合规则名称）
登录凭证	密码
密码	请输入8位以上包含大小写字母、数字和特殊字符的密码， 如openEuler@123
确认密码	请再次输入密码

云备份	暂不购买
云服务器组	不配置
高级选项	不勾选

配置完成后点击右下角“下一步：确认配置”。勾选同意协议，然后点击：
立即购买。

在提交任务成功后，点击“返回云服务器列表”，返回 ECS 控制台。

1.1.3 通过 ssh 登录系统

在 ECS 控制台查看 ECS 弹性公网 IP 地址。

<input type="checkbox"/>	名称/ID	监控	可用区	状态	规格/镜像	IP地址	计费...
<input type="checkbox"/>	openEuler 5e4b28a...		可用区1	运行中	4vCPUs 8GB kc1.xlarge.2 openEuler 20.03 64bit with ARM	121.36.... 192.168...	按需计费 2020/11/15 ...

在客户端机器操作系统里的 Console 控制台或 Terminal 终端里运行 ssh 命令：

```
$ ssh root@121.36.45.64
```

(注意：此处的 IP 地址 121.36.45.64 即是刚刚购买的弹性公网 IP 地址。)

在客户端（本地 PC）第一次登录时会有安全性验证的提示：

```
The authenticity of host '119.8.238.181 (119.8.238.181)' can't be established.
ECDSA key fingerprint is SHA256:RVxC1cSuMmqLtWdMw4n6f/VPsfWLkT/zDMT2q4qWxc0.
Are you sure you want to continue connecting (yes/no)? yes
```

在这里输入 yes 并按回车键继续：

```
Warning: Permanently added '119.8.238.181' (ECDSA) to the list of known hosts.
```

```
Authorized users only. All activities may be monitored and reported.
root@119.8.238.181's password:
```

输入密码(注意这里不会有任何回显)并回车，登录后的界面如下所示：

```
Welcome to Huawei Cloud Service
```

```
Last login: Mon May 18 15:35:37 2020
```

```
Welcome to Huawei Cloud Service
```

```
Last login: Mon May 18 15:35:37 2020
```

Welcome to 4.19.90-2003.4.0.0036.oe1.aarch64

System information as of time: Sun Nov 15 14:41:58 CST 2020

System load: 0.15
Processes: 131
Memory used: 5.0%
Swap used: 0.0%
Usage On: 9%
IP address: 192.168.1.5
Users online: 1

```
[root@openeuler ~]#
```

修改主机名

ECS 创建时被命名为 “openEuler” ,所以系统默认 hostname 为 “openeuler” ,为了和本实验手册另外两个版本保持行文上的一致,我们可以将主机名改为 “openEuler” 或 “localhost” (一般在虚拟机中,主机名被默认为 localhost,而 ECS 也是虚拟机。本文的上下文环境中可能同时用到这三种名称,请鉴别) :

```
[root@openeuler ~]# vi /etc/hostname
```

```
[root@openeuler ~]# cat /etc/hostname  
openEuler
```

```
[root@openeuler ~]# reboot
```

修改完成后重启系统并重新登录。



3. 实验任务

1.1.4 openEuler 内核编译与安装

安装工具, 构建开发环境:

```
[root@openEuler ~]# yum group install -y "Development Tools"
```

```
[root@openEuler ~]# yum install -y bc
[root@openEuler ~]# yum install -y openssl-devel
```

```
OpenSSH SSH client
Running transaction
Preparing : 1/1
Installing : zlib-devel-1.2.11-17.oe1.x86_64 1/9
Installing : pcre2-devel-10.33-2.oe1.x86_64 2/9
Installing : libveto-devel-0.3.1-2.oe1.x86_64 3/9
Installing : libsepol-devel-2.9-1.oe1.x86_64 4/9
Installing : libselinux-devel-2.9-1.oe1.x86_64 5/9
Installing : keyutils-libs-devel-1.5.10-11.oe1.x86_64 6/9
Installing : e2fsprogs-devel-1.45.3-4.oe1.x86_64 7/9
Running scriptlet: e2fsprogs-devel-1.45.3-4.oe1.x86_64 7/9
Installing : krb5-devel-1.17-9.oe1.x86_64 8/9
Installing : openssl-devel-1:1.1.1d-9.oe1.x86_64 9/9
Running scriptlet: openssl-devel-1:1.1.1d-9.oe1.x86_64 9/9
Verifying : e2fsprogs-devel-1.45.3-4.oe1.x86_64 1/9
Verifying : keyutils-libs-devel-1.5.10-11.oe1.x86_64 2/9
Verifying : krb5-devel-1.17-9.oe1.x86_64 3/9
Verifying : libselinux-devel-2.9-1.oe1.x86_64 4/9
Verifying : libsepol-devel-2.9-1.oe1.x86_64 5/9
Verifying : libveto-devel-0.3.1-2.oe1.x86_64 6/9
Verifying : openssl-devel-1:1.1.1d-9.oe1.x86_64 7/9
Verifying : pcre2-devel-10.33-2.oe1.x86_64 8/9
Verifying : zlib-devel-1.2.11-17.oe1.x86_64 9/9

Installed:
openssl-devel-1:1.1.1d-9.oe1.x86_64  e2fsprogs-devel-1.45.3-4.oe1.x86_64  keyutils-libs-devel-1.5.10-11.oe1.x86_64
krb5-devel-1.17-9.oe1.x86_64  libselinux-devel-2.9-1.oe1.x86_64  libsepol-devel-2.9-1.oe1.x86_64
libveto-devel-0.3.1-2.oe1.x86_64  pcre2-devel-10.33-2.oe1.x86_64  zlib-devel-1.2.11-17.oe1.x86_64

Complete!
[root@openEuler ~]#
```

备份 boot 目录以防后续步骤更新内核失败

```
[root@openEuler ~]# tar czvf boot.origin.tgz /boot/
```

保存当前内核版本信息

```
[root@openEuler ~]# uname -r > uname_r.log
```

```
OpenSSH SSH client
/boot/grub2/themes/starfield/boot_menu_nw.png
/boot/grub2/themes/starfield/terminal_box_ne.png
/boot/grub2/themes/starfield/dejavu_14.pf2
/boot/grub2/themes/starfield/starfield.png
/boot/grub2/themes/starfield/theme.txt
/boot/grub2/themes/starfield/COPYING.CC-BY-SA-3.0
/boot/grub2/themes/starfield/dejavu_16.pf2
/boot/grub2/themes/starfield/terminal_box_e.png
/boot/grub2/themes/starfield/slider_s.png
/boot/grub2/themes/starfield/boot_menu_w.png
/boot/grub2/themes/starfield/terminal_box_w.png
/boot/grub2/themes/starfield/terminal_box_sw.png
/boot/grub2/themes/starfield/boot_menu_se.png
/boot/grub2/themes/starfield/dejavu_10.pf2
/boot/grub2/themes/starfield/boot_menu_c.png
/boot/grub2/themes/starfield/dejavu_bold_14.pf2
/boot/grub2/grubenv
/boot/grub2/device.map
/boot/System.map-4.19.90-2003.4.0.0036.oe1.x86_64
/boot/loader/
/boot/loader/entries/
/boot/efi/
/boot/efi/EFI/
/boot/efi/EFI/openEuler/
/boot/.vmlinuz-4.19.90-2003.4.0.0036.oe1.x86_64.hmac
/boot/vmlinuz-0-rescue-1d6c803a58f6428fbee2d85ccf87dbd5
/boot/vmlinuz-4.19.90-2003.4.0.0036.oe1.x86_64
/boot/initramfs-4.19.90-2003.4.0.0036.oe1.x86_64.img
[root@openEuler ~]# uname -r > uname_r.log
uname: extra operand '-r'
```

1. 获取内核源代码并解压

```
[root@openEuler ~]# wget https://gitee.com/openeuler/kernel/repository/archive/kernel-4.19.zip
```

```
[root@openEuler ~]# unzip kernel-4.19.zip
```



```
OpenSSH SSH client
kernel/scripts/dtc/include-prefixes/dt-bindings -> ../../../../include/dt-bindings
kernel/scripts/dtc/include-prefixes/h8300 -> ../../../../arch/h8300/boot/dts
kernel/scripts/dtc/include-prefixes/microblaze -> ../../../../arch/microblaze/boot/dts
kernel/scripts/dtc/include-prefixes/mips -> ../../../../arch/mips/boot/dts
kernel/scripts/dtc/include-prefixes/nios2 -> ../../../../arch/nios2/boot/dts
kernel/scripts/dtc/include-prefixes/openrisc -> ../../../../arch/openrisc/boot/dts
kernel/scripts/dtc/include-prefixes/powerpc -> ../../../../arch/powerpc/boot/dts
kernel/scripts/dtc/include-prefixes/sh -> ../../../../arch/sh/boot/dts
kernel/scripts/dtc/include-prefixes/xtensa -> ../../../../arch/xtensa/boot/dts
kernel/tools/testing/selftests/powerpc/copyloops/copyuser_64.S -> ../../../../arch/powerpc/lib/copyuser_64.S
kernel/tools/testing/selftests/powerpc/copyloops/copyuser_power7.S -> ../../../../arch/powerpc/lib/copyuser_power7.S
kernel/tools/testing/selftests/powerpc/copyloops/memcpy_64.S -> ../../../../arch/powerpc/lib/memcpy_64.S
kernel/tools/testing/selftests/powerpc/copyloops/memcpy_power7.S -> ../../../../arch/powerpc/lib/memcpy_power7.S
kernel/tools/testing/selftests/powerpc/primitives/asm/asm-compat.h -> ../../../../arch/powerpc/include/asm/asm-compat.h
kernel/tools/testing/selftests/powerpc/primitives/asm/asm-const.h -> ../../../../arch/powerpc/include/asm/asm-const.h
kernel/tools/testing/selftests/powerpc/primitives/asm/feature-fixups.h -> ../../../../arch/powerpc/include/asm/feature-fixups.h
kernel/tools/testing/selftests/powerpc/primitives/asm/ppc_asm.h -> ../../../../arch/powerpc/include/asm/ppc_asm.h
kernel/tools/testing/selftests/powerpc/primitives/word-at-a-time.h -> ../../../../arch/powerpc/include/asm/word-at-a-time.h
kernel/tools/testing/selftests/powerpc/stringloops/memcmp_32.S -> ../../../../arch/powerpc/lib/memcmp_32.S
kernel/tools/testing/selftests/powerpc/stringloops/memcmp_64.S -> ../../../../arch/powerpc/lib/memcmp_64.S
kernel/tools/testing/selftests/powerpc/stringloops/strlen_32.S -> ../../../../arch/powerpc/lib/strlen_32.S
kernel/tools/testing/selftests/powerpc/vphn/vphn.c -> ../../../../arch/powerpc/mm/vphn.c
kernel/tools/testing/selftests/powerpc/vphn/vphn.h -> ../../../../arch/powerpc/mm/vphn.h
[root@openEuler ~]#
```

2. 编译内核

```
[root@openEuler ~]# cd kernel
[root@openEuler kernel]# make openeuler_defconfig
```

在这里，我们按源代码文件 `kernel/arch/arm64/configs/openeuler_defconfig` 的配置配置内核。

```
[root@openEuler kernel]# make help | grep Image
* Image.gz          - Compressed kernel image (arch/arm64/boot/Image.gz)
Image               - Uncompressed kernel image (arch/arm64/boot/Image)
```

这一步查看了可编译的 Image。

```
OpenSSH SSH client
kernel/tools/testing/selftests/powerpc/copyloops/memcpy_power7.S -> ../../../../arch/powerpc/lib/memcpy_power7.S
kernel/tools/testing/selftests/powerpc/primitives/asm/asm-compat.h -> ../../../../arch/powerpc/include/asm/asm-compat.h
kernel/tools/testing/selftests/powerpc/primitives/asm/asm-const.h -> ../../../../arch/powerpc/include/asm/asm-const.h
kernel/tools/testing/selftests/powerpc/primitives/asm/feature-fixups.h -> ../../../../arch/powerpc/include/asm/feature-fixups.h
kernel/tools/testing/selftests/powerpc/primitives/asm/ppc_asm.h -> ../../../../arch/powerpc/include/asm/ppc_asm.h
kernel/tools/testing/selftests/powerpc/primitives/word-at-a-time.h -> ../../../../arch/powerpc/include/asm/word-at-a-time.h
kernel/tools/testing/selftests/powerpc/stringloops/memcmp_32.S -> ../../../../arch/powerpc/lib/memcmp_32.S
kernel/tools/testing/selftests/powerpc/stringloops/memcmp_64.S -> ../../../../arch/powerpc/lib/memcmp_64.S
kernel/tools/testing/selftests/powerpc/stringloops/strlen_32.S -> ../../../../arch/powerpc/lib/strlen_32.S
kernel/tools/testing/selftests/powerpc/vphn/vphn.c -> ../../../../arch/powerpc/mm/vphn.c
kernel/tools/testing/selftests/powerpc/vphn/vphn.h -> ../../../../arch/powerpc/mm/vphn.h
[root@openEuler ~]# cd kernel
[root@openEuler kernel]# make openeuler_defconfig
HOSTCC scripts/basic/fixdep
HOSTCC scripts/kconfig/conf.o
YACC scripts/kconfig/zconf.tab.c
LEX scripts/kconfig/zconf.lex.c
HOSTCC scripts/kconfig/zconf.tab.o
HOSTLD scripts/kconfig/conf
arch/x86/configs/openeuler_defconfig:2531:warning: override: reassigning to symbol HINIC
#
# configuration written to .config
#
[root@openEuler kernel]# make help | grep Image
* bzImage          - Compressed kernel image (arch/x86/boot/bzImage)
```

```
[root@openEuler kernel]# make -j4 Image modules dtbs
```

这一步是编译内核的 Image、modules 和 dtbs。

```
OpenSSH SSH client
HOSTCC /root/kernel/tools/objtool/fixdep.o
HOSTLD /root/kernel/tools/objtool/fixdep-in.o
LINK /root/kernel/tools/objtool/fixdep
CC /root/kernel/tools/objtool/exec-cmd.o
CC /root/kernel/tools/objtool/help.o
CC /root/kernel/tools/objtool/pager.o
CC /root/kernel/tools/objtool/parse-options.o
CC /root/kernel/tools/objtool/run-command.o
CC /root/kernel/tools/objtool/sigchain.o
CC /root/kernel/tools/objtool/subcmd-config.o
LD /root/kernel/tools/objtool/libsubcmd-in.o
AR /root/kernel/tools/objtool/libsubcmd.a
GEN /root/kernel/tools/objtool/arch/x86/lib/inat-tables.c
CC /root/kernel/tools/objtool/arch/x86/decode.o
LD /root/kernel/tools/objtool/arch/x86/objtool-in.o
CC /root/kernel/tools/objtool/builtin-check.o
CC /root/kernel/tools/objtool/builtin-orc.o
CC /root/kernel/tools/objtool/check.o
make: *** No rule to make target 'dtbs'. Stop.
make: *** Waiting for unfinished jobs....
CC /root/kernel/tools/objtool/orc_gen.o
CC /root/kernel/tools/objtool/orc_dump.o
CC /root/kernel/tools/objtool/elf.o
CC /root/kernel/tools/objtool/special.o
CC /root/kernel/tools/objtool/objtool.o
CC /root/kernel/tools/objtool/libstring.o
CC /root/kernel/tools/objtool/str_error_r.o
LD /root/kernel/tools/objtool/objtool-in.o
LINK /root/kernel/tools/objtool/objtool
[root@openEuler kernel]#
```

3. 安装内核

```
[root@openEuler kernel]# make modules_install
.....
INSTALL sound/soundcore.ko
DEPMOD 4.19.154
[root@openEuler kernel]# make install
/bin/sh ./arch/arm64/boot/install.sh 4.19.154 \
arch/arm64/boot/Image System.map "/boot"
dracut-install: Failed to find module 'xen-blkfront'
dracut: FAILED: /usr/lib/dracut/dracut-install -D /var/tmp/dracut.tlIdPu/initramfs --kernel
dir /lib/modules/4.19.154/ -m virtio_gpu xen-blkfront xen-netfront virtio_blk virtio_scsi virtio_net
virtio_pci virtio_ring virtio
```

注意：在最后一步“make install”时出现的错误在这里可以忽略。

4. 以 VNC 登录 ECS

<input type="checkbox"/>	名称/ID	监控	可用区	状态	规格/镜像
<input type="checkbox"/>	openEuler 5e4b28a...				openEuler 5e4b28a8-e74c-4e62-8df5-f911704b7df1 2 vCPUs 8GB kc1.xlarge.2 openEuler 20.03 64bit with ARM

在控制台“弹性云服务器 ECS”的页面中点击刚刚创建的虚拟机“openEuler”的名字超链接，在新打开的页面中点击“远程登录”按钮：



然后以控制台提供的 VNC 方式登录：

登录Linux弹性云服务器

使用CloudShell登录 **New!**

优势：操作更流畅，命令支持复制粘贴
绑定弹性公网IP的云服务器推荐使用此登录方式。
请确保安全组已放通CloudShell连接实例使用的端口（默认使用22端口）

CloudShell登录

其他方式

1、使用控制台提供的VNC方式登录

立即登录

与以 ssh 登录一样，以 root 身份登录：

```
Authorized users only. All activities may be monitored and reported.
openEuler login: [ 429.483128] systemd-rc-local-generator[2892]: /etc/rc.d/rc.local is not marked executable, skipping.
openEuler login: root
Password:
Last login: Sun Nov 15 14:51:43 from 119.3.119.18

      Welcome to Huawei Cloud Service

Welcome to 4.19.90-2003.4.0.0036.oe1.aarch64

System information as of time: Sun Nov 15 15:51:58 CST 2020

System load:   0.00
Processes:    122
Memory used:   5.2%
Swap used:     0.0%
Usage On:      38%
IP address:    192.168.1.5
Users online:  2

[root@openEuler ~]# _
```

大部分的时间，我们仅将此作为一个监视器使用。

5. 重启系统

在 ssh 终端重启操作系统：

```
[root@openEuler kernel]# reboot
```

登录并验证

在 VNC 窗口中选择以新编译出来的内核启动系统：

```
openEuler (4.19.154) 20.03 (LTS)
openEuler (4.19.90-2003.4.0.0036.oe1.aarch64) 20.03 (LTS)
openEuler (0-rescue-95148c976dae4cd0bfb15a0242465b8a) 20.03 (LTS)
System setup

Use the ▲ and ▼ keys to change the selection.
Press 'e' to edit the selected item, or 'c' for a command prompt.
```

在这里新编译出来的内核版本为 4.19.154。您的子版本号可能与此不一样。

6. 登录系统并查看版本

请以 VNC 和 ssh 终端登录系统，并在其中之一查看内核版本：

```
[root@openEuler ~]# uname -r
4.19.154
```

可以看出内核版本已更新。

1.1.5 Hello, world!

7. 正确编写满足功能的源文件，包括.c 源文件和 Makefile 文件。在这里我们的示例源文件存放在 tasks_k/1/task3 目录下。

```
[root@openEuler ~]# cd tasks_k/1/task3
[root@openEuler task1]# ls
helloworld.c  Makefile
```



tasks_k.zip

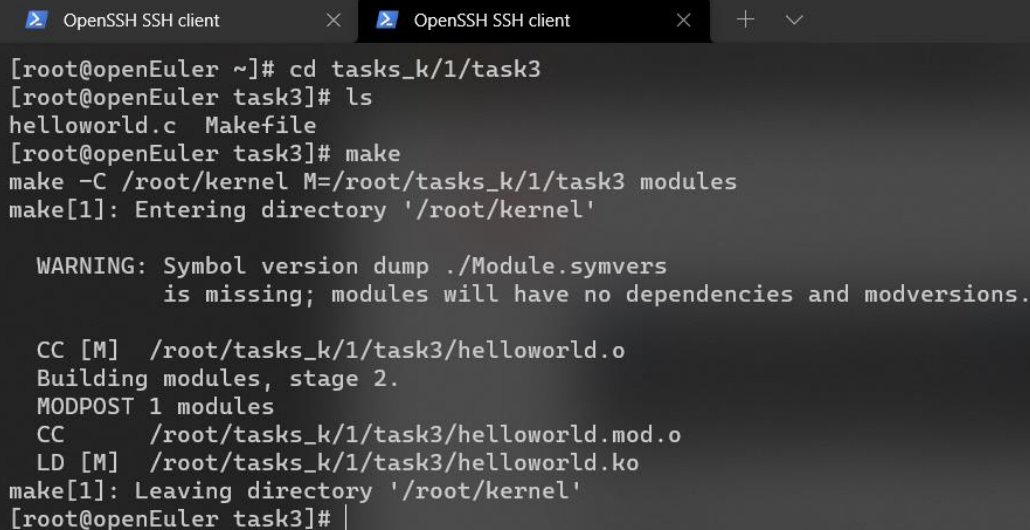
```
OpenSSH SSH client  ×  OpenSSH SSH client  ×  +  ▼
[root@openEuler ~]# cd tasks_k/1/task3
[root@openEuler task3]# ls
helloworld.c  Makefile
```

实验中的源文件可以参考以上压缩包中内容（您可以用 scp 命令将其上传到 ECS）。

8. 编译源文件

```
[root@openEuler task3]# make
make -C /root/kernel M=/root/tasks_k/1/task3 modules
```

```
make[1]: Entering directory '/root/kernel'
CC [M] /root/tasks_k/1/task3/helloworld.o
Building modules, stage 2.
MODPOST 1 modules
CC /root/tasks_k/1/task3/helloworld.mod.o
LD [M] /root/tasks_k/1/task3/helloworld.ko
make[1]: Leaving directory '/root/kernel'
```



```
OpenSSH SSH client x OpenSSH SSH client x + v
[root@openEuler ~]# cd tasks_k/1/task3
[root@openEuler task3]# ls
helloworld.c Makefile
[root@openEuler task3]# make
make -C /root/kernel M=/root/tasks_k/1/task3 modules
make[1]: Entering directory '/root/kernel'

WARNING: Symbol version dump ./Module.symvers
is missing; modules will have no dependencies and modversions.

CC [M] /root/tasks_k/1/task3/helloworld.o
Building modules, stage 2.
MODPOST 1 modules
CC /root/tasks_k/1/task3/helloworld.mod.o
LD [M] /root/tasks_k/1/task3/helloworld.ko
make[1]: Leaving directory '/root/kernel'
[root@openEuler task3]# |
```

9. 加载编译完成的内核模块，并查看加载结果。

```
[root@openEuler task3]# insmod helloworld.ko
[root@openEuler task3]# lsmod | grep helloworld
helloworld                262144  0
```

10. 卸载内核模块，并查看结果。

```
[root@openEuler task3]# rmmod helloworld
[root@openEuler task3]# dmesg | tail -n5
[ 708.247970] helloworld: loading out-of-tree module taints kernel.
[ 708.248513] helloworld: module verification failed: signature and/or required key missing -
tainting kernel
[ 708.249859] hello_init
[ 708.250043] Hello, world!
[ 747.518247] hello_exit
```

您在 VNC 窗口中也会看到同样的结果。

(在这里，请忽略掉最开始安装模块时出现的两行错误提示信息。)

11. 在虚拟机和 VNC 窗口中退出登录

```
[root@openEuler ~]# exit
```

注意：一般情况下不要 shutdown 虚拟机，若 shutdown 了虚拟机，需要联系实验管理员重启该虚拟机。

12. 关闭 VNC 客户端页面

1.2 云环境资源清理

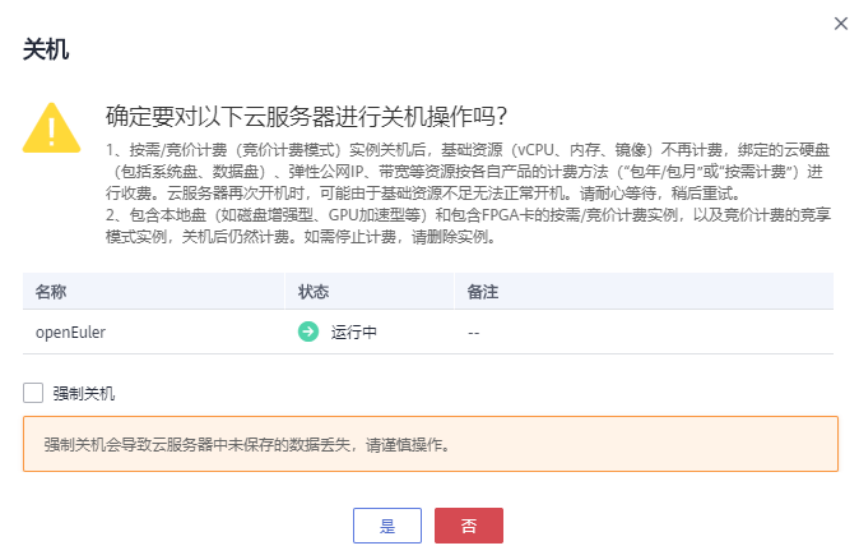
1.2.1 ECS 关机

当所实验完成后，应该对 ECS 进行关机以节约经费（ECS 关机后仍有少量扣费）。

回到 ECS 控制台，勾选 openEuler 虚拟机，进行关机。



在弹出的对话框中点击“是”按钮：



点击“是”按钮即可进行关机。

1.2.2 删除 ECS

可以等到所有的内核实验完成后再删除 ECS，否则每次都得重新编译内核。这里给出删除 ECS 的方法。

待关机完成后点击“更多” → “删除”



在弹出的对话框中勾选“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 ECS。



您可以在控制台点击“更多 | 资源 | 我的资源”菜单项，检查资源是否全部删除



注意：(1) 虚拟私有云 VPC 和安全组可以不删除，以留下次使用。(2) 若在除“华北-北京四”之外区域购买了 ECS 和 EIP，请切换到那个区域查看。