

镜像服务

用户指南

文档版本 01
发布日期 2024-12-18



版权所有 © 华为云计算技术有限公司 2024。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



HUAWEI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云计算技术有限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，华为云计算技术有限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。

目录

1 产品介绍	1
1.1 什么是镜像服务	1
1.2 相关支持列表	2
1.2.1 弹性云服务器类型与支持的操作系统版本	2
1.2.2 外部镜像文件支持的格式和操作系统类型	8
1.2.3 支持 UEFI 启动方式的操作系统版本	16
1.3 IMS 权限管理	18
1.4 基本概念	20
1.4.1 区域和可用区	20
1.4.2 镜像常见格式	21
1.5 镜像服务与其他服务的关系	22
2 通过 IAM 授予使用 IMS 的权限	24
2.1 创建用户并授权使用 IMS	24
2.2 IMS 自定义策略	25
3 创建私有镜像	28
3.1 创建方式导航	28
3.2 通过云服务器创建 Windows 系统盘镜像	28
3.3 通过云服务器创建 Linux 系统盘镜像	31
3.4 通过外部镜像文件创建 Windows 系统盘镜像	34
3.4.1 流程概览 (Windows)	34
3.4.2 准备镜像文件 (Windows)	35
3.4.3 上传镜像文件 (Windows)	37
3.4.4 注册镜像 (Windows)	37
3.4.5 使用镜像创建弹性云服务器 (Windows)	39
3.5 通过外部镜像文件创建 Linux 系统盘镜像	39
3.5.1 流程概览 (Linux)	39
3.5.2 准备镜像文件 (Linux)	40
3.5.3 上传镜像文件 (Linux)	42
3.5.4 注册镜像 (Linux)	43
3.5.5 使用镜像创建弹性云服务器 (Linux)	45
3.6 通过数据盘创建数据盘镜像	45
3.7 通过外部镜像文件创建数据盘镜像	47

3.8 通过云服务器创建整机镜像.....	49
3.9 通过云服务器备份创建整机镜像.....	52
3.10 通过云备份创建整机镜像.....	53
3.11 通过 ISO 文件创建 Windows 系统盘镜像.....	54
3.11.1 创建流程（Windows）.....	54
3.11.2 为原始 ISO 文件集成 virtio 驱动.....	56
3.11.3 将 ISO 文件注册为镜像（Windows）.....	58
3.11.4 使用 ISO 镜像创建 Windows 云服务器.....	59
3.11.5 安装 Windows 操作系统和 virtio 驱动.....	60
3.11.6 配置云服务器并创建 Windows 系统盘镜像.....	69
3.12 通过 ISO 文件创建 Linux 系统盘镜像.....	70
3.12.1 创建流程（Linux）.....	70
3.12.2 将 ISO 文件注册为镜像（Linux）.....	72
3.12.3 使用 ISO 镜像创建 Linux 云服务器.....	73
3.12.4 安装 Linux 操作系统.....	73
3.12.5 配置云服务器并创建 Linux 系统盘镜像.....	78
3.13 快速导入镜像文件.....	79
3.13.1 流程概览.....	79
3.13.2 快速导入镜像文件（基于 Linux 环境）.....	82
3.13.3 快速导入镜像文件（基于 Windows 环境）.....	86
4 管理私有镜像.....	89
4.1 通过镜像创建云服务器.....	89
4.2 修改镜像属性.....	90
4.3 导出镜像.....	91
4.4 导出镜像列表信息.....	92
4.5 查看镜像的磁盘容量.....	93
4.6 删除镜像.....	94
4.7 共享镜像.....	95
4.7.1 共享镜像概述.....	95
4.7.2 获取项目 ID.....	95
4.7.3 共享指定镜像.....	96
4.7.4 接受或拒绝共享镜像.....	97
4.7.5 拒绝已经接受的共享镜像.....	98
4.7.6 接受已经拒绝的共享镜像.....	99
4.7.7 取消共享镜像.....	99
4.7.8 添加镜像的共享租户.....	100
4.7.9 删除镜像的共享租户.....	101
4.7.10 复制共享镜像.....	101
4.8 复制镜像.....	102
4.9 优化私有镜像（Windows）.....	102
4.9.1 优化过程（Windows）.....	103
4.9.2 查看 Windows 操作系统云服务器虚拟化类型.....	103

4.9.3 相关软件及获取方式.....	104
4.9.4 安装 PV driver.....	104
4.9.5 安装 virtio 驱动.....	106
4.9.6 清除系统日志.....	112
4.10 优化私有镜像 (Linux)	112
4.10.1 优化过程 (Linux)	113
4.10.2 确认是否需要优化私有镜像.....	113
4.10.3 在 Linux 系统中卸载 PV driver.....	115
4.10.4 修改 grub 文件磁盘标识方式为 UUID.....	116
4.10.5 修改 fstab 文件磁盘标识方式为 UUID.....	120
4.10.6 安装原生的 XEN 和 KVM 驱动.....	121
4.10.7 安装原生的 KVM 驱动.....	128
4.10.8 清除日志文件.....	134
4.11 转换镜像格式.....	135
5 Windows 操作系统相关操作.....	140
5.1 设置网卡属性为 DHCP (Windows)	140
5.2 开启远程桌面连接功能.....	142
5.3 安装并配置 Cloudbase-Init 工具.....	143
5.4 安装一键式重置密码插件 (Windows)	148
5.5 执行 Sysprep.....	151
6 Linux 操作系统相关操作.....	155
6.1 设置网卡属性为 DHCP (Linux)	155
6.2 清理网络规则文件.....	157
6.3 安装 Cloud-Init 工具.....	158
6.4 配置 Cloud-Init 工具.....	164
6.5 安装一键式重置密码插件 (Linux)	168
6.6 安装 Linux 特殊驱动.....	171
6.7 卸载云服务器的数据盘.....	172
7 标签管理.....	174
8 配额管理.....	176
9 使用 CTS 审计 IMS 操作事件.....	177
9.1 IMS 支持审计的关键操作列表.....	177
9.2 查看 IMS 审计日志.....	178
10 常见问题.....	180
10.1 镜像咨询类.....	180
10.1.1 镜像怎么选?	180
10.1.2 镜像和备份有什么区别?	181
10.1.3 可以裁剪镜像吗?	182
10.1.4 如何备份云服务器当前状态, 方便以后系统故障时进行恢复?	182
10.1.5 创建的私有镜像如何使用到已有的云服务器上?	183

10.1.6 数据盘镜像中的数据可以导入到之前已有的数据盘内吗？	183
10.1.7 能否跨账号使用私有镜像？	183
10.2 镜像创建类	183
10.2.1 创建镜像 FAQ	183
10.2.2 整机镜像 FAQ	184
10.2.3 使用备份创建镜像和使用云服务器创建镜像是否有区别？	185
10.2.4 为什么创建云服务器或者为云服务器切换操作系统时选不到 ISO 镜像？	185
10.2.5 弹性云服务器存在跨区卷时如何制作整机镜像？	185
10.2.6 Windows 操作系统制作私有镜像为什么要执行 Sysprep 操作？	185
10.2.7 Windows 操作系统镜像执行 Sysprep 之后，使用该镜像创建的弹性云服务器启动失败怎么办？	186
10.2.8 通过 API 无法创建 ZVHD2 格式的镜像怎么办？	187
10.3 镜像共享类	187
10.3.1 镜像共享 FAQ	188
10.3.2 共享镜像和复制镜像的区别？	189
10.3.3 创建的镜像为什么不能共享？	190
10.4 操作系统类	190
10.4.1 如何选择操作系统？	190
10.4.2 UEFI 启动方式与 BIOS 启动方式有哪些区别？	190
10.4.3 Windows 弹性云服务器如何删除多余的网络连接？	190
10.4.4 弹性云服务器启动缓慢怎么办？	192
10.4.5 一键式重置密码插件启动失败时如何操作？	192
10.4.6 如何解决 SR-IOV 场景下 Windows 7 弹性云服务器安装 82599 网卡驱动报错？	194
10.4.7 云服务器创建或切换操作系统时，为什么选不到我的私有镜像？	195
10.5 镜像导入类	195
10.5.1 除了文档中支持的镜像格式类型，我可以其他镜像格式吗？	195
10.5.2 没有对云服务器进行相关预操作会带来什么影响？	195
10.5.3 如果操作系统类型选择错误或者系统磁盘容量填写错误怎么办？	196
10.5.4 为什么 VHD 格式的镜像上传失败，任务中心查看报错为外部镜像文件的系统盘容量大于用户设置的系统磁盘容量？	196
10.6 镜像导出类	196
10.6.1 制作的私有镜像可以下载到本地吗？支持哪些镜像格式？	196
10.6.2 为什么镜像导出到 OBS 桶后，大小和在镜像服务中显示的不一致？	197
10.6.3 公共镜像能否直接下载到本地，怎么操作？	197
10.6.4 镜像为什么没有“导出”按钮？	197
10.7 镜像优化类	197
10.7.1 一定要在云服务器中安装 Guest OS driver 吗？	197
10.7.2 Windows 操作系统为什么要安装并更新 virtio？	198
10.7.3 通过镜像文件注册私有镜像过程中，系统会对镜像做哪些修改？	198
10.7.4 创建私有镜像前云服务器或镜像文件需要完成哪些初始化配置？	199
10.7.5 Windows 外部镜像文件在导出前未完成初始化配置，怎么办？	200
10.7.6 Linux 外部镜像文件在导出前未完成初始化配置，怎么办？	203
10.7.7 如何设置镜像的网卡多队列属性？	205
10.7.8 如何配置 IPv6 地址	210

10.7.9 如何优化系统盘镜像使其支持实例快速发放?	226
10.7.10 为什么 Windows 云服务器安装 Guest OS driver 会失败?	226
10.7.11 怎样安装原生的 XEN 和 KVM 驱动.....	227
10.8 镜像复制类.....	234
10.9 镜像删除类.....	234
10.10 Cloud-Init 操作类.....	235
10.10.1 安装 Cloud-Init FAQ.....	235
10.10.2 云服务器安装 Cloud-Init 可以做什么?	239
10.10.3 安装 NetworkManager 后, 使用 Cloud-Init 注入密钥或密码失败怎么办?	239
10.10.4 SUSE 11 SP4 如何安装 growpart?	240
10.11 云服务器创建类.....	241
10.11.1 云服务器购买成功后可以换镜像吗?	241
10.11.2 使用私有镜像创建的云服务器, 是否可以与生成镜像的云服务器硬件规格不同?	241
10.11.3 使用镜像创建云服务器, 可以指定系统盘大小吗?	241
10.11.4 使用外部导入的私有镜像所创建的云服务器在启动过程中提示找不到分区, 如何处理?	241
10.11.5 注册的镜像操作系统是 CentOS 类型, 使用该镜像创建的云服务器找不到磁盘该如何处理?	244
10.11.6 如何解决注册 Windows 镜像时已勾选“进行后台自动化设置”, 但云服务器仍然无法正常启动?	245
10.11.7 使用 UEFI 启动方式的镜像创建云服务器, 云服务器启动异常怎么办?	246
10.12 驱动安装类.....	246
10.12.1 一定要在云服务器中安装 Guest OS driver 吗?	246
10.12.2 Windows 操作系统为什么要安装并更新 virtio?	246
10.12.3 为什么 Windows 云服务器安装 Guest OS driver 会失败?	247
10.12.4 Windows 系统如何安装 virtio?	247
10.12.5 Linux 系统如何安装原生的 KVM 驱动?.....	247
10.12.6 怎样安装原生的 XEN 和 KVM 驱动.....	247

1 产品介绍

1.1 什么是镜像服务

镜像与镜像服务

镜像是用于创建服务器或磁盘的模板，包含操作系统、业务数据或应用软件。

镜像服务（Image Management Service）提供镜像的生命周期管理能力。用户可以灵活地使用公共镜像、私有镜像或共享镜像申请弹性云服务器。同时，用户还能通过已有的云服务器或使用外部镜像文件创建私有镜像，实现业务上云或云上迁移。

镜像类型

镜像分为公共镜像、私有镜像、共享镜像。公共镜像为系统默认提供的镜像，私有镜像为用户自己创建的镜像，共享镜像为其他用户共享的私有镜像。

镜像类型	说明
公共镜像	常见的标准操作系统镜像，所有用户可见，包括操作系统以及预装的公共应用。公共镜像具有高度稳定性，皆为正版授权，请放心使用，您也可以根据实际需求自助配置应用环境或相关软件。
私有镜像	包含操作系统或业务数据、预装的公共应用以及用户的私有应用的镜像，仅用户个人可见。 私有镜像包括系统盘镜像、数据盘镜像、ISO 镜像和整机镜像，其中： <ul style="list-style-type: none">● 系统盘镜像：包含用户运行业务所需的操作系统、应用软件的镜像。系统盘镜像可以用于创建云服务器，迁移用户业务到云。● 数据盘镜像：只包含用户业务数据的镜像。数据盘镜像可以用于创建云硬盘，将用户的业务数据迁移到云上。● ISO 镜像：将外部镜像的ISO文件注册到云平台的私有镜像。ISO 镜像是特殊的镜像，只能发放用作临时过渡的云服务器。● 整机镜像：也叫全镜像，包含用户运行业务所需的操作系统、应用软件和业务数据的镜像。整机镜像基于增量备份制作，相比同样磁盘容量的系统盘镜像和数据盘镜像，创建效率更高。

镜像类型	说明
共享镜像	由其他用户共享而来的私有镜像。 更多关于共享镜像的使用，请参见《镜像服务用户指南》的“共享镜像”章节。

镜像服务的功能

镜像服务主要有以下功能：

- 提供常见的主流操作系统公共镜像。
- 由现有运行的云服务器，或由外部导入的方式来创建私有镜像。
- 管理公共镜像，例如：按操作系统类型/名称/ID搜索，查看镜像ID、系统盘大小等详情，查看镜像支持的特性（用户数据注入、磁盘热插拔等）。
- 管理私有镜像，例如：修改镜像属性，共享镜像，复制镜像等。
- 通过镜像创建云服务器。

访问方式

云服务平台提供了Web化的服务管理平台（即管理控制台）和基于HTTPS请求的API（Application programming interface）管理方式。

- API方式
如果用户需要将镜像服务集成到第三方系统，用于二次开发，请使用API方式访问镜像服务。具体操作请参见《镜像服务接口参考》。
- 管理控制台方式
其他相关操作，请使用管理控制台方式访问镜像服务。如果用户已在云平台注册，可直接登录管理控制台，从主页选择“镜像服务”。

1.2 相关支持列表

1.2.1 弹性云服务器类型与支持的操作系统版本

弹性云服务器类型与支持的操作系统版本如下所述。

x86 系统架构云服务器系列

- 以下云服务器支持的操作系统请参考[表1-1](#)。
通用入门型：t6
通用计算型：s2、s3、s6、sn3
通用计算增强型：c3、c6
内存优化型：m2、m3、m6
高性能计算型：hc2、h3
磁盘增强型：d2、d3、d6

- 以下云服务器支持的操作系统请参考[表1-2](#)。
通用计算增强型：c3ne
内存优化型：m3ne
- 以下云服务器支持的操作系统请参考[表1-3](#)。
超大内存型：e3
- 以下云服务器支持的操作系统请参考[表1-4](#)。
超高I/O型：i3、ir3
- 以下云服务器支持的操作系统请参考[表 OS-支持版本](#)
通用型：gs系列
通用计算增强型：gc系列
内存优化型：gm系列
- 其他GPU加速型实例参考GPU产品说明。

📖 说明

- 建议您使用操作系统官方发行版本进行系统构建，避免对发行版本进行裁剪及高度定制，以减少问题发生概率。
- 操作系统发行版本由操作系统厂商进行不定期版本更迭，部分系统版本厂商已停止维护，不再发布问题修复及安全补丁，建议及时关注厂商公告进行系统升级更新，保证您系统的健壮性。

表 1-1 OS-支持版本

OS发行系列	支持版本
Windows	Windows Server 2019 Standard/Datacenter Windows Server 2016 Standard/Datacenter Windows Server 2012 R2 Standard/Datacenter Windows Server 2012 Standard/Datacenter Windows Server 2008 R2 Standard/Enterprise/Datacenter/Web Windows Server Core Version 1709
CentOS	64bit: CentOS 8: 8.3/8.2/8.1/8.0 64bit: CentOS 7: 7.9/7.8/7.7/7.6/7.5/7.4/7.3/7.2/7.1/7.0 64bit: CentOS 6: 6.10/6.9/6.8/6.7/6.6/6.5/6.4/6.3
CentOS Stream	64bit: CentOS Stream 9 64bit: CentOS Stream 8
Ubuntu	64bit: Ubuntu 22.04/20.04/18.04/16.04/14.04/12.04 Server
EulerOS	64bit: EulerOS 2.5/2.3/2.2

OS发行系列	支持版本
Red Hat	64bit: Red Hat 8: 8.0 64bit: Red Hat 7: 7.9/7.8/7.6/7.5/7.4/7.3/7.2/7.1/7.0 64bit: Red Hat 6: 6.10/6.9/6.8/6.7/6.6/6.5/6.4
SUSE Linux Enterprise	64bit: SLES 15: 15/15 SP1/15 SP2 64bit: SLES 12: 12 SP4/12 SP3/12 SP2/12 SP1/12 64bit: SLES 11: 11 SP4/11 SP3
Debian	64bit: Debian 10: 10.0.0~10.5.0、10.7.0 64bit: Debian 9: 9.13.0/9.12.0/9.11.0/9.9.0/9.8.0/9.7.0/9.6.0/9.5.0/9.4.0/9.3.0/9.0.0 64bit: Debian 8: 8.0.0~8.10.0
openSUSE	64bit: openSUSE Leap 42: 42.3/42.2/42.1 64bit: openSUSE Leap 15: 15.1/15.0 64bit: openSUSE 13: 13.2
Fedora	64bit: Fedora 22~32
CoreOS	64bit: CoreOS 2079.4.0
FreeBSD	64bit: FreeBSD 12.1、FreeBSD 11.0、FreeBSD 10.3
openEuler	64bit: openEuler 20.03

表 1-2 OS-支持版本

OS发行版本	支持版本
Windows	Windows Server 2019 Datacenter Windows Server 2016 Standard/Datacenter Windows Server 2012 R2 Standard/Datacenter Windows Server 2008 R2 Enterprise/ Datacenter/Web/Standard Windows Server Version 1709 Datacenter
CentOS	64bit: CentOS 8 CentOS 7 CentOS 6

OS发行版本	支持版本
Ubuntu	64bit: Ubuntu 22.04 Server Ubuntu 20.04 Server Ubuntu 18.04 Server Ubuntu 16.04 Server Ubuntu 14.04 Server
EulerOS	64bit: EulerOS 2.5 EulerOS 2.3 EulerOS 2.2
Red Hat	64bit: Red Hat 7 Red Hat 6
SUSE Linux Enterprise	64bit: SLES 12 SLES 11
Debian	64bit: Debian 9 Debian 8
openSUSE	64bit: openSUSE 15.1 openSUSE 15.0
Fedora	64bit: Fedora 2x
openEuler	64bit: openEuler 20.03

表 1-3 OS-支持版本

OS发行版本	支持版本
CentOS	64bit: CentOS 8 CentOS 7 CentOS 6

OS发行版本	支持版本
Ubuntu	64bit: Ubuntu 20.04 Server Ubuntu 18.04 Server Ubuntu 16.04 Server Ubuntu 14.04 Server
EulerOS	64bit: EulerOS 2.5 EulerOS 2.3 EulerOS 2.2
Red Hat	64bit: Red Hat 7 Red Hat 6
SUSE Linux Enterprise	64bit: SLES 15 SLES 12 SLES 11
Debian	64bit: Debian 10 Debian 9 Debian 8
openSUSE	64bit: openSUSE 15.1 openSUSE 15.0
Fedora	64bit: Fedora 3x Fedora 2x
openEuler	64bit: openEuler 20.03

表 1-4 OS-支持版本

OS发行版本	支持版本
CentOS	64bit: CentOS 7

OS发行版本	支持版本
Ubuntu	64bit: Ubuntu 18.04 Server Ubuntu 16.04 Server Ubuntu 14.04 Server
EulerOS	64bit: EulerOS 2.3 EulerOS 2.2
SUSE Linux Enterprise	64bit: SLES 12
Debian 说明 ir3系列云服务器不支持 Debian操作系统。	64bit: Debian 9 Debian 8

表 1-5 OS-支持版本

OS发行系列	支持版本
OpenEuler	64bit: OpenEuler 22.03
Kylin 银河麒麟	64bit: KylinOS_V10_SP3 64bit: KylinOS_V10_SP2 64bit: KylinOS_V10_SP1
UnionTechOS 统信	64 bit: UnionTechOS Server 20 1050e
Ubuntu	Ubuntu Server 20.10 Ubuntu Server 20.04.2

鲲鹏系统架构云服务器系列

- 以下云服务器支持的操作系统请参考[表1-6](#)。
鲲鹏通用计算增强型: kc1
鲲鹏内存优化型: km1
- 鲲鹏超高I/O型云服务器: ki1系列支持CentOS 7.6。
- 鲲鹏AI推理加速型云服务器: kAi1s系列仅支持CentOS 7.6、Ubuntu 18.04 Server。

表 1-6 OS-支持版本

OS发行版本	支持版本
CentOS	64bit: CentOS 8: 8.0 64bit: CentOS 7: 7.6/7.5/7.4
Ubuntu	64bit: Ubuntu 18.04 Server
EulerOS	64bit: EulerOS 2.8
Red Hat	64bit: Red Hat 7: 7.6/7.5
SUSE Linux Enterprise	64bit: SLES 12 SP5/SP4 64bit: SLES 15
openSUSE	64bit: openSUSE Leap 15: 15.0
Fedora	64bit: Fedora 29
Debian	64bit: Debian 10.2.0
openEuler	64bit: openEuler 20.03
Kylin 银河麒麟	64bit: KylinOS_V10_SP3 64bit: KylinOS_V10_SP2 64bit: KylinOS_V10_SP1
UnionTechOS 统信	64 bit: UnionTechOS Server 20 1050e
NeoKylin 中标麒麟	64bit: NeoKylin 7.6

1.2.2 外部镜像文件支持的格式和操作系统类型

支持的文件格式

通过外部镜像文件创建私有镜像时，支持的镜像文件格式包括VMDK、VHD、QCOW2、RAW、VHDX、QED、VDI、QCOW、ISO、ZVHD2和ZVHD。请按需选择不同格式的镜像文件。

支持的操作系统类型

在管理控制台上传外部镜像文件时，系统会自动对镜像的操作系统进行识别。外部镜像文件支持的操作系统类型如表1-7和表1-8所示。

如果待注册的操作系统版本无法识别或不包含在支持列表中：

- 对于Windows操作系统，在注册镜像过程中系统会按照“Other_Windows (64_bit)”或“Other_Windows (32_bit)”类型进行处理。
- 对于Linux操作系统，在注册镜像过程中系统会按照“Other_Linux (64_bit)”或“Other_Linux (32_bit)”类型进行处理。

 说明

- 未包含在表1和表2中的操作系统类型有可能不支持镜像上传功能，建议咨询管理员确认。
- 上传CoreOS系列操作系统时，请选择操作系统类型为CoreOS，否则会被设置为Other (64bit)，同时请确保CoreOS操作系统中已安装配置coreos-cloudinit。由于系统自动升级可能会导致使用该镜像创建的云服务器无法正常使用，因此，需要关闭系统的自动更新功能。

表 1-7 外部镜像文件支持的操作系统类型（x86 架构类型）

操作系统类型	操作系统版本
Rocky Linux	Rocky Linux 9.3 64bit Rocky Linux 9.2 64bit Rocky Linux 9.1 64bit Rocky Linux 9.0 64bit Rocky Linux 8.10 64bit Rocky Linux 8.9 64bit Rocky Linux 8.8 64bit Rocky Linux 8.7 64bit Rocky Linux 8.6 64bit Rocky Linux 8.5 64bit Rocky Linux 8.4 64bit Rocky Linux 8.3 64bit
AlmaLinux	AlmaLinux 9.2 64bit AlmaLinux 9.1 64bit AlmaLinux 9.0 64bit AlmaLinux 8.9 64bit AlmaLinux 8.8 64bit AlmaLinux 8.7 64bit AlmaLinux 8.6 64bit AlmaLinux 8.5 64bit AlmaLinux 8.4 64bit AlmaLinux 8.3 64bit

操作系统类型	操作系统版本
Windows	Windows 10 64bit Windows Server 2022 Standard 64bit Windows Server 2022 Datacenter 64bit Windows Server 2019 Standard 64bit Windows Server 2019 Datacenter 64bit Windows Server 2016 Standard 64bit Windows Server 2016 Datacenter 64bit Windows Server 2012 R2 Standard 64bit Windows Server 2012 R2 Essentials 64bit Windows Server 2012 R2 Datacenter 64bit Windows Server 2012 Datacenter 64bit Windows Server 2012 Standard 64bit Windows Server 2008 WEB R2 64bit Windows Server 2008 R2 Standard 64bit Windows Server 2008 R2 Enterprise 64bit Windows Server 2008 R2 Datacenter 64bit
SUSE	SUSE Linux Enterprise Server 15 SP5 64bit SUSE Linux Enterprise Server 15 SP4 64bit SUSE Linux Enterprise Server 15 SP3 64bit SUSE Linux Enterprise Server 15 SP2 64bit SUSE Linux Enterprise Server 15 SP1 64bit SUSE Linux Enterprise Server 15 64bit SUSE Linux Enterprise Server 12 SP5 64bit SUSE Linux Enterprise Server 12 SP4 64bit SUSE Linux Enterprise Server 12 SP3 64bit SUSE Linux Enterprise Server 12 SP2 64bit SUSE Linux Enterprise Server 12 SP1 64bit SUSE Linux Enterprise Server 11 SP4 64bit SUSE Linux Enterprise Server 11 SP3 64bit SUSE Linux Enterprise Server 11 SP3 32bit

操作系统类型	操作系统版本
Oracle Linux	Oracle Linux Server release 9.4 64bit Oracle Linux Server release 9.2 64bit Oracle Linux Server release 8.10 64bit Oracle Linux Server release 8.8 64bit Oracle Linux Server release 7.6 64bit Oracle Linux Server release 7.5 64bit Oracle Linux Server release 7.4 64bit Oracle Linux Server release 7.3 64bit Oracle Linux Server release 7.2 64bit Oracle Linux Server release 7.1 64bit Oracle Linux Server release 7.0 64bit Oracle Linux Server release 6.10 64bit Oracle Linux Server release 6.9 64bit Oracle Linux Server release 6.8 64bit Oracle Linux Server release 6.7 64bit Oracle Linux Server release 6.5 64bit
Red Hat	Red Hat Linux Enterprise 8.0 64bit Red Hat Linux Enterprise 7.9 64bit Red Hat Linux Enterprise 7.8 64bit Red Hat Linux Enterprise 7.6 64bit Red Hat Linux Enterprise 7.5 64bit Red Hat Linux Enterprise 7.4 64bit Red Hat Linux Enterprise 7.3 64bit Red Hat Linux Enterprise 7.2 64bit Red Hat Linux Enterprise 7.1 64bit Red Hat Linux Enterprise 7.0 64bit Red Hat Linux Enterprise 6.10 64bit Red Hat Linux Enterprise 6.9 64bit Red Hat Linux Enterprise 6.8 64bit Red Hat Linux Enterprise 6.7 64bit Red Hat Linux Enterprise 6.6 64bit Red Hat Linux Enterprise 6.6 32bit Red Hat Linux Enterprise 6.5 64bit Red Hat Linux Enterprise 6.4 64bit Red Hat Linux Enterprise 6.4 32bit

操作系统类型	操作系统版本
Ubuntu	Ubuntu 22.04 Server 64bit Ubuntu 20.04 Server 64bit Ubuntu 19.04 Server 64bit Ubuntu 18.04.2 Server 64bit Ubuntu 18.04.1 Server 64bit Ubuntu 18.04 Server 64bit Ubuntu 16.04.6 Server 64bit Ubuntu 16.04.5 Server 64bit Ubuntu 16.04.4 Server 64bit Ubuntu 16.04 Server 64bit Ubuntu 14.04.4 Server 64bit Ubuntu 14.04.4 Server 32bit Ubuntu 14.04.3 Server 64bit Ubuntu 14.04.3 Server 32bit Ubuntu 14.04.1 Server 64bit Ubuntu 14.04.1 Server 32bit Ubuntu 14.04 Server 64bit Ubuntu 14.04 Server 32bit
openSUSE	openSUSE 42.3 64bit openSUSE 42.2 64bit openSUSE 42.1 64bit openSUSE 15.5 64bit openSUSE 15.4 64bit openSUSE 15.3 64bit openSUSE 15.1 64bit openSUSE 15.0 64bit openSUSE 13.2 64bit openSUSE 11.3 64bit

操作系统类型	操作系统版本
CentOS	CentOS 8.3 64bit
	CentOS 8.2 64bit
	CentOS 8.1 64bit
	CentOS 8.0 64bit
	CentOS 8.0 64bit
	CentOS 7.9 64bit
	CentOS 7.8 64bit
	CentOS 7.7 64bit
	CentOS 7.6 64bit
	CentOS 7.5 64bit
	CentOS 7.4 64bit
	CentOS 7.3 64bit
	CentOS 7.2 64bit
	CentOS 7.1 64bit
	CentOS 7.0 64bit
	CentOS 7.0 32bit
	CentOS 6.10 64bit
	CentOS 6.10 32bit
	CentOS 6.9 64bit
	CentOS 6.8 64bit
	CentOS 6.7 64bit
	CentOS 6.7 32bit
	CentOS 6.6 64bit
	CentOS 6.6 32bit
	CentOS 6.5 64bit
	CentOS 6.5 32bit
	CentOS 6.4 64bit
	CentOS 6.4 32bit
	CentOS 6.3 64bit
	CentOS 6.3 32bit

操作系统类型	操作系统版本
Debian	Debian GNU/Linux 12.5.0 64bit Debian GNU/Linux 12.4.0 64bit Debian GNU/Linux 12.3.0 64bit Debian GNU/Linux 12.2.0 64bit Debian GNU/Linux 12.1.0 64bit Debian GNU/Linux 12.0.0 64bit Debian GNU/Linux 11.1.0 64bit Debian GNU/Linux 10.7.0 64bit Debian GNU/Linux 10.5.0 64bit Debian GNU/Linux 10.4.0 64bit Debian GNU/Linux 10.3.0 64bit Debian GNU/Linux 10.2.0 64bit Debian GNU/Linux 10.1.0 64bit Debian GNU/Linux 10.0.0 64bit Debian GNU/Linux 9.13.0 64bit Debian GNU/Linux 9.3.0 64bit Debian GNU/Linux 9.0.0 64bit Debian GNU/Linux 8.10.0 64bit Debian GNU/Linux 8.8.0 64bit Debian GNU/Linux 8.7.0 64bit Debian GNU/Linux 8.6.0 64bit Debian GNU/Linux 8.5.0 64bit Debian GNU/Linux 8.4.0 64bit Debian GNU/Linux 8.2.0 64bit Debian GNU/Linux 8.1.0 64bit
Fedora	Fedora 32 64bit Fedora 31 64bit Fedora 30 64bit Fedora 29 64bit Fedora 28 64bit Fedora 27 64bit Fedora 26 64bit Fedora 25 64bit Fedora 24 64bit Fedora 23 64bit Fedora 22 64bit

操作系统类型	操作系统版本
EulerOS	EulerOS 2.10 64bit EulerOS 2.9 64bit EulerOS 2.5 64bit EulerOS 2.3 64bit EulerOS 2.2 64bit EulerOS 2.1 64bit
CoreOS	CoreOS 1800.1.0 CoreOS 1745.2.0 CoreOS 1632.0.0 CoreOS 1520.8.0 CoreOS 1465.8.0 CoreOS 1298.5.0 CoreOS 1122.3.0 CoreOS 1122.2.0 CoreOS 1185.5.0 CoreOS 1068.10.0 CoreOS 1010.5.0 CoreOS 1298.6.0
openEuler	openEuler 24.03 64bit openEuler 22.03 64bit openEuler 20.03 64bit
中标麒麟	NeoKylin 7.6 64bit NeoKylin 7.4 64bit NeoKylin Server release 5.0 U2 64bit NeoKylin Linux Advanced Server release 7.0 U5 64bit

表 1-8 外部镜像文件支持的操作系统类型（ARM 架构类型）

操作系统类型	操作系统版本
AlmaLinux	AlmaLinux 8.4 64bit AlmaLinux 8.3 64bit
CentOS	CentOS 8.0 64bit CentOS 7.6 64bit CentOS 7.5 64bit CentOS 7.4 64bit
Debian	Debian GNU/Linux 10.2.0 64bit

操作系统类型	操作系统版本
EulerOS	EulerOS 2.10 64bit EulerOS 2.9 64bit EulerOS 2.8 64bit
Fedora	Fedora 29 64bit
Ubuntu	Ubuntu 20.04 Server 64bit Ubuntu 19.04 Server 64bit Ubuntu 18.04 Server 64bit
SUSE	SUSE Linux Enterprise Server 12 SP5 64bit
openEuler	openEuler 24.03 64bit openEuler 22.03 64bit openEuler 20.03 64bit
openSUSE	openSUSE 15.0 64bit
中标麒麟	NeoKylin V7 64bit NeoKylin 7.7 64bit
统信	UnionTechOS V20 1070e UnionTechOS V20 1060e UnionTechOS V20 1050e UnionTechOS 20 64bit
银河麒麟	Kylin V10 SP3 64bit Kylin V10 SP2 64bit Kylin V10 SP1 64bit Kylin V10 64bit Kylin Desktop V10 64bit
麒麟信安	KylinSec 3.3 64bit
普华	iSoft 5.1 64bit

相关操作

如何上传外部镜像文件，请参见《镜像服务快速入门》的“上传镜像文件（Windows）”和“上传镜像文件（Linux）”章节。

外部镜像文件上传成功后，您可以通过注册镜像，将外部镜像文件注册为云平台的私有镜像。具体操作，请参考《镜像服务快速入门》的“注册镜像（Windows）”和“注册镜像（Linux）”章节。

1.2.3 支持 UEFI 启动方式的操作系统版本

云服务器的启动方式包括BIOS启动和UEFI启动，二者区别请参见《镜像服务用户指南》常见问题中的“UEFI启动方式与BIOS启动方式有哪些区别？”章节。

支持UEFI启动方式的操作系统版本如表1-9所示。

表 1-9 支持 UEFI 启动方式的操作系统版本

操作系统类型	操作系统版本
Windows	Windows Server 2019 Standard 64bit Windows Server 2019 Datacenter 64bit Windows Server 2016 Standard 64bit Windows Server 2016 Datacenter 64bit Windows Server 2012 R2 Standard 64bit Windows Server 2012 R2 Datacenter 64bit Windows Server 2012 Essentials R2 64bit Windows Server 2012 Standard 64bit Windows Server 2012 Datacenter 64bit Windows 10 64bit
Ubuntu	Ubuntu 19.04 Server 64bit Ubuntu 18.04 Server 64bit Ubuntu 16.04 Server 64bit Ubuntu 14.04 Server 64bit
Red Hat	Red Hat Linux Enterprise 7.4 64bit Red Hat Linux Enterprise 7.3 64bit Red Hat Linux Enterprise 7.1 64bit Red Hat Linux Enterprise 7.0 64bit Red Hat Linux Enterprise 6.9 64bit Red Hat Linux Enterprise 6.6 32bit Red Hat Linux Enterprise 6.5 64bit
Oracle Linux	Oracle Linux Server release 7.4 64bit Oracle Linux Server release 6.9 64bit
openSUSE	openSUSE 42.1 64bit
SUSE	SUSE Linux Enterprise Server 12 SP5 64bit SUSE Linux Enterprise Server 12 SP1 64bit SUSE Linux Enterprise Server 11 SP3 64bit
Fedora	Fedora 29 64bit Fedora 24 64bit
Debian	Debian GNU/Linux 8.8.0 64bit

操作系统类型	操作系统版本
CentOS	CentOS 7.6 64bit CentOS 7.5 64bit CentOS 7.4 64bit CentOS 7.0 64bit CentOS 6.9 64bit CentOS 6.6 64bit
EulerOS	EulerOS 2.8 64bit EulerOS 2.7 64bit EulerOS 2.5 64bit EulerOS 2.3 64bit EulerOS 2.2 64bit
openEuler	openEuler 20.03 64bit

1.3 IMS 权限管理

如果您需要对创建的镜像服务（Image Management Service）资源，给企业中的员工设置不同的访问权限，以达到不同员工之间的权限隔离，您可以使用统一身份认证服务（Identity and Access Management，简称IAM）进行精细的权限管理。该服务提供用户身份认证、权限分配、访问控制等功能，可以帮助您安全地控制资源的访问。

通过IAM，您可以在账号中给员工创建IAM用户，并使用策略来控制IAM用户对资源的访问范围。例如您的员工中有负责软件开发的人员，您希望开发人员拥有镜像服务（IMS）的使用权限，但是不希望开发人员拥有删除镜像等高危操作的权限，那么您可以使用IAM为开发人员创建用户，通过授予仅能使用IMS，但是不允许删除IMS的权限策略，控制开发人员对IMS资源的使用范围。

如果账号已经能满足您的要求，不需要创建独立的IAM用户进行权限管理，您可以跳过本章节，不影响您使用IMS服务的其它功能。

IAM是提供权限管理的基础服务，无需付费即可使用，您只需要为您账号中的资源进行付费。关于IAM的详细介绍，请参见：《IAM产品介绍》。

IMS 权限

默认情况下，管理员创建的IAM用户没有任何权限，需要将其加入用户组，并给用户组授予策略或角色，才能使得用户组中的用户获得对应的权限，这一过程称为授权。授权后，用户就可以基于被授予的权限对云服务进行操作。

IMS部署时通过物理区域划分，为项目级服务。授权时，“作用范围”需要选择“区域级项目”，然后在指定区域对应的项目中设置相关权限，并且该权限仅对此项目生效；如果在“所有项目”中设置权限，则该权限在所有区域下的项目中都生效。**访问IMS时，需要先切换至授权区域。**

根据授权精度分为角色和策略。

- **角色**：IAM最初提供的一种根据用户的工作职能定义权限的**粗粒度**授权机制。该机制以服务为粒度，提供有限的服务相关角色用于授权。由于各服务之间存在业务依赖关系，因此给用户授予角色时，可能需要一并授予依赖的其他角色，才能正确完成业务。角色并不能满足用户对精细化授权的要求，无法完全达到企业对权限最小化的安全管控要求。

表 1-10 IMS 系统角色

系统角色	描述	依赖关系
IMS Administrator	镜像服务的管理员权限。	该角色有依赖，需要勾选依赖的角色：Tenant Administrator。
Server Administrator	拥有该权限的用户可以创建、删除、查询、修改及上传镜像。	该角色有依赖，需要在同一项目中勾选依赖的角色：IMS Administrator。

- **策略（推荐）**：IAM最新提供的一种**细粒度**授权的能力，可以精确到具体服务的操作、资源以及请求条件等。基于策略的授权是一种更加灵活的授权方式，能够满足企业对权限最小化的安全管控要求。例如：针对IMS服务，管理员能够控制IAM用户仅能对某一类镜像资源进行指定的管理操作。

多数策略以API接口为粒度进行权限拆分，IMS支持的API授权项请参见：《镜像服务API参考》中“权限及授权项说明”章节。

表 1-11 IMS 系统策略

策略名称	描述	依赖关系
IMS FullAccess	镜像服务所有权限。	无
IMS ReadOnlyAccess	镜像服务只读权限，拥有该权限的用户仅能查看镜像服务数据。	无

表1-12列出了镜像服务（IMS）常用操作与系统权限的授权关系，您可以参照该表选择合适的系统权限。

表 1-12 常用操作与系统权限的关系

操作	IMS FullAccess	IMS ReadOnlyAccess	IMS Administrator (需依赖Tenant Administrator)
创建镜像	√	x	√
删除镜像	√	x	√
查询镜像	√	√	√
更新镜像信息	√	x	√

1.4 基本概念

1.4.1 区域和可用区

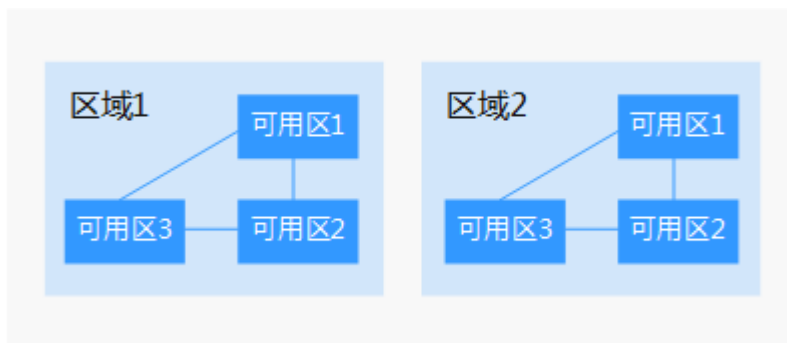
什么是区域、可用区？

区域和可用区用来描述数据中心的位置，您可以在特定的区域、可用区创建资源。

- 区域（Region）指物理的数据中心。每个区域完全独立，这样可以实现最大程度的容错能力和稳定性。资源创建成功后不能更换区域。
- 可用区（AZ, Availability Zone）是同一区域内，电力和网络互相隔离的物理区域，一个可用区不受其他可用区故障的影响。一个区域内可以有多个可用区，不同可用区之间物理隔离，但内网互通，既保障了可用区的独立性，又提供了低价、低时延的网络连接。

图1-1阐明了区域和可用区之间的关系。

图 1-1 区域和可用区



如何选择区域？

建议就近选择靠近您或者您的目标用户的区域，这样可以减少网络时延，提高访问速度。

如何选择可用区？

是否将资源放在同一可用区内，主要取决于您对容灾能力和网络时延的要求。

- 如果您的应用需要较高的容灾能力，建议您将资源部署在同一区域的不同可用区内。
- 如果您的应用要求实例之间的网络延时较低，则建议您将资源创建在同一可用区内。

区域和终端节点

当您通过API使用资源时，您必须指定其区域终端节点。有关的区域和终端节点的更多信息，请参阅[地区和终端节点](#)。

1.4.2 镜像常见格式

镜像服务目前支持多种格式，而在镜像服务内部统一使用镜像服务自研格式ZVHD或ZVHD2。

常见镜像格式说明请参考[表1-13](#)。

表 1-13 镜像常见格式说明

镜像格式	介绍	备注
ZVHD	采用ZLIB压缩算法，支持顺序读写。	镜像服务底层通用格式。镜像服务导入和导出支持格式。 说明 zvhd镜像文件格式不支持延迟加载特性，大文件在导入时依赖延迟加载特性，建议您使用zvhd2文件格式导入镜像文件。
ZVHD2	采用ZSTD压缩算法，支持延迟加载。	镜像服务延迟加载特性专用格式。镜像服务导入支持格式。
QCOW2	QCOW2格式镜像是QEMU模拟器支持的一种磁盘镜像，是用一个文件的形式来表示一块固定大小的块设备磁盘。与普通的RAW格式镜像相比，QCOW2格式有如下几个特性： <ul style="list-style-type: none"> • 支持更小的磁盘占用。 • 支持写时拷贝（CoW, Copy-On-Write），镜像文件只反映底层磁盘变化。 • 支持快照，可以包含多个历史快照。 • 支持压缩和加密，可以选择ZLIB压缩和AES加密。 	镜像服务导入和导出支持格式。
VMDK	VMDK是VMware创建的虚拟硬盘格式。一个VMDK文件代表VMFS（云服务器文件系统）在云服务器上的一个物理硬盘驱动。	镜像服务导入和导出支持格式。
VHD	VHD是微软提供的一种虚拟硬盘文件格式。VHD文件格式可以被压缩成单个文件存放到宿主机的文件系统上，主要包括云服务器启动所需的文件系统。	镜像服务导入和导出支持格式。 说明 vhd镜像文件格式不支持延迟加载特性，大文件在导入时依赖延迟加载特性，建议您使用zvhd2文件格式导入镜像文件。

镜像格式	介绍	备注
VHDX	微软在Windows Server 2012中的Hyper-V引入的一个新版本的VHD格式，称为VHDX。与VHD格式相比，VHDX具有更大的存储容量。它在电源故障期间提供数据损坏保护，并且优化了磁盘结构对齐方式，以防止新的大扇区物理磁盘性能降级。	镜像服务导入支持格式。
RAW	RAW格式是直接给云服务器进行读写的文件。RAW不支持动态增长空间，是镜像中I/O性能较好的一种格式。	镜像服务导入支持格式。
QCOW	QCOW通过二级索引表来管理整个镜像的空间分配，其中第二级的索引用了内存CACHE技术，需要查找动作，这方面导致性能的损失。QCOW优化性能低于QCOW2，读写性能低于RAW。	镜像服务导入支持格式。
VDI	VDI是Oracle公司Virtual BOX虚拟化软件所用的硬盘镜像文件格式，支持快照。	镜像服务导入支持格式。
QED	QED格式是QCOW2格式的一种改进，存储定位查询方式和数据块大小与QCOW2一样。但在实现CoW（Copy-On-Write）的机制时，QED将QCOW2的引用计数表用了一个重写标记（Dirty Flag）来替代。	镜像服务导入支持格式。

1.5 镜像服务与其他服务的关系

表 1-14 镜像服务与其他服务的关系

服务名称	镜像服务与其他服务的关系	主要交互功能
弹性云服务器（Elastic Cloud Server, ECS）	通过镜像创建云服务器，或者将云服务器转换为镜像	<ul style="list-style-type: none"> 通过镜像创建云服务器 通过云服务器创建Windows系统盘镜像 通过云服务器创建Linux系统盘镜像
对象存储服务（Object Storage Service, OBS）	镜像保存在对象存储中，上传外部镜像文件或者导出私有镜像时均通过对象存储服务的OBS桶来存储	导出镜像

服务名称	镜像服务与其他服务的关系	主要交互功能
云硬盘 (Elastic Volume Service, EVS)	可以通过云服务器上挂载的数据盘创建数据盘镜像，数据盘镜像可用来创建新的云硬盘	通过云服务器的数据盘创建数据盘镜像
云备份 (Cloud Backup and Recovery, CBR)	使用已有的云备份制作整机镜像	通过云备份创建整机镜像

2 通过 IAM 授予使用 IMS 的权限

2.1 创建用户并授权使用 IMS

操作场景

如果需要对您所拥有的IMS资源进行精细的权限管理，可以使用统一身份认证服务（Identity and Access Management，简称IAM）。通过IAM，您可以：

- 根据企业的业务组织，在您的账号中，给企业中不同职能部门的员工创建IAM用户，让员工拥有唯一安全凭证，并使用IMS资源。
- 根据企业用户的职能，设置不同的访问权限，以达到用户之间的权限隔离。
- 将IMS资源委托给更专业、高效的其他账号或者云服务，这些账号或者云服务可以根据权限进行代运维。

如果账号已经能满足您的要求，不需要创建独立的IAM用户，您可以跳过本章节，不影响您使用IMS的其它功能。

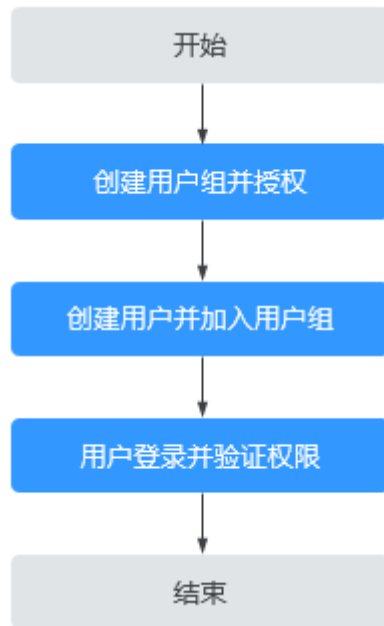
本章节以授予“IMS ReadOnlyAccess”权限为例介绍为用户授权的方法，操作流程如图2-1所示。

前提条件

给用户组授权之前，请您了解用户组可以添加的IMS权限，并结合实际需求进行选择，IMS支持的系统权限，请参见：[IMS权限](#)。

示例流程

图 2-1 给用户授予 IMS 权限流程



1. 创建用户组并授权
在IAM控制台创建用户组，并在“操作”列下选择“授权”，授予镜像服务只读权限“IMS ReadOnlyAccess”。
2. 创建用户并加入用户组
在IAM控制台创建用户，并在“操作”列下选择“授权”，将其加入1中创建的用户组。
3. 用户登录并验证权限
新建的用户登录控制台，切换至授权区域，验证权限（假设当前权限仅包含IMS ReadOnlyAccess）。
 - 在“服务列表”中选择镜像服务，进入IMS主界面，执行除查询以外的其他操作，如：创建、修改、删除等。
示例：单击右上角“创建私有镜像”，若提示权限不足，表示“IMS ReadOnlyAccess”已生效。
 - 在“服务列表”中选择除镜像服务外的任意一个服务，如“虚拟私有云”，若提示权限不足，表示“IMS ReadOnlyAccess”已生效。

2.2 IMS 自定义策略

操作场景

如果系统预置的IMS权限不满足您的授权要求，可以创建自定义策略。自定义策略中可以添加的授权项（Action）请参考：《镜像服务API参考》的“权限及授权项说明”章节。

目前支持以下两种方式创建自定义策略：

- 可视化视图创建自定义策略：无需了解策略语法，按可视化视图导航栏选择云服务、操作、资源、条件等策略内容，可自动生成策略。
- JSON视图创建自定义策略：可以在选择策略模板后，根据具体需求编辑策略内容；也可以直接在编辑框内编写JSON格式的策略内容。

具体创建步骤请参见《统一身份认证服务用户指南（旧版）》的“创建自定义策略”章节。本章为您介绍常用的IMS自定义策略样例。

策略样例

- 示例1：授权用户制作镜像

```
{
  "Version": "1.1",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ims:serverImages:create"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "KMS:*:*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "ecs:cloudServers:get",
        "ecs:servers:get",
        "ecs:serverVolumes:use",
        "ecs:cloudServers:list",
        "ecs:serverVolumeAttachments:list",
        "ecs:servers:list"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "bms:servers:list",
        "bms:servers:get",
        "bms:serverFlavors:get"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "evs:volumes:*"
      ]
    }
  ]
}

{
  "Version": "1.1",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "OBS:*:*"
      ]
    }
  ]
}
```

📖 说明

制作镜像对应的授权项为：ims:serverImages:create，其他授权项均为制作镜像所依赖的授权项。

- 示例2：拒绝用户删除镜像

拒绝策略需要同时配合其他策略使用，否则没有实际作用。用户被授予的策略中，一个授权项的作用如果同时存在Allow和Deny，则遵循**Deny优先原则**。

如果您给用户授予IMS FullAccess的系统策略，但不希望用户拥有IMS FullAccess中定义的删除镜像权限，您可以创建一条拒绝删除镜像的自定义策略，然后同时将IMS FullAccess和拒绝策略授予用户，根据Deny优先原则，则用户可以对IMS执行除了删除镜像外的所有操作。拒绝策略示例如下：

```
{
  "Version": "1.1",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": [
        "ims:images:delete"
      ]
    }
  ]
}
```

3 创建私有镜像

3.1 创建方式导航

私有镜像是包含操作系统、预装的公共应用以及用户私有应用的镜像，仅用户个人可见。私有镜像包括系统盘镜像、数据盘镜像和整机镜像，由现有运行的云服务器创建而来，或由外部导入而来。

创建镜像不会影响云服务器上当前业务的运行，不会造成数据丢失，请放心使用。

本章介绍以下创建私有镜像的方法：

- [通过云服务器创建Windows系统盘镜像](#)
- [通过云服务器创建Linux系统盘镜像](#)
- [通过外部镜像文件创建Windows系统盘镜像](#)
- [通过外部镜像文件创建Linux系统盘镜像](#)
- [通过数据盘创建数据盘镜像](#)
- [通过外部镜像文件创建数据盘镜像](#)
- [通过云服务器创建整机镜像](#)
- [通过云服务器备份创建整机镜像](#)
- [通过云备份创建整机镜像](#)
- [通过ISO文件创建Windows系统盘镜像](#)
- [通过ISO文件创建Linux系统盘镜像](#)

系统盘镜像创建成功后，您可以使用镜像创建云服务器或者为已有云服务器切换操作系统。

3.2 通过云服务器创建 Windows 系统盘镜像

操作场景

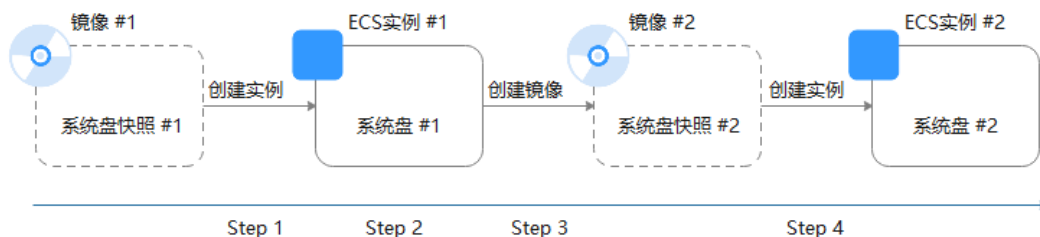
如果您已经创建了一台Windows云服务器，并根据业务需要进行了自定义配置（如安装软件、部署应用环境等），您可以为更新后的云服务器创建系统盘镜像。使用该镜像创建新的云服务器，会包含您已配置的自定义项，省去您重复配置的时间。

创建系统盘镜像不会影响云服务器上当前业务的运行，不会造成数据丢失，请放心使用。

背景知识

通过云服务器创建系统盘镜像的过程如下：从现有镜像开始，创建一个云服务器实例，自定义该实例，从该实例创建新的镜像，并最终创建新的实例，如图3-1所示。

图 3-1 系统盘镜像使用流程



- 创建系统盘镜像常见于应用扩容场景。也可用于混合云部署场景，为了实现云上及线下资源同步，可以借助镜像导入导出功能，实现过程如下：

a. 基于云服务器制作系统盘镜像

📖 说明

此处的云服务器必须为私有镜像所创建，否则基于该云服务器制作的系统盘镜像不支持导出。

b. 将镜像导出到OBS桶，详情可参阅[导出镜像](#)。

c. 下载OBS桶中的镜像文件

- 无需关闭云服务器即可创建镜像。

创建过程中，对云服务器正在运行的业务没有影响。

创建过程中，请勿改变云服务器状态，不要关闭、开启或者重启云服务器，避免创建失败。

- 镜像创建时间取决于云服务器系统盘的大小，也与网络状态、并发任务数有关。
- 云服务器及通过云服务器创建的系统盘镜像属于同一个区域。
- 如果云服务器已到期或释放，使用您提前创建的系统盘镜像创建新的云服务器，可找回原云服务器中的数据。

前提条件

创建私有镜像前，请您务必执行以下操作：

- 请将云服务器中的敏感数据删除后再创建私有镜像，避免数据安全隐患。
- 确保云服务器处于运行中或关机状态。
- 检查云服务器的网络配置，确保网卡属性为DHCP方式，按需开启远程桌面连接功能。详情请参见[设置网卡属性为DHCP（Windows）](#)和[开启远程桌面连接功能](#)。
- 检查云服务器中是否已安装一键式重置密码插件，保证镜像创建的新云服务器可以使用控制台的“重置密码”功能进行密码重置。详情请参见[安装一键式重置密码插件（Windows）](#)。

- 检查云服务器中是否已安装Cloudbase-Init工具，保证镜像创建的新云服务器可以使用控制台的“用户数据注入”功能注入初始化自定义信息（例如为云服务器设置登录密码）。详情请参见[安装并配置Cloudbase-Init工具](#)。
- 检查并安装PV driver和virtio驱动，确保镜像创建的新云服务器同时支持KVM虚拟化和XEN虚拟化，并且可以提升云服务器网络性能。
详细操作请参见[优化过程（Windows）](#)中的步骤2~步骤5。
- 执行Sysprep操作，确保镜像创建的新云服务器加入域后SID唯一。对于集群部署场景，SID需要保持唯一。详情请参见[执行Sysprep](#)。

📖 说明

如果待创建私有镜像的云服务器使用的是公共镜像，那么默认已安装一键式重置密码插件和Cloudbase-Init工具，指导中均提供了验证是否安装的方法，您可以参考相应内容确认。

如果待创建私有镜像的云服务器使用的是公共镜像，那么默认已安装Cloudbase-Init工具，指导中提供了验证是否安装的方法，您可以参考相应内容确认。

操作步骤

步骤1 登录IMS控制台。

1. 登录管理控制台。
2. 选择“计算 > 镜像服务”。
进入镜像服务页面。

步骤2 创建系统盘镜像。

1. 单击右上角的“创建私有镜像”，进入创建私有镜像页面。
2. 根据界面要求填写如下信息：
包含“镜像类型和来源”和“配置信息”两个信息块，各参数说明参见[表3-1](#)和[表3-2](#)。

表 3-1 镜像类型和来源

参数	说明
创建方式	选择“系统盘镜像”。
选择镜像源	选择“云服务器”，然后从列表中选择已完成相关配置的云服务器。

表 3-2 配置信息

参数	说明
名称	设置一个便于您识别的镜像名称。
描述	可选参数，对镜像进行描述。

📖 说明

以上为全量参数，不同企业、单位支持情况不同，以实际界面为准。

3. 单击“立即创建”。
4. 确认镜像参数，单击“提交申请”。

步骤3 返回私有镜像列表，查看创建的系统盘镜像。

镜像创建时间取决于云服务器系统盘大小，也与网络状态、并发任务数有关，请耐心等待。当镜像的状态为“正常”时，表示创建完成。

📖 说明

- 在创建镜像过程中，请勿对所选择的云服务器及其相关联资源进行其他操作。

---结束

后续操作

系统盘镜像创建成功后，您可以通过如下2种方式实现两台服务器之间的数据迁移：

- 使用该镜像创建云服务器，详见[通过镜像创建云服务器](#)
- 为已有云服务器切换操作系统

3.3 通过云服务器创建 Linux 系统盘镜像

操作场景

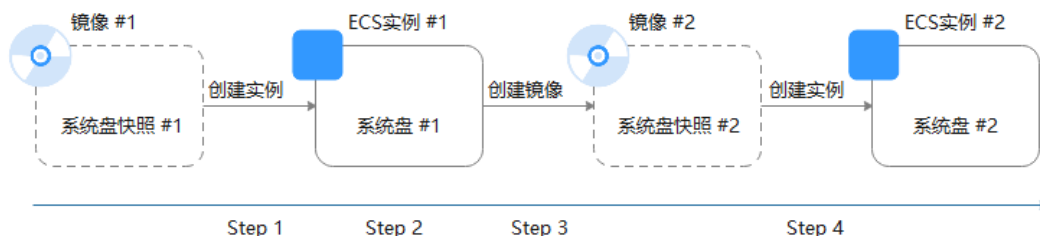
如果您已经创建了一台Linux云服务器，并根据业务需要进行了自定义配置（如安装软件、部署应用环境等），您可以为更新后的云服务器创建系统盘镜像。使用该镜像创建新的云服务器，会包含您已配置的自定义项，省去您重复配置的时间。

创建系统盘镜像不会影响云服务器上当前业务的运行，不会造成数据丢失，请放心使用。

背景知识

通过云服务器创建系统盘镜像的过程如下：从现有镜像开始，创建一个云服务器实例，自定义该实例，从该实例创建新的镜像，并最终创建新的实例，如图3-1所示。

图 3-2 系统盘镜像使用流程



- 创建系统盘镜像常见于应用扩容场景。也可用于混合云部署场景，为了实现云上及线下资源同步，可以借助镜像导入导出功能，实现过程如下：
 - a. 基于云服务器制作系统盘镜像

📖 说明

基于如下镜像类型创建的云服务器所制作的系统盘镜像不支持导出：

- ISO镜像
 - SUSE、Red Hat、Ubuntu、Oracle Linux公共镜像所创建的私有镜像
- b. 将镜像导出到OBS桶，详情可参阅[导出镜像](#)。
- c. 下载OBS桶中的镜像文件
- 无需关闭云服务器即可创建镜像。
创建过程中，对云服务器正在运行的业务没有影响。
创建过程中，请勿改变云服务器状态，不要关闭、开启或者重启云服务器，避免创建失败。
 - 镜像创建时间取决于云服务器系统盘的大小，也与网络状态、并发任务数有关。
 - 云服务器及通过云服务器创建的系统盘镜像属于同一个区域。
 - 如果云服务器已到期或释放，使用您提前创建的系统盘镜像创建新的云服务器，可找回原云服务器中的数据。

前提条件

创建私有镜像前，请您务必执行以下操作：

- 请将云服务器中的敏感数据删除后再创建私有镜像，避免数据安全隐患。
- 确保云服务器处于运行中或关机状态。
- 检查云服务器的网络配置，确保网卡属性为DHCP方式。详情请参见[设置网卡属性为DHCP \(Linux\)](#)。
- 有些云服务器正常运行或者高级功能依赖某些驱动，例如：H2型云服务器依赖InfiniBand网卡驱动。因此，需要提前安装特殊驱动。详情请参见[安装Linux特殊驱动](#)。
- 检查云服务器中是否已安装一键式重置密码插件，保证镜像创建的新云服务器可以使用控制台的“重置密码”功能进行密码重置。详情请参见[安装一键式重置密码插件 \(Linux\)](#)。
- 检查云服务器中是否已安装Cloud-Init工具，保证镜像创建的新云服务器可以使用控制台的“用户数据注入”功能注入初始化自定义信息（例如为云服务器设置登录密码）。详情请参见[安装Cloud-Init工具](#)和[配置Cloud-Init工具](#)。
- 清理网络规则文件，避免镜像创建的新云服务器发生网卡名称漂移。详情请参见[清理网络规则文件](#)。
- 为了确保镜像创建的新云服务器同时支持XEN虚拟化和KVM虚拟化，请优化Linux云服务器，包括修改“grub”、“fstab”文件的磁盘标识方式为UUID、安装原生的XEN和KVM驱动等操作。
详细操作请参考[优化过程 \(Linux\)](#) 中的步骤2~步骤6。
- 如果云服务器挂载了多个数据盘，可能导致由私有镜像创建的新云服务器无法使用。因此在创建私有镜像前，需要卸载原云服务器中挂载的所有数据盘。详情请参见[卸载云服务器的数据盘](#)。
- 如果云服务器挂载了数据盘，并在初始化时设置了开机自动挂载磁盘分区，在创建私有镜像前，需要删除fstab文件中的开机自动挂载磁盘分区的配置。

📖 说明

如果待创建私有镜像的云服务器使用的是公共镜像，那么默认已安装一键式重置密码插件和Cloud-Init工具，指导中均提供了验证是否安装的方法，您可以参考相应内容确认。

如果待创建私有镜像的云服务器使用的是公共镜像，那么默认已安装Cloud-Init工具，指导中提供了验证是否安装的方法，您可以参考相应内容确认。

操作步骤

步骤1 登录IMS控制台。

1. 登录管理控制台。
 2. 选择“计算 > 镜像服务”。
- 进入镜像服务页面。

步骤2 创建系统盘镜像。

1. 单击右上角的“创建私有镜像”，进入创建私有镜像页面。
2. 根据界面要求填写如下信息：
包含“镜像类型和来源”和“配置信息”两个信息块，各参数说明参见[表3-3](#)和[表3-4](#)。

表 3-3 镜像类型和来源

参数	说明
创建方式	选择“系统盘镜像”。
选择镜像源	选择“云服务器”，然后从列表中选择已完成相关配置的云服务器。

表 3-4 配置信息

参数	说明
名称	设置一个便于您识别的镜像名称。
描述	可选参数，对镜像进行描述。

📖 说明

以上为全量参数，不同企业、单位支持情况不同，以实际界面为准。

3. 单击“立即创建”。
4. 确认镜像参数，单击“提交申请”。

步骤3 返回私有镜像列表，查看创建的系统盘镜像。

镜像创建时间取决于云服务器系统盘大小，也与网络状态、并发任务数有关，请耐心等待。当镜像的状态为“正常”时，表示创建完成。

📖 说明

- 在创建镜像过程中，请勿对所选择的云服务器及其相关联资源进行其他操作。

---结束

后续操作

系统盘镜像创建成功后，您可以通过如下2种方式实现两台服务器之间的数据迁移：

- 使用该镜像创建云服务器，详见[通过镜像创建云服务器](#)
- 为已有云服务器切换操作系统

3.4 通过外部镜像文件创建 Windows 系统盘镜像

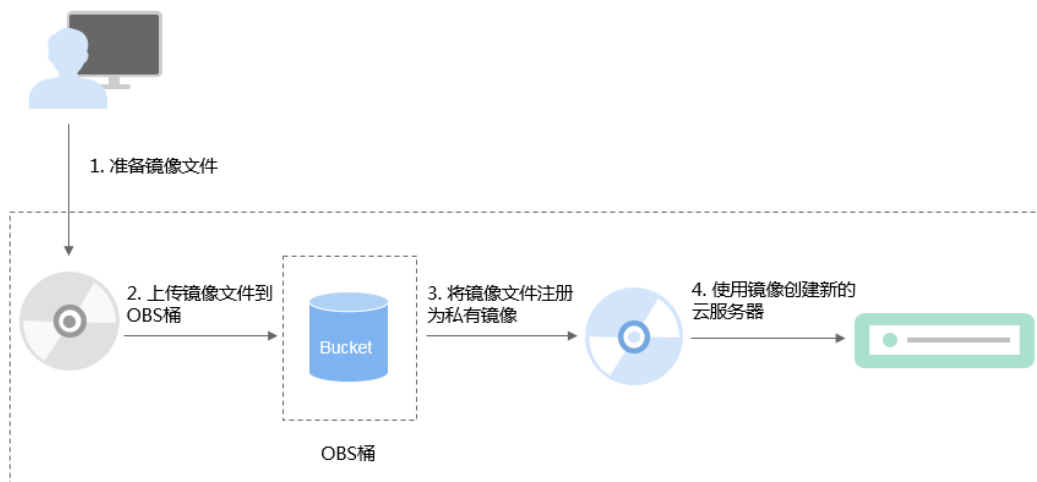
3.4.1 流程概览（Windows）

除了可以通过云服务器创建私有镜像，系统也支持外部镜像导入功能，可将您本地或者其他云平台的服务器系统盘镜像文件导入至镜像服务私有镜像中。导入后，您可以使用该镜像创建新的云服务器，或对已有云服务器的系统进行重装。

创建过程

私有镜像创建过程如[图3-3](#)所示。

图 3-3 Windows 系统盘镜像创建过程



步骤说明如下：

1. 准备符合平台要求的外部镜像文件，请参考[准备镜像文件（Windows）](#)。
2. 上传外部镜像文件到OBS个人桶中，请参考[上传镜像文件（Windows）](#)。
3. 通过管理控制台选择上传的镜像文件，并将镜像文件注册为私有镜像，请参考[注册镜像（Windows）](#)。
4. 私有镜像注册成功后，使用该镜像创建新的云服务器，请参考[使用镜像创建弹性云服务器（Windows）](#)。

3.4.2 准备镜像文件（Windows）

您需要提前准备好符合条件的镜像文件。

说明

大文件导入（不超过1TB）目前仅支持RAW和ZVHD2格式，RAW镜像文件除了要满足表3-6的要求外，还需要生成位表文件，同镜像文件一并上传。详细操作请参见[快速导入镜像文件](#)。

外部镜像文件导出前的初始化配置

以下操作配置需要在虚拟机内部完成，强烈建议您在原平台的虚拟机实施修改后，再导出镜像文件。如果您在导出镜像文件前未完成以下配置，推荐您使用云服务器完成这些配置，具体操作请参见[Windows外部镜像文件在导出前未完成初始化配置，怎么办？](#)。

表 3-5 外部镜像文件导出前的初始化配置

配置分类	配置项
网络能力	<p>必选项，不设置会导致云服务器启动异常或网络能力异常，包括：</p> <p>设置网卡属性为DHCP</p> <p>可选项，即增值能力，主要包括：</p> <ul style="list-style-type: none"> 开启网卡多队列 开启网卡多队列功能可以将网卡中断分散给不同的CPU处理，实现负载均衡，从而提升网络PPS和带宽性能。操作方法请参考如何设置镜像的网卡多队列属性？。 配置动态获取IPv6地址 IPv6的使用，可以有效弥补IPv4网络地址资源有限的问题。镜像中配置动态获取IPv6地址，发放的云服务器能够同时支持IPv4和IPv6地址。配置方法请参考如何配置IPv6地址。
工具	<p>强烈建议安装Cloudbase-Init工具。</p> <p>Cloudbase-Init是开源的云初始化工具，使用安装了Cloudbase-Init的镜像创建云服务器时可以通过“用户数据注入”功能，注入初始化自定义信息（例如为云服务器设置登录密码）；还可以通过查询、使用元数据，对正在运行的云服务器进行配置和管理。不安装Cloudbase-Init工具，将无法对云服务器进行自定义配置，只能使用镜像原有密码登录云服务器。</p> <p>安装方法请参考安装并配置Cloudbase-Init工具。</p> <p>在Cloudbase-Init安装完成后，请确认云服务器是否需要加入域，或是否需要保证SID唯一。如果是，请参考执行Sysprep确保在云服务器加入域后SID唯一。</p>
插件	<p>为了保证使用私有镜像创建的新云服务器可以实现一键式重置密码功能，建议您在创建私有镜像前安装密码重置插件CloudResetPwdAgent。</p> <p>详情请参见安装一键式重置密码插件（Windows）。</p>

配置分类	配置项
驱动	<p>云服务器的正常运行依赖于XEN Guest OS driver (PV driver) 和KVM Guest OS driver (virtio驱动)，为了同时支持XEN虚拟化和KVM虚拟化，以及提升云服务器网络性能，需要确保镜像安装了PV driver和virtio驱动。</p> <ul style="list-style-type: none"> • 安装PV driver • 安装virtio驱动

镜像文件属性限制

表 3-6 Windows 操作系统的镜像文件属性限制

镜像文件属性	条件
操作系统	<ul style="list-style-type: none"> • Windows Server 2008相关版本、Windows Server 2012相关版本、Windows Server 2016相关版本 • 支持32位和64位 • 操作系统不能与特定的硬件绑定 • 操作系统必须支持全虚拟化 <p>所支持的操作系统版本请参考外部镜像文件支持的格式和操作系统类型，在此范围内的操作系统支持后台自动化配置（详情请参阅通过镜像文件注册私有镜像过程中，系统会对镜像做哪些修改？），在此之外的操作系统请您自行排查及安装Guest OS driver驱动，在注册镜像页面选择Other Windows，导入后系统启动情况取决于驱动完备度。</p>
镜像格式	VMDK、VHD、QCOW2、RAW、VHDX、QED、VDI、QCOW、ZVHD2和ZVHD。
镜像大小	<p>如果镜像大小不超过128GB，使用导入镜像功能。</p> <p>如果镜像大小介于128GB和1TB之间，需要将镜像文件转换为RAW或ZVHD2格式，然后使用快速导入功能进行导入。</p> <ul style="list-style-type: none"> • 参考“转换镜像格式”章节转换镜像格式。 • 参考“快速导入镜像文件”了解快速导入功能。

其他约束限制

- 暂不支持创建带有数据盘的镜像，镜像文件中必须只能包含系统盘，且系统盘大小范围为：[40GB, 1024GB]
- 镜像文件的初始密码至少包含以下4种字符：大写字母、小写字母、数字、特殊字符（!@\$%^&_+=+[{ } ; , / ?）
- 镜像启动分区和系统分区必须包含在同一个磁盘中。
- 外部镜像文件必须包含可用的Tenant Administrator账号和密码。
- 镜像引导方式一般为BIOS，部分操作系统镜像支持UEFI，支持列表请参见“[支持UEFI启动方式的操作系统版本](#)”。

3.4.3 上传镜像文件（Windows）

推荐您使用OBS Browser+工具将外部镜像文件上传至OBS个人桶，详细操作请参见《对象存储服务用户指南》手册中“控制台指南 > 入门”的“上传文件”章节。

📖 说明

- OBS桶和镜像文件的存储类别必须是标准存储。

3.4.4 注册镜像（Windows）

操作场景

镜像文件上传到OBS桶以后，请参考本节指导注册镜像。

操作步骤

步骤1 登录IMS控制台。

1. 登录管理控制台。
 2. 选择“计算 > 镜像服务”。
- 进入镜像服务页面。


步骤2 创建系统盘镜像。

1. 单击右上角的“创建私有镜像”，进入创建私有镜像页面。
2. 根据界面要求填写如下信息：
包含“镜像类型和来源”和“配置信息”两个信息块，各参数说明参见[表3-7](#)和[表3-8](#)。

表 3-7 镜像类型和来源

参数	说明
创建方式	选择“系统盘镜像”。
选择镜像源	选择“镜像文件”，然后从列表中选择保存镜像文件的桶，再选择对应的镜像文件。
快速通道	<p>可选参数，仅在选择zvhd2或raw格式的镜像文件时出现。快速通道功能可快速完成镜像制作，并且支持大文件上传（不超过1TB），但是镜像文件需要转换为zvhd2或raw格式并完成镜像优化。如果您已提前准备好符合要求的镜像文件，请勾选“开启快速通道”，然后勾选“镜像文件准备”后的确认信息。</p> <p>说明 镜像文件格式转换、生成位表文件等操作请参考快速导入镜像文件。</p>

表 3-8 配置信息

参数	说明
进行后台自动化配置	勾选后，后台系统将会对镜像文件进行相关检查及优化，具体包括哪些操作请参见 通过镜像文件注册私有镜像过程中，系统会对镜像做哪些修改？ 。
进行自动化签名	勾选后，系统将采用签名方式创建私有镜像，以确保镜像数据的可靠性。 <ul style="list-style-type: none"> - 签名密钥名称：请在下拉列表中选择使用的密钥。如果没有，请登录密钥管理服务DEW控制台，创建密钥。 - 签名密钥ID：使用的密钥ID。
镜像用途	取值为“ECS系统盘镜像”，表示此镜像用来创建弹性云服务器。
启动方式	可选参数，取值为“BIOS”和“UEFI”，两者的区别请参见 UEFI启动方式与BIOS启动方式有哪些区别？ 。 支持UEFI启动方式的操作系统版本请参见 支持UEFI启动方式的操作系统版本 。 此选项需用户确认待注册镜像文件本身的启动方式，并通过此选项告知云平台，以便于云平台完成镜像文件启动方式的相关配置。请选择正确的启动方式，否则，使用该镜像创建的弹性云服务器无法启动。
操作系统	为保证镜像的正常创建和使用，请确保选择的操作系统与镜像文件的操作系统类型一致。未选择时，系统会自动识别镜像文件的操作系统。 说明 <ul style="list-style-type: none"> - 系统识别的镜像文件操作系统与用户设置的操作系统不同时，以系统识别的操作系统为准。 - 系统不能识别镜像文件的操作系统时，以用户选择的操作系统为准。 - 用户选择或系统识别的镜像文件操作系统与实际不一致时，可能会对由此镜像文件最终创建的弹性云服务器的性能产生影响。
系统盘	设置系统盘容量，范围为40-1024GB，请确保输入的大小不小于源主机镜像文件的系统盘大小。 说明 如果上传使用vhd格式的镜像是通过qemu-img或者其他工具转换生成的，设置系统盘容量时请参考 为什么VHD格式的镜像上传失败，任务中心查看报错为外部镜像文件的系统盘容量大于用户设置的系统磁盘容量？ 进行检查。
数据盘	您还可以增加多块数据盘随系统盘镜像一起创建，需要事先制作好数据盘镜像文件。该功能一般适用于将其他平台的虚拟机及其数据盘一起迁移至本平台。 操作方法：单击  图标增加一块数据盘，设置数据盘容量，然后单击“选择镜像文件”，从列表中先选择保存镜像文件的桶，再选择对应的数据盘镜像文件。 最多添加3块数据盘。

参数	说明
名称	设置一个便于您识别的镜像名称。
描述	可选参数，对镜像进行描述。

📖 说明

以上为全量参数，不同企业、单位支持情况不同，以实际界面为准。

3. 单击“立即创建”，确认镜像参数，然后单击“提交申请”。

步骤3 返回私有镜像列表，等待镜像状态变为“正常”，注册成功。

📖 说明

根据镜像文件大小不同，将镜像文件注册为私有镜像所使用的时间不同，请耐心等待。

---结束

3.4.5 使用镜像创建弹性云服务器（Windows）

操作场景

将外部镜像文件注册成云平台的私有镜像后，您可以使用该镜像创建新的云服务器，或对已有云服务器的操作系统进行更换。

本节介绍使用镜像创建云服务器的操作。

操作步骤

请按照[通过镜像创建云服务器](#)中的操作指导创建弹性云服务器。

在配置参数时，需要注意以下几点：

- 区域：必须选择私有镜像所在的区域。
- 规格：在选择规格时，需要结合镜像的操作系统类型以及[弹性云服务器类型与支持的操作系统版本](#)了解支持选择的规格范围。
- 镜像：选择“私有镜像”，并在下拉列表中选择所创建的私有镜像。

后续操作

系统盘镜像创建成功后，您可以为已有云服务器切换操作系统。

3.5 通过外部镜像文件创建 Linux 系统盘镜像

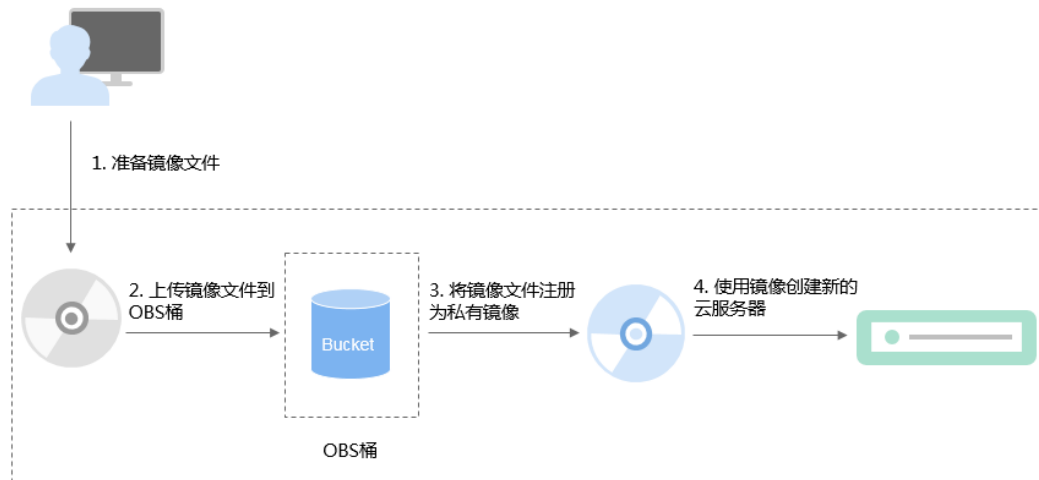
3.5.1 流程概览（Linux）

除了可以通过云服务器创建私有镜像，系统也支持外部镜像导入功能，可将您本地或者其他云平台的服务器系统盘镜像文件导入至镜像服务私有镜像中。导入后，您可以使用该镜像创建新的云服务器，或对已有云服务器的系统进行重装。

创建过程

私有镜像创建过程如[图3-4](#)所示。

图 3-4 Linux 系统盘镜像创建过程



步骤说明如下：

1. 准备符合平台要求的外部镜像文件，请参考[准备镜像文件（Linux）](#)。
2. 上传外部镜像文件到OBS个人桶中，请参考[上传镜像文件（Linux）](#)。
3. 通过管理控制台选择上传的镜像文件，并将镜像文件注册为私有镜像，请参考[注册镜像（Linux）](#)。
4. 私有镜像注册成功后，使用该镜像创建新的云服务器，请参考[使用镜像创建弹性云服务器（Linux）](#)。

3.5.2 准备镜像文件（Linux）

您需要提前准备好符合条件的镜像文件。

📖 说明

大文件导入（不超过1TB）目前仅支持RAW和ZVHD2格式，RAW镜像文件除了要满足[表3-10](#)的要求外，还需要生成位表文件，同镜像文件一并上传。详细操作请参见[快速导入镜像文件](#)。

外部镜像文件导出前的初始化配置

以下操作配置需要在虚拟机内部完成，强烈建议您在原平台的虚拟机实施修改后，再导出镜像文件。如果您在导出镜像文件前未完成以下配置，推荐您使用云服务器完成这些配置，具体操作请参见[Linux外部镜像文件在导出前未完成初始化配置，怎么办？](#)。

表 3-9 外部镜像文件导出前的初始化配置

配置分类	配置项
网络能力	<p>必选项，不设置会导致云服务器启动异常或网络能力异常，包括：</p> <ul style="list-style-type: none"> • 清理网络规则文件 • 设置网卡属性为DHCP <p>可选项，即增值能力，主要包括：</p> <ul style="list-style-type: none"> • 开启网卡多队列 开启网卡多队列功能可以将网卡中断分散给不同的CPU处理，实现负载均衡，从而提升网络PPS和带宽性能。操作方法请参考如何设置镜像的网卡多队列属性？。 • 配置动态获取IPv6地址 IPv6的使用，可以有效弥补IPv4网络地址资源有限的问题。镜像中配置动态获取IPv6地址，发放的云服务器能够同时支持IPv4和IPv6地址。配置方法请参考如何配置IPv6地址。
工具	<p>强烈建议安装Cloud-Init工具。</p> <p>Cloud-Init是开源的云初始化工具，使用安装了Cloud-Init的镜像创建云服务器时可以通过“用户数据注入”功能，注入初始化自定义信息（例如为云服务器设置登录密码）；还可以通过查询、使用元数据，对正在运行的云服务器进行配置和管理。不安装Cloud-Init工具，将无法对云服务器进行自定义配置，只能使用镜像原有密码登录云服务器。</p> <p>安装方法请参考安装Cloud-Init工具。</p>
插件	<p>为了保证使用私有镜像创建的新云服务器可以实现一键式重置密码功能，建议您在创建私有镜像前安装密码重置插件CloudResetPwdAgent。</p>
驱动	<p>安装原生的KVM驱动</p>
文件系统	<ul style="list-style-type: none"> • 修改grub文件磁盘标识方式为UUID • 修改fstab文件磁盘标识方式为UUID
数据盘	<p>如果创建私有镜像所使用的原云服务器挂载了多个数据盘，可能导致由私有镜像创建的新云服务器无法使用。因此在创建私有镜像前，需要卸载原云服务器中挂载的所有数据盘。</p> <p>详情请参考卸载云服务器的数据盘。</p>

镜像文件属性限制

表 3-10 Linux 操作系统的镜像文件属性限制

镜像文件属性	条件
操作系统	<ul style="list-style-type: none"> • SUSE、Oracle Linux、Red Hat、Ubuntu、openSUSE、CentOS、Debian、Fedora、EulerOS、中标麒麟 • 支持32位和64位 • 操作系统不能与特定的硬件绑定 • 操作系统必须支持全虚拟化 <p>所支持的操作系统版本请参考外部镜像文件支持的格式和操作系统类型，在此范围内的操作系统支持后台自动化配置（详情请参阅通过镜像文件注册私有镜像过程中，系统会对镜像做哪些修改？），在此之外的操作系统请您自行排查及安装virtio驱动（详见安装原生的KVM驱动），在注册镜像页面选择Other Linux，导入后系统启动情况取决于驱动完备度。</p>
镜像格式	VMDK、VHD、QCOW2、RAW、VHDX、QED、VDI、QCOW、ZVHD2和ZVHD
镜像大小	<p>镜像大小不超过128GB。</p> <p>如果镜像大小介于128GB和1TB之间，需要将镜像文件转换为RAW或ZVHD2格式，然后使用快速导入功能进行导入。</p> <ul style="list-style-type: none"> • 参考“转换镜像格式”章节转换镜像格式。 • 参考“快速导入镜像文件”了解快速导入功能。

其他约束限制

- 暂不支持创建带有数据盘的镜像，镜像文件中必须只能包含系统盘，且系统盘大小范围为：[40GB, 1024GB]
- 镜像文件的初始密码至少包含以下4种字符：大写字母、小写字母、数字、特殊字符（!@\$%^&_+=+[{]!./?）
- 镜像启动分区和系统分区必须包含在同一个磁盘中
- 镜像引导方式一般为BIOS，部分操作系统镜像支持UEFI，支持列表请参见《镜像服务用户指南》的“支持UEFI启动方式的操作系统版本”。
- “/etc/fstab”文件中不能包含非系统盘的自动挂载信息，否则创建的云服务器可能无法正常登录。
- 如果外部镜像文件的系统盘为LVM设备，通过该镜像文件注册的私有镜像用来创建云服务器时，不支持文件注入。
- 外部镜像文件所在虚拟机如果经历了关机过程，则必须是优雅关机，否则使用私有镜像创建的云服务器在启动时可能会出现蓝屏。

3.5.3 上传镜像文件（Linux）

推荐您使用OBS Browser+工具将外部镜像文件上传至OBS个人桶，详细操作请参见《对象存储服务用户指南》手册中“控制台指南 > 入门”的“上传文件”章节。

 说明

- OBS桶和镜像文件的存储类别必须是标准存储。

3.5.4 注册镜像（Linux）

操作场景

镜像文件上传到OBS桶以后，请参考本节指导注册镜像。

操作步骤

步骤1 登录IMS控制台。

1. 登录管理控制台。
2. 选择“计算 > 镜像服务”。
进入镜像服务页面。

步骤2 创建系统盘镜像。


1. 单击右上角的“创建私有镜像”，进入创建私有镜像页面。
2. 根据界面要求填写如下信息：
包含“镜像类型和来源”和“配置信息”两个信息块，各参数说明参见[表3-11](#)和[表3-12](#)。

表 3-11 镜像类型和来源

参数	说明
创建方式	选择“系统盘镜像”。
选择镜像源	选择“镜像文件”，然后从列表中选择保存镜像文件的桶，再选择对应的镜像文件。
快速通道	<p>可选参数，仅在选择zvhd2或raw格式的镜像文件时出现。快速通道功能可快速完成镜像制作，并且支持大文件上传（不超过1TB），但是镜像文件需要转换为zvhd2或raw格式并完成镜像优化。如果您已提前准备好符合要求的镜像文件，请勾选“开启快速通道”，然后勾选“镜像文件准备”后的确认信息。</p> <p>说明 镜像文件格式转换、生成位表文件等操作请参考快速导入镜像文件。</p>

表 3-12 配置信息

参数	说明
进行后台自动化配置	勾选后，后台系统将会对镜像文件进行相关检查及优化，具体包括哪些操作请参见 通过镜像文件注册私有镜像过程中，系统会对镜像做哪些修改？ 。

参数	说明
进行自动化签名	<p>勾选后，系统将采用签名方式创建私有镜像，以确保镜像数据的可靠性。</p> <ul style="list-style-type: none"> - 签名密钥名称：请在下拉列表中选择使用的密钥。如果没有，请登录密钥管理服务DEW控制台，创建密钥。 - 签名密钥ID：使用的密钥ID。
镜像用途	取值为“ECS系统盘镜像”，表示此镜像用来创建弹性云服务器。
启动方式	<p>可选参数，取值为“BIOS”和“UEFI”，两者的区别请参见UEFI启动方式与BIOS启动方式有哪些区别？。</p> <p>支持UEFI启动方式的操作系统版本请参见支持UEFI启动方式的操作系统版本。</p> <p>此选项需用户确认待注册镜像文件本身的启动方式，并通过此选项告知云平台，以便于云平台完成镜像文件启动方式的相关配置。请选择正确的启动方式，否则，使用该镜像创建的弹性云服务器无法启动。</p>
操作系统	<p>为保证镜像的正常创建和使用，请确保选择的操作系统与镜像文件的操作系统类型一致。未选择时，系统会自动识别镜像文件的操作系统。</p> <p>说明</p> <ul style="list-style-type: none"> - 系统识别的镜像文件操作系统与用户设置的操作系统不同时，以系统识别的操作系统为准。 - 系统不能识别镜像文件的操作系统时，以用户选择的操作系统为准。 - 用户选择或系统识别的镜像文件操作系统与实际不一致时，可能会对由此镜像文件最终创建的弹性云服务器的性能产生影响。
系统盘	<p>设置系统盘容量，范围为40-1024GB，请确保输入的大小不小于源主机镜像文件的系统盘大小。</p> <p>说明</p> <p>如果上传使用vhd格式的镜像是通过qemu-img或者其他工具转换生成的，设置系统盘容量时请参考为什么VHD格式的镜像上传失败，任务中心查看报错为外部镜像文件的系统盘容量大于用户设置的系统磁盘容量？进行检查。</p>
数据盘	<p>您还可以增加多块数据盘随系统盘镜像一起创建，需要事先制作好数据盘镜像文件。该功能一般适用于将其他平台的虚拟机及其数据盘一起迁移至本平台。</p> <p>操作方法：单击图标增加一块数据盘，设置数据盘容量，然后单击“选择镜像文件”，从列表中先选择保存镜像文件的桶，再选择对应的数据盘镜像文件。</p> <p>最多添加3块数据盘。</p>
名称	设置一个便于您识别的镜像名称。
描述	可选参数，对镜像进行描述。

📖 说明

以上为全量参数，不同企业、单位支持情况不同，以实际界面为准。

- 单击“立即创建”，确认镜像参数，然后单击“提交申请”。

步骤3 返回私有镜像列表，等待镜像状态变为“正常”，注册成功。

📖 说明

根据镜像文件大小不同，将镜像文件注册为私有镜像所使用的时间不同，请耐心等待。

----结束

3.5.5 使用镜像创建弹性云服务器（Linux）

操作场景

将外部镜像文件注册成云平台的私有镜像后，您可以使用该镜像创建新的云服务器，或对已有云服务器的操作系统进行更换。

本节介绍使用镜像创建云服务器的操作。

操作步骤

请按照[通过镜像创建云服务器](#)中的操作指导创建弹性云服务器。

在配置参数时，需要注意以下几点：

- 区域：必须选择私有镜像所在的区域。
- 规格：在选择规格时，需要结合镜像的操作系统类型以及[弹性云服务器类型与支持的操作系统版本](#)了解支持选择的规格范围。
- 镜像：选择“私有镜像”，并在下拉列表中选择所创建的私有镜像。

后续操作

系统盘镜像创建成功后，您可以为已有云服务器切换操作系统。

3.6 通过数据盘创建数据盘镜像

操作场景

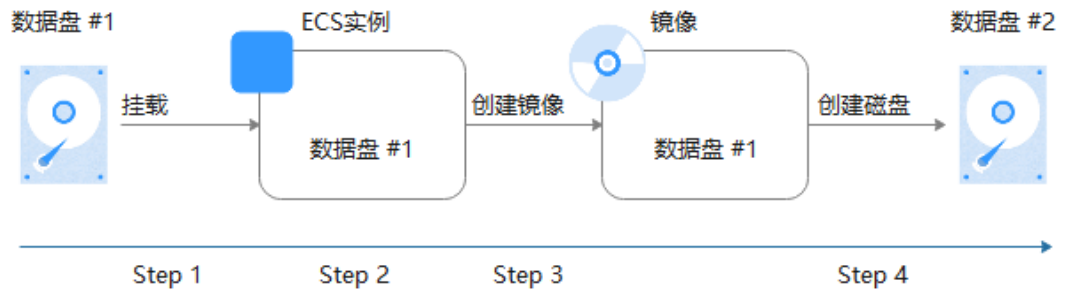
数据盘镜像是指只包含用户业务数据的镜像，您可以通过创建数据盘镜像将云服务器数据盘上的业务数据保存。数据盘镜像可用来创建新的云硬盘，从而实现用户业务数据的迁移。

场景示例：通过制作数据盘镜像，将快到期云服务器的磁盘数据导出。

背景知识

通过云服务器的数据盘创建数据盘镜像的过程如下：从现有数据盘开始，挂载至云服务器实例，初始化数据盘并写入数据，创建数据盘镜像，并最终创建新的数据盘，如图3-5所示。

图 3-5 数据盘镜像使用流程



前提条件

- 云服务器已挂载数据盘，并且云服务器处于开机或关机状态。挂载数据盘请参考《弹性云服务器用户指南》。
- 用于创建数据盘镜像的数据盘，磁盘容量需 $\leq 1\text{TB}$ 。
数据盘容量大于1TB时，不支持创建数据盘镜像，此时请选择创建整机镜像。

操作步骤

步骤1 登录IMS控制台。

1. 登录管理控制台。
2. 选择“计算 > 镜像服务”。
进入镜像服务页面。

步骤2 创建数据盘镜像。

1. 单击右上角的“创建私有镜像”，进入创建私有镜像页面。
2. 在“镜像类型和来源”区域，选择镜像的创建方式为“数据盘镜像”。
3. 镜像源选择“云服务器”，在云服务器已挂载的磁盘列表中选择一块数据盘。
4. 在“配置信息”区域，输入镜像的基本信息名称，并根据需要，输入镜像的描述。

📖 说明

以上为全量参数，不同企业、单位支持情况不同，以实际界面为准。

5. 单击“立即创建”。
6. 确认镜像参数，单击“提交申请”。

步骤3 返回私有镜像列表，查看创建的数据盘镜像。

----结束

后续操作

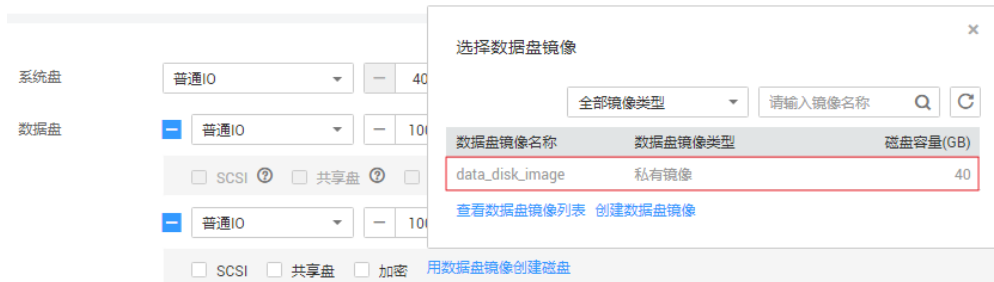
数据盘镜像创建成功后，您可能需要创建新的数据盘并挂载到目标云服务器上，有两种方法供您参考：

- 在创建好的数据盘镜像所在行，单击“申请数据盘”创建新的数据盘（支持批量创建），然后挂载至目标云服务器。
- 在创建弹性云服务器向导页面，为云服务器添加数据盘时，单击“用数据盘镜像创建磁盘”，然后选择一个数据盘镜像。

说明

此方式下，每个数据盘镜像只可以用来创建一个数据盘，不能重复选择。例如，为云服务器添加了一块数据盘，这个数据盘来源是数据盘镜像data_disk_image，如果想再添加一块数据盘，则不能使用数据盘镜像data_disk_image来创建。

图 3-6 添加数据盘



3.7 通过外部镜像文件创建数据盘镜像

操作场景

数据盘镜像只包含用户业务数据的镜像，您可以通过本地或者其他云平台上的外部镜像文件创建数据盘镜像。数据盘镜像可以用于创建云硬盘，将用户的业务数据迁移到云上。

流程说明

通过外部镜像文件创建数据盘镜像的过程如下：

图 3-7 创建过程



1. 准备符合格式要求的外部镜像文件。当前支持vhd、vmdk、qcow2、raw、vhdx、qcow、vdi、qed、zvhd或zvhd2格式，其他镜像文件，需要转换格式后再导入。
2. 上传外部镜像文件到OBS个人桶中，注意OBS桶和镜像文件的存储类别必须是标准存储。具体操作可参考[上传镜像文件（Linux）](#)。
3. 创建数据盘镜像，具体操作请参见[操作步骤](#)。
4. 使用数据盘镜像创建新的数据盘，具体操作请参见[后续操作](#)。

操作步骤

步骤1 登录IMS控制台。

1. 登录管理控制台。
2. 选择“计算 > 镜像服务”。
进入镜像服务页面。

步骤2 创建数据盘镜像。

1. 单击右上角的“创建私有镜像”，进入创建私有镜像页面。
2. 在“镜像类型和来源”区域，选择镜像的创建方式为“数据盘镜像”。
3. 镜像源选择“镜像文件”，从列表中先选择保存镜像文件的桶，再选择对应的镜像文件。
4. 在“配置信息”区域，完成以下参数填写：
 - 操作系统类型：必须明确指定操作系统类型，取值为Windows或Linux。
 - 数据盘：输入数据盘的大小，范围为1~2048GB，并且请确保不小于镜像文件的数据盘大小。
 - 名称：设置一个便于您识别的镜像名称。
 - （可选）描述：对镜像进行描述。

说明

以上为全量参数，不同企业、单位支持情况不同，以实际界面为准。

5. 单击“立即创建”。
6. 确认镜像参数，单击“提交申请”。

步骤3 返回私有镜像列表，查看创建的数据盘镜像。

当镜像的状态为“正常”时，表示创建完成。

----结束

后续操作

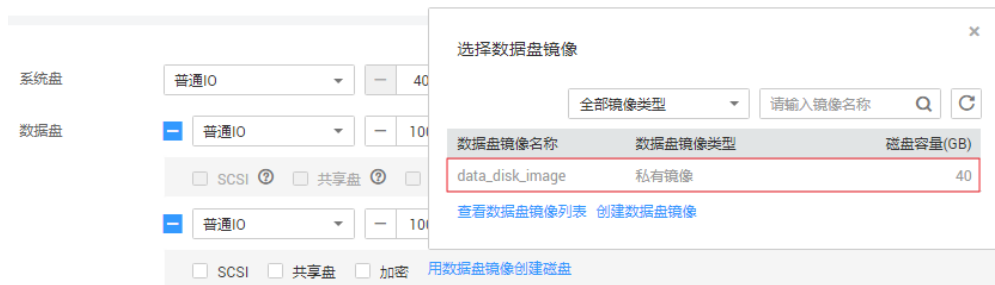
数据盘镜像创建成功后，您可能需要创建新的数据盘并挂载到目标云服务器上，有两种方法供您参考：

- 在创建好的数据盘镜像所在行，单击“申请数据盘”创建新的数据盘（支持批量创建），然后挂载至目标云服务器。
- 在创建弹性云服务器向导页面，为云服务器添加数据盘时，单击“用数据盘镜像创建磁盘”，然后选择一个数据盘镜像。

说明

此方式下，每个数据盘镜像只可以用来创建一个数据盘，不能重复选择。例如，为云服务器添加了一块数据盘，这个数据盘来源是数据盘镜像data_disk_image，如果想再添加一块数据盘，则不能使用数据盘镜像data_disk_image来创建。

图 3-8 添加数据盘



3.8 通过云服务器创建整机镜像

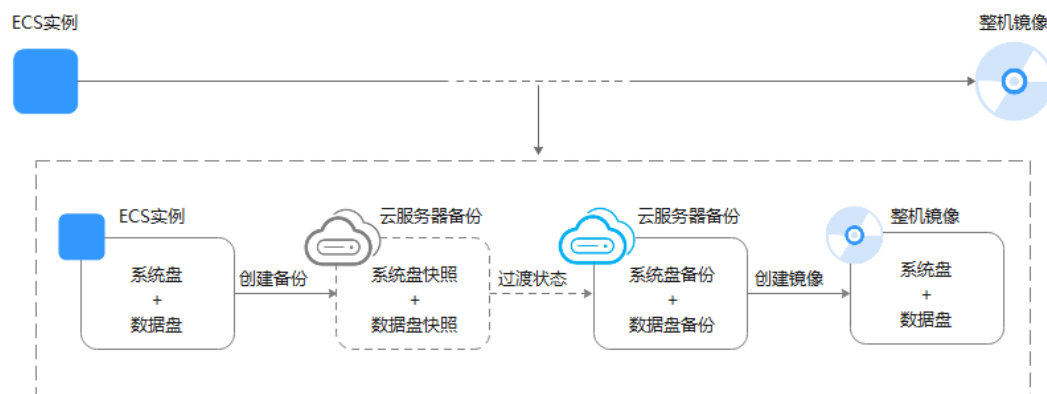
操作场景

使用弹性云服务器携带其挂载的数据盘一起创建整机镜像，创建的整机镜像包含操作系统、应用软件，以及用户的业务数据，可用于快速发放相同配置的弹性云服务器，实现数据搬迁。

背景知识

创建整机镜像的流程：先为云服务器创建云服务器备份，再通过云服务器备份创建镜像，中间过程为系统自动完成的，您只需要耐心等待即可。

图 3-9 整机镜像创建流程



- 创建整机镜像的时间与磁盘大小、网络状态、并发任务数等有关。
- 使用云服务器制作整机镜像时，云服务器应处于开机或者关机状态。如果镜像中包含数据库，请在关机状态下制作。
- 在云服务器关机状态下，制作整机镜像的过程中，用户不能启动云服务器。
- 整机镜像创建过程中，请勿对源云服务器做任何操作，避免创建失败。
- 图3-9中，当云服务器备份还未创建成功，只是创建好磁盘快照时，整机镜像状态为“可用区x可用”（可用区x表示镜像源云服务器所在的可用区），此时，这个整机镜像只能在该可用区中发放云服务器。如果需要在区域内发放云服务器，需要等云服务器备份完全创建好，并且整机镜像状态为“正常”。

- 选择整机镜像切换云服务器操作系统时，仅能恢复整机镜像的系统盘数据。如果希望恢复或者迁移整机镜像中的数据盘数据，必须通过整机镜像创建新的云服务器。

约束与限制

- 创建整机镜像时，请确保云服务器已完成相关配置，否则可能导致整机镜像创建云服务器失败。
- 使用Windows操作系统云服务器制作整机镜像时，不允许云服务器存在跨区卷，否则制作的整机镜像创建云服务器时可能会导致数据丢失。
- 使用Linux操作系统云服务器制作整机镜像时，不允许云服务器存在由多个物理卷组成的卷组、或由多个物理卷组成的逻辑卷，否则制作的整机镜像创建云服务器时可能会导致数据丢失。
- 带有专属存储磁盘的云服务器不支持创建整机镜像。
- 整机镜像不支持如下操作：不允许导出、复制、共享。
- 使用Windows操作系统云服务器制作整机镜像时，需修改云服务器SAN策略为OnlineAll类型。否则可能导致使用镜像创建云服务器时，云服务器磁盘处于脱机状态。

Windows操作系统SAN策略分为三种类型：OnlineAll、OfflineShared、OfflineInternal

表 3-13 Windows 操作系统 SAN 策略类型

类型	说明
OnlineAll	表示所有新发现磁盘都置于在线模式。
OfflineShared	表示所有共享总线上（比如FC、iSCSI）的新发现磁盘都置于离线模式，非共享总线上的磁盘都置于在线模式。
OfflineInternal	表示所有新发现磁盘都置于离线模式。

- 运行cmd.exe，执行以下命令，使用DiskPart工具来查询云服务器当前的SAN策略。
diskpart
- 执行以下命令查看云服务器当前的SAN策略。
san
 - 如果SAN策略为OnlineAll，请执行**exit**命令退出DiskPart。
 - 否，请执行步骤c。
- 执行以下命令修改云服务器SAN策略为OnlineAll。
san policy=onlineall

操作步骤

步骤1 登录IMS控制台。

1. 登录管理控制台。
2. 选择“计算 > 镜像服务”。
进入镜像服务页面。

步骤2 创建整机镜像。

1. 单击右上角的“创建私有镜像”，进入创建私有镜像页面。
2. 在“镜像类型和来源”区域，选择镜像的创建方式为“整机镜像”。
3. 镜像源选择“云服务器”，从列表中选择相应的云服务器。
4. 选择“云服务器备份存储库”，用于存放备份。
创建的整机镜像及中间产物备份副本会默认存放在备份存储库中，因此必须选择一个云服务器备份存储库。
如果没有可用的存储库，请单击“新建云服务器备份存储库”进行创建，注意“保护类型”需选择“备份”。更多关于存储库、云备份的相关信息，请参见《云备份用户指南》。
5. 在“配置信息”区域，填写镜像的基本信息。例如，镜像的名称和镜像描述。
6. 单击“立即创建”。
7. 确认镜像参数，单击“提交申请”。

步骤3 返回私有镜像列表，查询创建的整机镜像的状态。

- 当镜像的状态为“正常”时，表示创建完成。
- 整机镜像状态为“可用区x可用”（可用区x表示镜像源云服务器所在的可用区）时：表示云服务器备份还未创建成功，只是创建好磁盘快照。

此时，这个整机镜像只能在该可用区中发放云服务器。如果需要在区域内发放云服务器，需要等云服务器备份完全创建好，并且整机镜像状态为“正常”，该过程大概需要10分钟，具体由云服务器的数据量决定，数据量越大，时间越长。

图 3-10 整机镜像状态

名称	状态	操作系统类型	操作系统	镜像类型
full-image-ecs-00	正常 可用区2可用	Windows	Windows Server 20...	ECS整机镜像(x86)

----结束

后续操作

- 整机镜像创建成功后，如果您想使用该镜像创建弹性云服务器，请在操作列单击“申请服务器”，进入创建云服务器的向导页面，请参考《弹性云服务器用户指南》完成操作。

说明

使用整机镜像创建弹性云服务器时：

- 系统会自动设置好系统盘和数据盘参数。
- 如果整机镜像中包含多块数据盘，需要加载一段时间才能正常显示系统盘信息，请耐心等待。
- 选择整机镜像切换云服务器操作系统时，仅能恢复整机镜像的系统盘数据。如果希望恢复或者迁移整机镜像中的数据盘数据，必须通过整机镜像创建新的云服务器。

3.9 通过云服务器备份创建整机镜像

操作场景

使用已有的云服务器备份制作整机镜像，可用于将云服务器备份创建为新的弹性云服务器。

约束与限制

- 创建整机镜像时，请确保创建该云服务器备份所属的弹性云服务器已完成相关配置，否则可能导致整机镜像创建云服务器失败。
- 在云服务器关机状态下，制作整机镜像的过程中，用户不能启动云服务器。
- 使用云服务器备份创建整机镜像时，不允许云服务器备份存在共享卷。
- 使用云服务器备份创建整机镜像时，云服务器备份必须是可用状态，且每个云服务器备份只能创建一个整机镜像。
- 整机镜像不支持如下操作：不允许导出、复制、共享。

操作步骤

步骤1 登录IMS控制台。

1. 登录管理控制台。
2. 选择“计算 > 镜像服务”。
进入镜像服务页面。

步骤2 创建整机镜像。

1. 单击右上角的“创建私有镜像”，进入创建私有镜像页面。
2. 在“镜像类型和来源”区域，选择镜像的创建方式为“整机镜像”。
3. 镜像源选择“云服务器备份”，从列表中选择相应的云服务器备份。
4. 在“配置信息”区域，填写镜像的基本信息。例如，镜像的名称和镜像描述。
5. 单击“立即创建”。
6. 确认镜像参数，单击“提交申请”。

步骤3 返回私有镜像界面查询创建的整机镜像的状态。

当镜像的状态为“正常”时，表示创建完成。

----结束

后续操作

- 整机镜像创建成功后，如果您想使用该镜像创建弹性云服务器，请在操作列单击“申请服务器”，进入创建云服务器的向导页面，请参考《弹性云服务器用户指南》完成操作。

📖 说明

使用整机镜像创建弹性云服务器时：

- 系统会自动设置好系统盘和数据盘参数。
- 如果整机镜像中包含多块数据盘，需要加载一段时间才能正常显示系统盘信息，请耐心等待。
- 选择整机镜像切换云服务器操作系统时，仅能恢复整机镜像的系统盘数据。如果希望恢复或者迁移整机镜像中的数据盘数据，必须通过整机镜像创建新的云服务器。
- 如果您想共享整机镜像给其他租户，需要将资源迁移至云备份（CBR），因为只有通过云备份（CBR）创建的整机镜像才支持共享功能。

3.10 通过云备份创建整机镜像

操作场景

使用已有的云备份制作整机镜像，可用于将云备份创建为新的弹性云服务器。

背景知识

- 云备份（Cloud Backup and Recovery, CBR）服务提供对云硬盘、弹性云服务器的备份保护服务，并支持利用备份数据恢复服务器和磁盘的数据。如果您已通过CBR服务为弹性云服务器创建云备份，可以参考本章节，使用已有的云备份制作整机镜像。
- 选择整机镜像切换云服务器操作系统时，仅能恢复整机镜像的系统盘数据。如果希望恢复或者迁移整机镜像中的数据盘数据，必须通过整机镜像创建新的云服务器。

约束与限制

- 创建整机镜像前，请确保创建该云备份所属的弹性云服务器已完成相关配置，否则可能导致整机镜像创建云服务器失败。
- 每个云备份只能创建一个整机镜像。
- 在云服务器关机状态下，制作整机镜像的过程中，用户不能启动云服务器。
- 整机镜像不支持如下操作：不允许导出、复制、共享。

操作步骤

步骤1 登录IMS控制台。

1. 登录管理控制台。
 2. 选择“计算 > 镜像服务”。
- 进入镜像服务页面。

步骤2 创建整机镜像。

1. 单击右上角的“创建私有镜像”，进入创建私有镜像页面。
2. 在“镜像类型和来源”区域，选择镜像的创建方式为“整机镜像”。
3. 镜像源选择“云备份”，从列表中选择相应的云备份。

4. 在“配置信息”区域，填写镜像的基本信息。例如，镜像的名称和镜像描述。
5. 单击“立即创建”。
6. 确认镜像参数，单击“提交申请”。

步骤3 返回私有镜像界面查询创建的整机镜像的状态。

当镜像的状态为“正常”时，表示创建完成。

----结束

后续操作

整机镜像创建成功后，您可以进行以下操作：

- 如果想使用该镜像创建弹性云服务器，请在操作列单击“申请服务器”，进入创建云服务器的向导页面，在选择镜像时，选择“私有镜像”即可。详细操作请参考《弹性云服务器用户指南》完成操作。

📖 说明

使用整机镜像创建弹性云服务器时：

- 系统会自动设置好系统盘和数据盘参数。
- 如果整机镜像中包含多块数据盘，需要加载一段时间才能正常显示系统盘信息，请耐心等待。
- 如果要将该镜像共享给其他账号，请在操作列单击“更多 > 共享”，输入对方的账号名即可。详细操作请参考[共享指定镜像](#)。
- 选择整机镜像切换云服务器操作系统时，仅能恢复整机镜像的系统盘数据。如果希望恢复或者迁移整机镜像中的数据盘数据，必须通过整机镜像创建新的云服务器。

3.11 通过 ISO 文件创建 Windows 系统盘镜像

3.11.1 创建流程（Windows）

ISO是一种光盘映像文件，通过特定的压缩方式，将大量的数据文件统一为一个后缀名为iso的映像文件。ISO文件可以理解为从光盘中复制出来的数据文件，所以ISO文件无法直接使用，需要利用一些工具进行解压后才能使用。例如使用虚拟光驱打开，或者将ISO文件刻录到光盘中后，使用光驱来进行读取。

本章介绍通过ISO文件创建Windows系统盘镜像的操作过程。

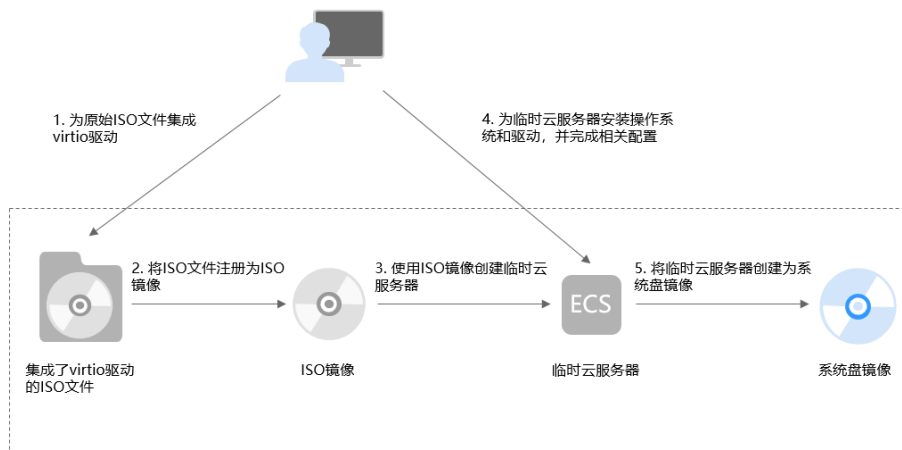
📖 说明

仅适用于管理控制台操作。如果您是API用户，请参考《镜像服务API参考》的“使用ISO文件创建镜像”章节。

创建过程

通过ISO文件创建Windows系统盘镜像的流程如[图3-11](#)所示。

图 3-11 创建过程（Windows 系统盘镜像）



步骤说明如下：

1. 为原始ISO文件集成virtio驱动

由于Windows操作系统采用ide磁盘和virtio网卡，因此需要在注册为云平台的镜像前，先在Windows操作系统的ISO文件中集成virtio驱动。详情请参考[为原始ISO文件集成virtio驱动](#)。

2. 将ISO文件注册为ISO镜像

在管理控制台上将已集成virtio驱动的ISO文件注册为镜像。该镜像为ISO镜像，不能直接用于发放云服务器。详情请参考[将ISO文件注册为镜像（Windows）](#)。

3. 使用ISO镜像创建临时云服务器

使用注册成功的ISO镜像创建一个临时云服务器，该云服务器为空云服务器，即没有操作系统，也未安装任何驱动。详情请参考[使用ISO镜像创建Windows云服务器](#)。

4. 为临时云服务器安装操作系统和驱动，并完成相关配置

包含如下操作：安装操作系统、安装virtio驱动、安装PV driver、配置网卡属性等。详情请参考[安装Windows操作系统和virtio驱动](#)和[配置云服务器并创建Windows系统盘镜像](#)中的步骤步骤1。

5. 将临时云服务器创建为系统盘镜像

在管理控制台上将已完成安装和配置的临时云服务器创建为系统盘镜像。镜像创建成功后，删除临时云服务器避免继续占用计算资源空间。详情请参考[通过云服务器创建Windows系统盘镜像](#)。

约束与限制

- 使用ISO文件注册的ISO镜像无法在创建云服务器页面选择到，也无法用来切换操作系统，必须完成操作系统及驱动安装做成系统盘镜像后才能使用。
- ISO镜像在镜像服务页面创建的临时云服务器只建议用来装机，部分功能受限。如：不支持挂载磁盘操作。

3.11.2 为原始 ISO 文件集成 virtio 驱动

操作场景

由于Windows操作系统采用ide磁盘和virtio网卡，因此需要在注册为云平台的镜像前，先在Windows操作系统的ISO文件中集成virtio驱动。由于ISO通常是将光盘文件做成一个文件，而部分光盘软件设定只能从光驱进行安装，直接解压后无法使用，因此需要用到虚拟光驱软件。

本节操作以AnyBurn、UltraISO软件为例，介绍如何为ISO文件集成virtio驱动。

📖 说明

- AnyBurn是一款轻量级但专业的CD/DVD/蓝光刻录软件，并提供免费版本。
- UltraISO是收费软件，试用版的UltraISO软件不能保存大小超过300M的ISO文件，请使用正式版本的软件。
- virtio是一种用于虚拟化环境的标准化设备接口，旨在提高虚拟机（VM）与宿主机之间的I/O性能。关于virtio驱动程序的详细解释可以参考[官网](#)，virtio-win/kvm-guest-drivers-windows开源代码请参见<https://github.com/virtio-win/kvm-guest-drivers-windows>。

前提条件

已准备ISO文件。

📖 说明

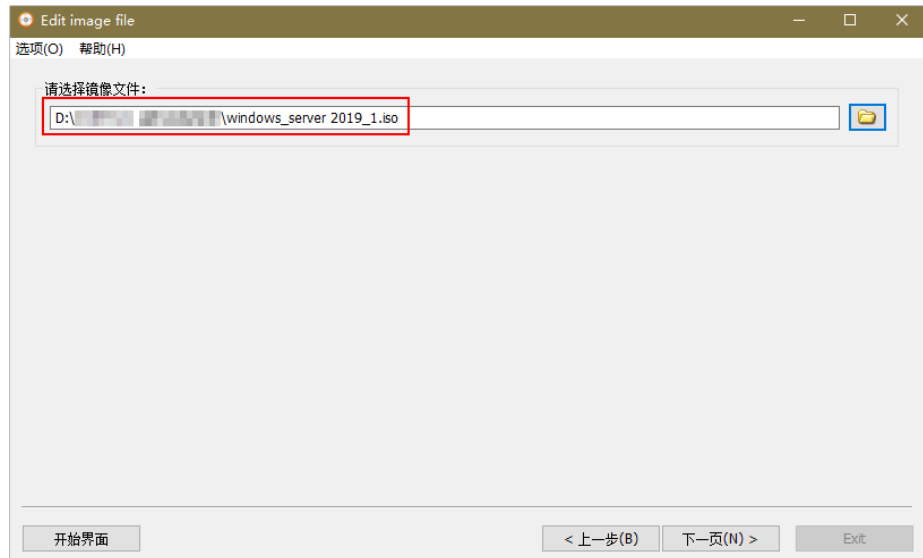
ISO文件名称只能包含英文字母、数字、中划线（-）和下划线（_）。如果不符合要求，请先修改名称。

AnyBurn 软件操作步骤

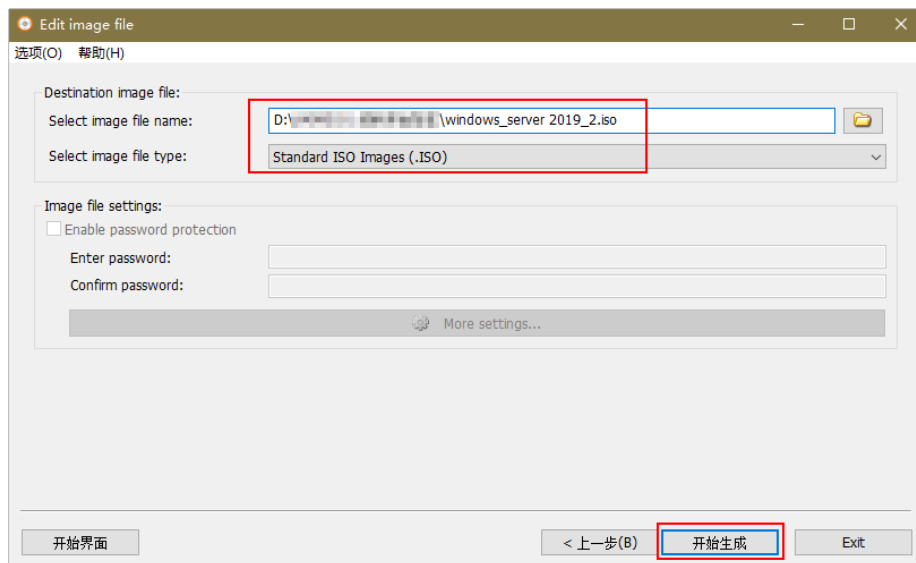
1. 下载**AnyBurn**，并安装至本地。
2. 下载virtio驱动。
下载地址：<https://fedorapeople.org/groups/virt/virtio-win/direct-downloads/stable-virtio/virtio-win.iso>
其他版本获取地址如下：
<https://fedorapeople.org/groups/virt/virtio-win/direct-downloads/archive-virtio/>
3. 使用AnyBurn打开ISO文件。
 - a. 打开AnyBurn软件，选择“编辑镜像文件”。



- b. 选择ISO文件，单击“下一页”。



4. 编辑ISO文件，集成virtio驱动。
- 解压步骤2中下载的“virtio-win.iso”文件。
 - 单击“增加”，将解压后的所有文件添加到ISO文件父节点下，单击“下一步”。
 - 指定保存的路径及ISO文件名称，选择文件保存为ISO格式，单击“开始生成”。
- 生成ISO文件结束后，请在保存路径下查看集成virtio驱动的ISO文件。



UltraISO 软件操作步骤

1. 下载UltraISO，并安装至本地。
下载地址：<https://www.ultraiso.com/>
2. 下载virtio驱动。
下载地址：<https://fedorapeople.org/groups/virt/virtio-win/direct-downloads/stable-virtio/virtio-win.iso>
其他版本获取地址如下：
<https://fedorapeople.org/groups/virt/virtio-win/direct-downloads/archive-virtio/>
3. 使用UltraISO打开ISO文件。

⚠ 注意

ISO格式的镜像文件请使用UltraISO工具打开，请勿使用普通压缩工具打开，否则会丢失ISO的引导数据。

4. 将下载的virtio驱动文件整体拖到ISO文件父节点下。
5. 在UltraISO中，将已集成virtio驱动的ISO文件另存为iso格式，导出至本地。

3.11.3 将 ISO 文件注册为镜像（Windows）

操作场景

通过注册镜像操作，将外部镜像ISO文件注册为云平台的私有镜像，即ISO镜像。注册镜像前，需先将[为原始ISO文件集成virtio驱动](#)中保存的ISO文件上传到OBS桶中。

使用ISO文件注册的ISO镜像不支持复制、导出和加密。

前提条件

- 待注册文件需为iso格式。

- 已上传ISO镜像文件到OBS的个人桶中。具体操作请参见[上传镜像文件 \(Windows\)](#)。

操作步骤

步骤1 登录IMS控制台。

1. 登录管理控制台。
2. 选择“计算 > 镜像服务”。
进入镜像服务页面。

步骤2 创建ISO镜像。

1. 单击右上角的“创建私有镜像”，进入创建私有镜像页面。
2. 在“镜像类型和来源”区域，选择镜像的创建方式为“ISO镜像”。
3. 从镜像文件列表中先选择保存ISO镜像文件的桶，再选择对应的ISO文件。
4. 在“配置信息”区域，填写镜像的基本信息。
 - 架构类型：根据待注册的镜像文件的架构类型，选择“x86”或“ARM”。

📖 说明

该参数仅在支持ARM规格的区域需要配置。

- 启动方式：取值为“BIOS”和“UEFI”。请确保选择的启动方式与镜像文件中的启动方式一致，否则，使用该镜像创建的弹性云服务器无法启动。ARM架构类型不支持BIOS启动方式。
- 操作系统：选择ISO镜像文件对应的操作系统。为保证镜像的正常创建和使用，请确保选择的操作系统与镜像文件的操作系统类型一致。
- 系统盘：设置系统盘容量，范围为40-1024GB，要求不小于镜像文件的系统盘大小。
- 名称：输入镜像的名称。
- 描述：可选配置，根据需要输入描述信息。

📖 说明

以上为全量参数，不同企业、单位支持情况不同，以实际界面为准。

5. 单击“立即创建”。
6. 确认镜像参数，单击“提交申请”。

步骤3 返回私有镜像界面，查看创建的ISO镜像的状态。

当镜像的状态为“正常”时，表示创建成功。

----结束

3.11.4 使用 ISO 镜像创建 Windows 云服务器

操作场景

该任务指导用户使用已注册成功的ISO镜像创建弹性云服务器。

约束与限制

暂不支持专属云用户通过ISO镜像创建弹性云服务器。

用户在指定区域申请了专属云服务，那么该用户就是专属云用户。

操作步骤

步骤1 登录IMS控制台。

1. 登录管理控制台。
2. 选择“计算 > 镜像服务”。
进入镜像服务页面。

步骤2 创建弹性云服务器。

1. 单击“私有镜像”页签，在ISO镜像所在行的“操作”列下，单击“安装服务器”，创建云服务器。
2. 根据界面提示完成云服务器的配置，并单击“确定”。

----结束

后续操作

弹性云服务器创建成功后，使用平台提供的“远程登录”方式，登录弹性云服务器，继续执行后续的安装操作系统和相关驱动操作。

3.11.5 安装 Windows 操作系统和 virtio 驱动

操作场景

该任务以“Windows Server 2019 64位”操作系统为例，指导用户安装Windows操作系统。

由于镜像文件不同，安装步骤稍有不同，请根据实际的安装界面提示进行操作。

说明

请根据实际情况完成时区、KMS地址、补丁服务器、输入法、语言等相关配置。

前提条件

已使用平台提供的“远程登录”方式（即VNC登录），连接云服务器并进入安装界面。

操作步骤

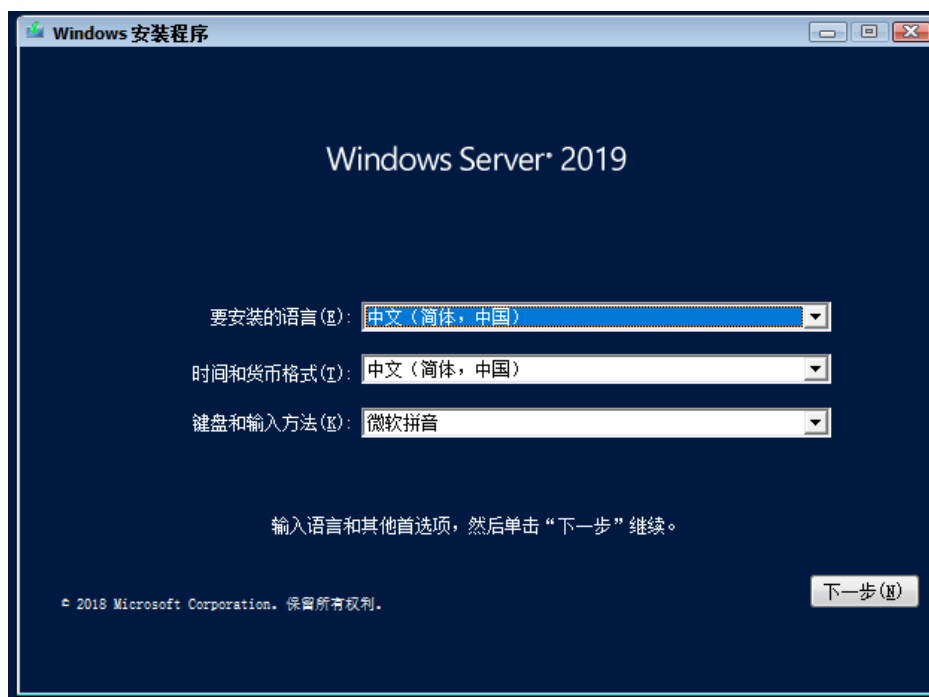
注意

操作系统安装成功前，禁止关闭或重启弹性云服务器，否则会导致操作系统安装失败。

步骤1 安装Windows操作系统。

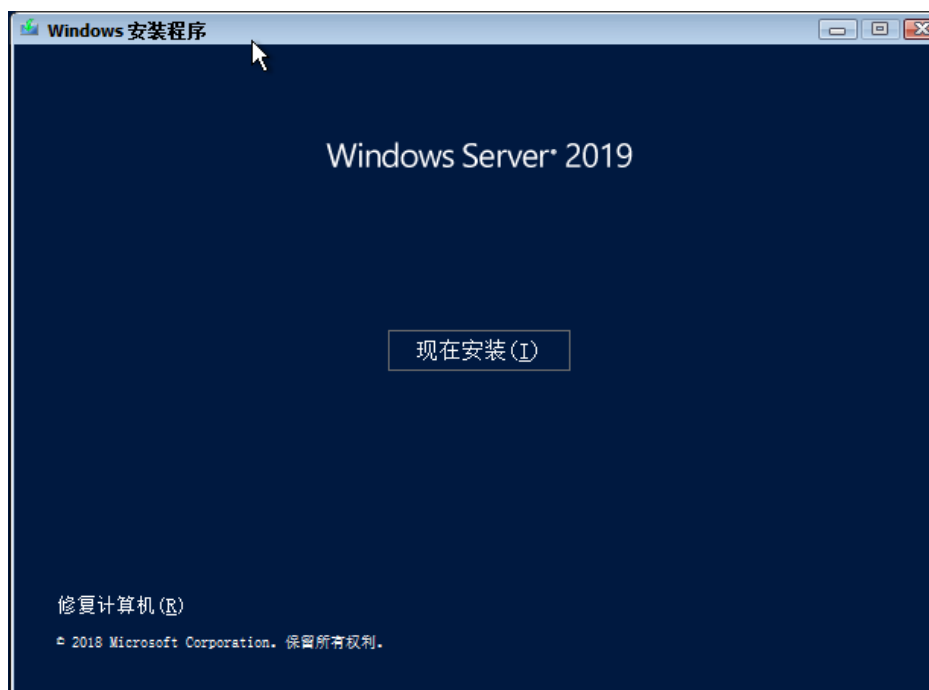
1. 在安装界面设置安装首选项，如[图3-12](#)所示。

图 3-12 安装首选项



2. 单击“下一步”。
进入安装确认界面，如[图3-13](#)所示。

图 3-13 安装确认



3. 单击“现在安装”。
进入“选择要安装的操作系统”界面。
4. 选择待安装的操作系统版本，并单击“下一步”。
进入“请阅读许可条款”界面。

- 勾选“我接受许可条款”，并单击“下一步”。
进入“您想进行何种类型的安装？”界面，如图3-14所示。

图 3-14 安装类型



- 选择“自定义（高级）”。
进入“您想将Windows安装在何处？”界面。
 - 如果系统提示未查找到驱动器，如图3-15所示，则执行步骤1.7。

图 3-15 安装位置



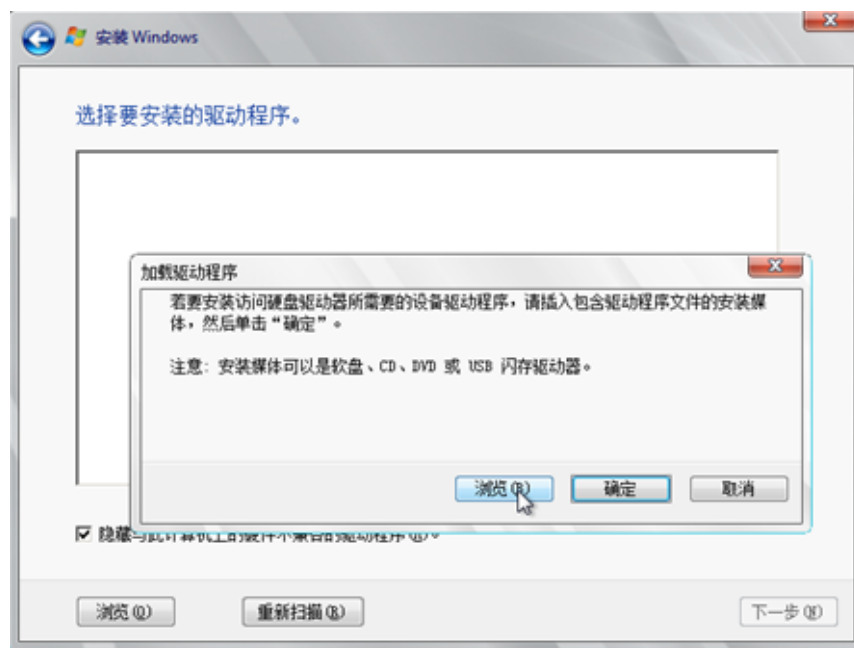
- 如果系统显示磁盘，如图3-16所示，则执行步骤1.9。

图 3-16 安装位置



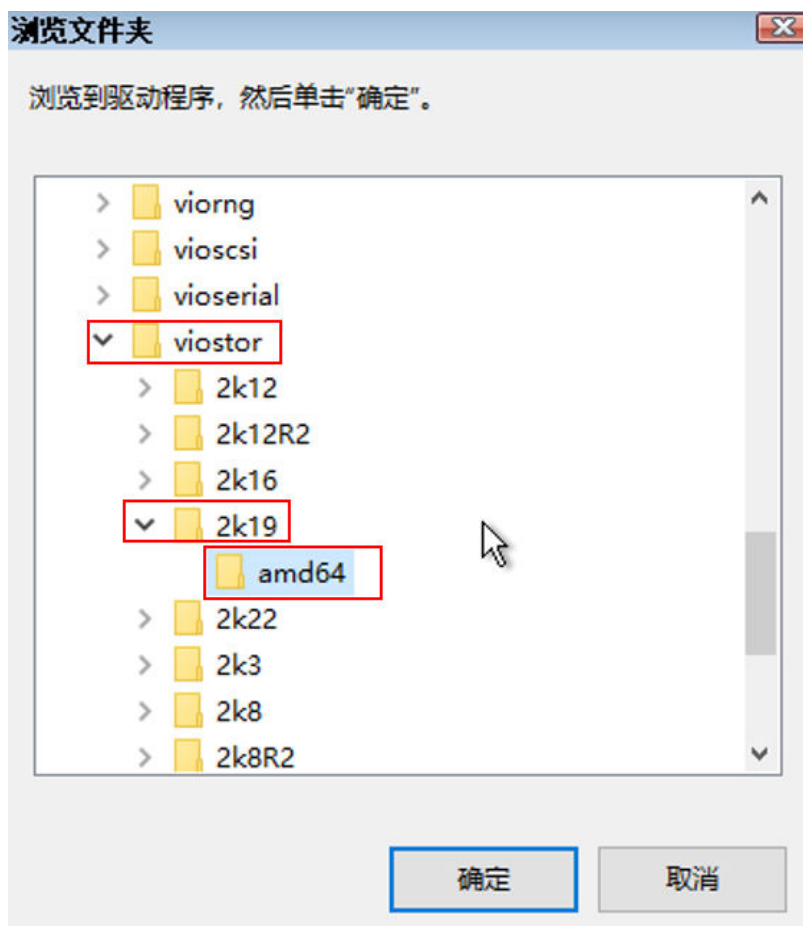
7. 依次单击“加载驱动程序 > 浏览”。

图 3-17 浏览



8. 根据磁盘类型不同，选择对应的驱动。
 - 如果磁盘类型为VBD，选择“viostor> 2k19 > amd64”，单击“确定”。

图 3-18 选择 viostor 驱动文件夹



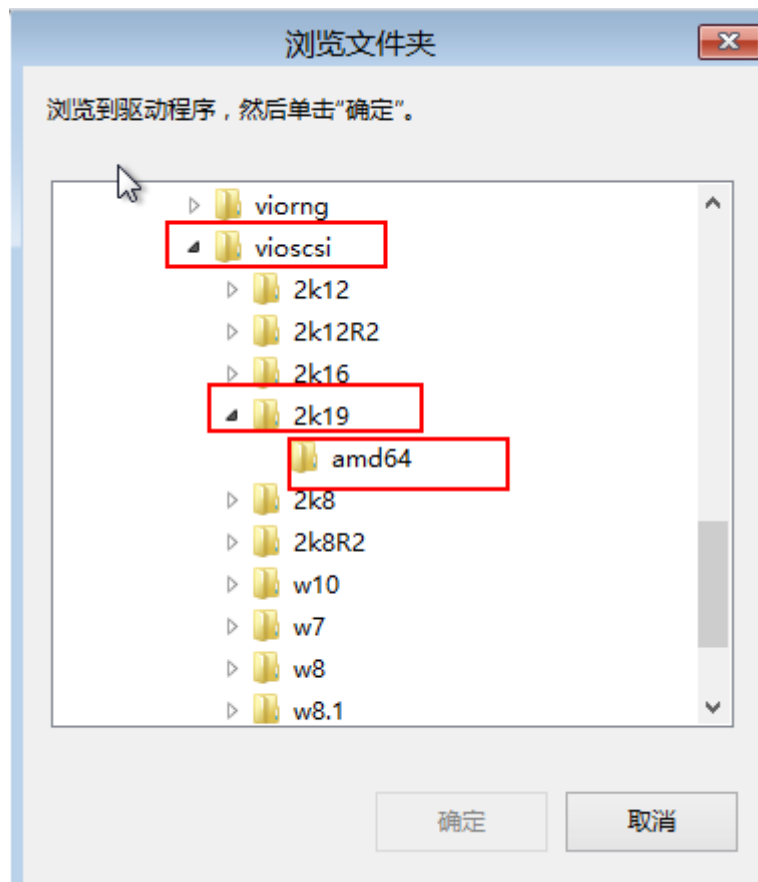
选择viostor.inf驱动并单击“下一步”。

图 3-19 选择要安装的驱动程序 viostor.inf



- 如果磁盘类型为SCSI，选择“vioscsi > 2k19 > amd64”，单击“确定”。

图 3-20 选择 vioscsi 驱动文件夹



选择vioscsi.inf驱动并单击“下一步”。

图 3-21 选择要安装的驱动程序 vioscsi.inf



9. 选择磁盘，并单击“下一步”。

图 3-22 安装位置



说明

如果磁盘类型为“脱机”状态，如图3-23所示，需先关机并开机云服务器，然后再重新安装操作系统。

图 3-23 磁盘脱机



10. 进入“正在安装Windows”界面，开始安装操作系统，如图3-24所示。
安装过程大约耗时50分钟，过程中云服务器会自动重启，请重新登录弹性云服务器并按照提示完成操作系统的后续配置。

说明

后续配置中，需要设置操作系统用户的密码。
支持的特殊字符包括：!@\$%^&_+[{]~;./?

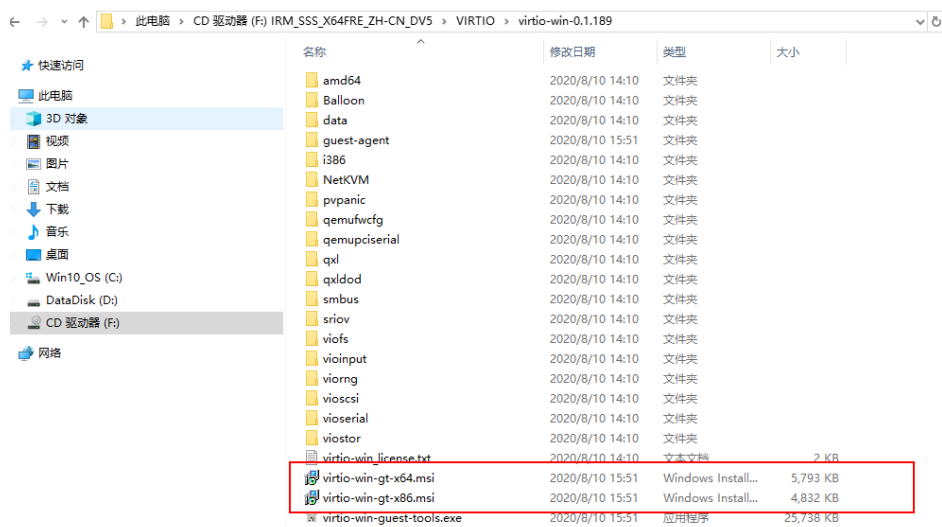
图 3-24 安装进度



步骤2 安装相关驱动。

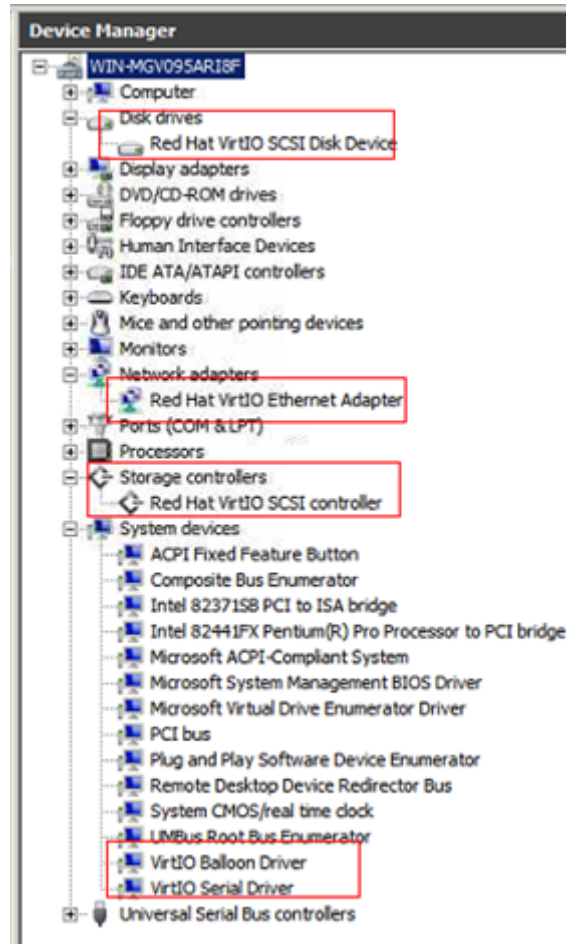
1. 打开“计算机”，并双击驱动器，如图3-25所示。

图 3-25 打开驱动器



2. 双击virtio-win-gt-x64或virtio-win-gt-x86驱动，根据界面提示进行驱动的安装。
3. 驱动安装完成后，进入“设备管理器”，检查红框标注的驱动是否均安装成功。

图 3-26 查看设备管理器



----结束

3.11.6 配置云服务器并创建 Windows 系统盘镜像

操作场景

安装完操作系统后的临时云服务器还需要进行相关配置，并安装云平台提供的Guest OS driver，才能保证后续创建的云服务器正常使用。

说明

Guest OS driver包括virtio驱动和PV driver，在前面步骤中已为云服务器安装virtio驱动，因此本节只需要安装PV driver即可。

该任务指导用户完成Windows云服务器的相关配置与驱动安装，从而创建为Windows系统盘镜像。

操作步骤

步骤1 配置云服务器。

1. 检查网卡属性是否为DHCP。如果云服务器网络配置为静态IP地址，请参考[设置网卡属性为DHCP \(Windows\)](#)修改为DHCP方式。

2. 对于需要使用Windows远程桌面连接方式进行访问的云服务器，需要开启远程桌面连接功能，请参考[开启远程桌面连接功能](#)。
3. 安装PV driver，请参考[安装PV driver](#)。
安装完以后需要清除系统日志，请参考[清除系统日志](#)。
4. 安装并配置Cloudbase-Init，保证镜像创建的新云服务器可以使用控制台的“用户数据注入”功能注入初始化自定义信息（例如为云服务器设置登录密码）。请参考[安装并配置Cloudbase-Init工具](#)。
5. （可选）配置增值功能。
 - 开启网卡多队列，请参考[如何设置镜像的网卡多队列属性？](#)。
 - 配置IPv6地址，请参考[如何配置IPv6地址](#)。

步骤2 关机云服务器，使上述配置生效。

步骤3 创建Windows系统盘镜像。

具体操作请参考[通过云服务器创建Windows系统盘镜像](#)。

----结束

后续操作

系统盘镜像创建成功后，请及时删除临时云服务器，避免继续占用计算资源空间。

3.12 通过 ISO 文件创建 Linux 系统盘镜像

3.12.1 创建流程（Linux）

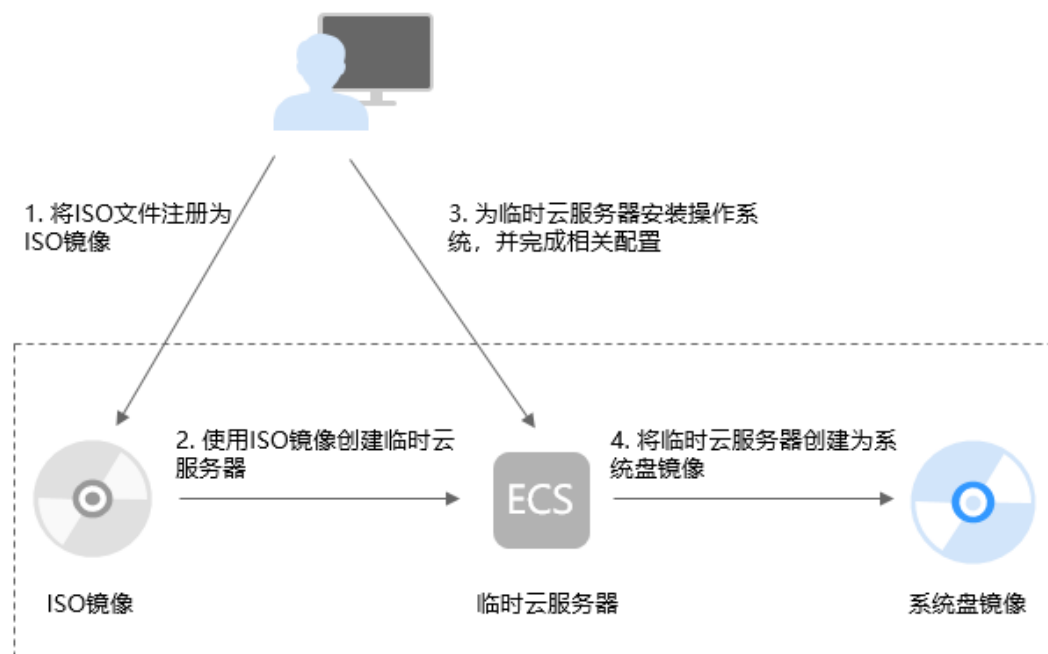
ISO是一种光盘映像文件，通过特定的压缩方式，将大量的数据文件统一为一个后缀名为iso的映像文件。ISO文件可以理解为从光盘中复制出来的数据文件，所以ISO文件无法直接使用，需要利用一些工具进行解压后才能使用。例如使用虚拟光驱打开，或者将ISO文件刻录到光盘中后，使用光驱来进行读取。

本章介绍通过ISO文件创建Linux系统盘镜像的操作过程。

创建过程

通过ISO文件创建Linux系统盘镜像的流程如[图3-27](#)所示。

图 3-27 创建过程（Linux 系统盘镜像）



步骤说明如下：

1. 将ISO文件注册为ISO镜像

在管理控制台上将准备好的ISO文件注册为镜像。该镜像为ISO镜像，不能直接用于发放云服务器。详情请参考[将ISO文件注册为镜像（Linux）](#)。

2. 使用ISO镜像创建临时云服务器

使用注册成功的ISO镜像创建一个临时云服务器，该云服务器为空云服务器，即没有操作系统，也未安装任何驱动。详情请参考[使用ISO镜像创建Linux云服务器](#)。

3. 为临时云服务器安装操作系统和驱动，并完成相关配置

包含如下操作：安装操作系统、安装原生的XEN和KVM驱动、配置网卡属性、清理网络规则文件等。详情请参考[安装Linux操作系统和配置云服务器并创建Linux系统盘镜像](#)中的步骤步骤1。

4. 将临时云服务器创建为系统盘镜像

在管理控制台上将已完成安装和配置的临时云服务器创建为系统盘镜像。镜像创建成功后，删除临时云服务器避免继续占用计算资源空间。详情请参考[通过云服务器创建Linux系统盘镜像](#)。

约束与限制

- 使用ISO文件注册的ISO镜像无法在创建云服务器页面选择到，也无法用来切换操作系统，必须完成操作系统及驱动安装做成系统盘镜像后才能使用。
- ISO镜像在镜像服务页面创建的临时云服务器只建议用来装机，部分功能受限。如：不支持挂载磁盘操作。

3.12.2 将 ISO 文件注册为镜像（Linux）

操作场景

通过注册镜像操作，将外部镜像ISO文件注册为云平台的私有镜像，即ISO镜像。注册镜像前，需先将ISO文件上传到OBS桶中。

使用ISO文件注册的ISO镜像不支持复制、导出和加密。

前提条件

- 待注册文件需为iso格式。
- 已上传ISO镜像文件到OBS的个人桶中。具体操作请参见[上传镜像文件（Linux）](#)。

说明

ISO镜像文件名称只能包含英文字母、数字、中划线（-）和下划线（_）。如果不符合要求，请先修改名称再上传至OBS桶。

操作步骤

步骤1 登录IMS控制台。

1. 登录管理控制台。
 2. 选择“计算 > 镜像服务”。
- 进入镜像服务页面。

步骤2 创建ISO镜像。

1. 单击右上角的“创建私有镜像”，进入创建私有镜像页面。
2. 在“镜像类型和来源”区域，选择镜像的创建方式为“ISO镜像”。
3. 从镜像文件列表中先选择保存ISO镜像文件的桶，再选择对应的ISO文件。
4. 在“配置信息”区域，填写镜像的基本信息。
 - 架构类型：根据待注册的镜像文件的架构类型，选择“x86”或“ARM”。

说明

该参数仅在支持ARM规格的区域需要配置。

- 启动方式：取值为“BIOS”和“UEFI”。请确保选择的启动方式与镜像文件中的启动方式一致，否则，使用该镜像创建的弹性云服务器无法启动。ARM架构类型不支持BIOS启动方式。
- 操作系统：选择ISO镜像文件对应的操作系统。为保证镜像的正常创建和使用，请确保选择的操作系统与镜像文件的操作系统类型一致。
- 系统盘：设置系统盘容量，范围为40-1024GB，要求不小于镜像文件的系统盘大小。
- 名称：输入镜像的名称。
- 描述：可选配置，根据需要输入描述信息。

说明

以上为全量参数，不同企业、单位支持情况不同，以实际界面为准。

5. 单击“立即创建”。
6. 确认镜像参数，单击“提交申请”。

步骤3 返回私有镜像界面，查看创建的ISO镜像的状态。

当镜像的状态为“正常”时，表示创建成功。

----结束

3.12.3 使用 ISO 镜像创建 Linux 云服务器

操作场景

该任务指导用户使用已注册成功的ISO镜像创建弹性云服务器。

约束与限制

暂不支持专属云用户通过ISO镜像创建弹性云服务器。

用户在指定区域申请了专属云服务，那么该用户就是专属云用户。

操作步骤

步骤1 登录IMS控制台。

1. 登录管理控制台。
 2. 选择“计算 > 镜像服务”。
- 进入镜像服务页面。

步骤2 创建弹性云服务器。

1. 单击“私有镜像”页签，在ISO镜像所在行的“操作”列下，单击“安装服务器”，创建云服务器。

说明

对于**专属云**用户，不支持使用ISO镜像创建临时云服务器操作，此时“安装服务器”按钮灰化。

2. 根据界面提示完成云服务器的配置，并单击“确定”。

----结束

后续操作

弹性云服务器创建成功后，使用平台提供的“远程登录”方式，登录弹性云服务器，继续执行后续的安装操作系统和相关驱动操作。

3.12.4 安装 Linux 操作系统

操作场景

该任务以“CentOS 7 64位”操作系统为例，指导用户安装Linux操作系统。

由于镜像文件不同，安装步骤稍有不同，请根据实际的安装界面提示进行操作。

说明

请根据实际情况完成时区和Repo源更新地址，输入法，语言等相关配置。

前提条件

已使用平台提供的“远程登录”方式（即VNC登录），连接云服务器并进入安装界面。

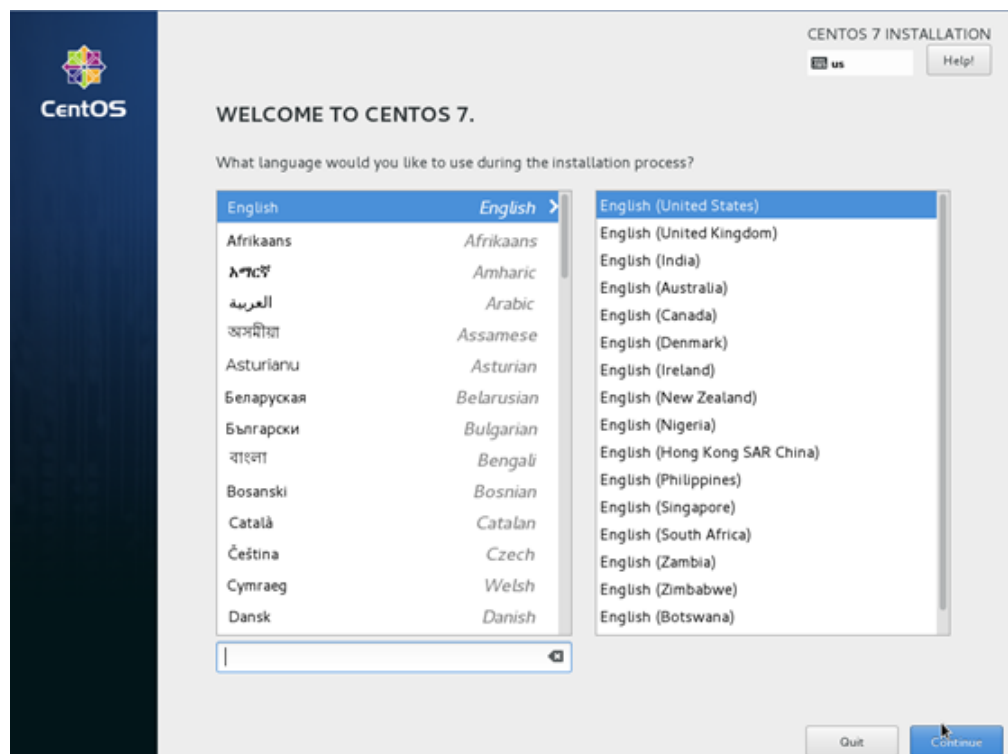
操作步骤

注意

操作系统安装成功前，禁止关闭或重启弹性云服务器，否则会导致操作系统安装失败。

1. 在安装界面设置语言项，并单击“Continue”。

图 3-28 安装界面



2. 在“安装信息摘要”界面，选择“系统 > 安装位置(D)”。

图 3-29 安装信息摘要



3. 选择待安装系统的磁盘设备，并单击“完成”。

图 3-30 安装目标位置



4. 单击“开始安装(B)”。

图 3-31 开始安装



5. 等待系统自动安装，待进度条为100%时，CentOS安装完成。

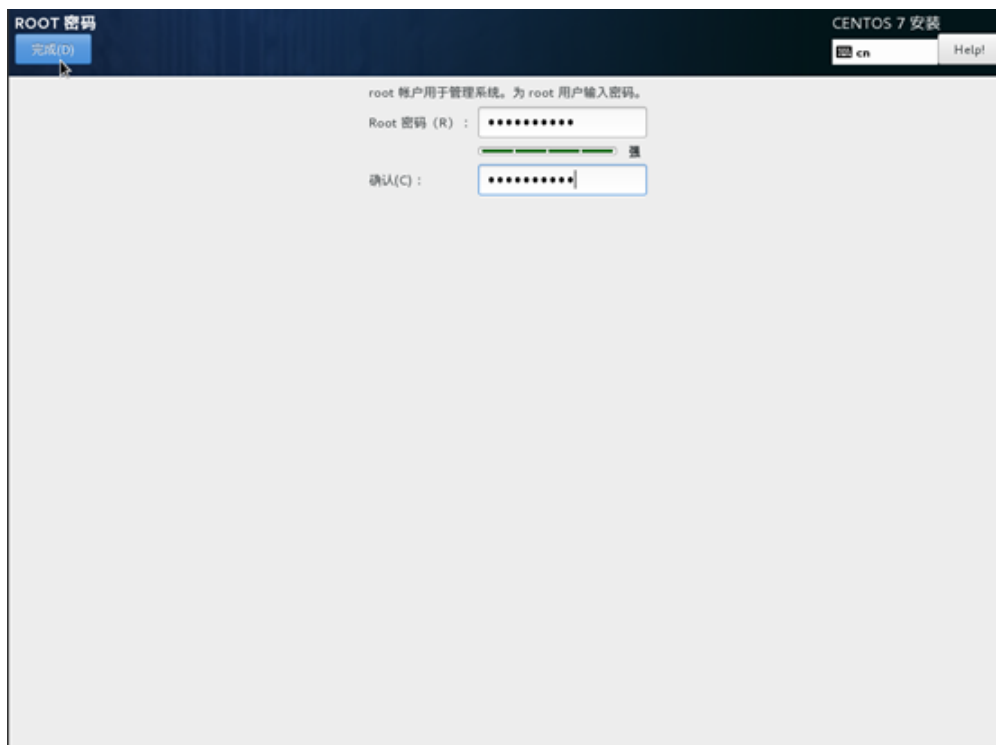
图 3-32 安装完成



6. 单击“用户设置”栏的“ROOT密码”。
系统进入“ROOT密码”页面。

7. 根据界面提示，为root用户设置密码，并单击“完成”。

图 3-33 设置 root 密码



8. 单击“完成配置”。

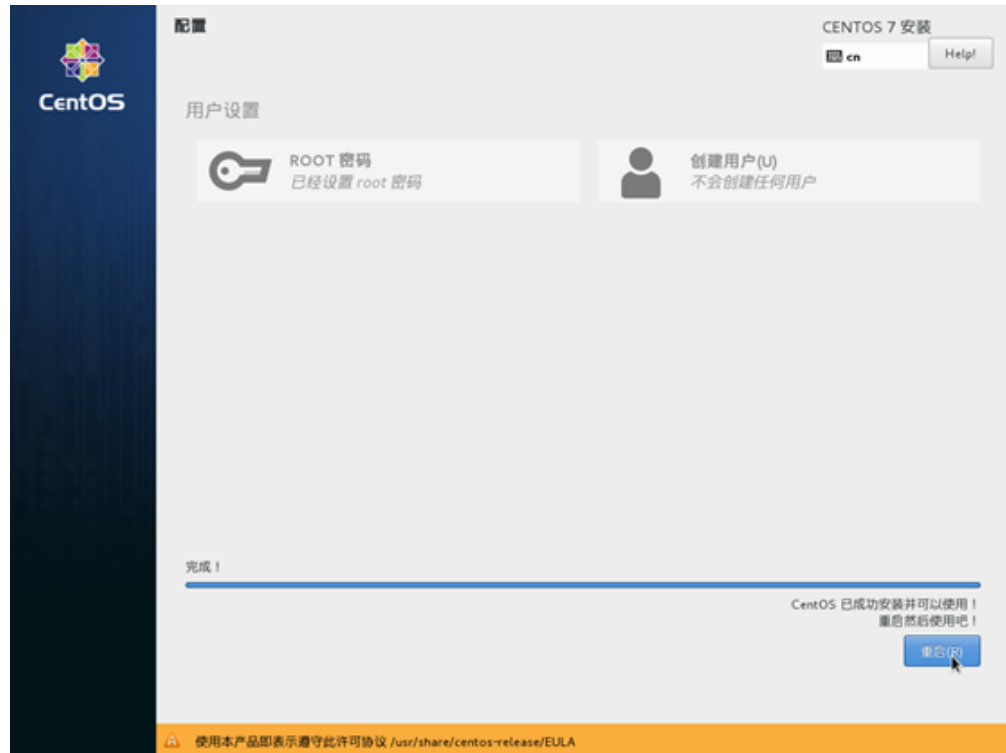
图 3-34 完成配置



9. 单击“重启”。

重启后，如果系统再次提示您安装操作系统，此时，请退出当前VNC登录界面，并使用控制台提供的“关机”、“开机”按钮，重启弹性云服务器。

图 3-35 重启



3.12.5 配置云服务器并创建 Linux 系统盘镜像

操作场景

安装完操作系统后的临时云服务器需要进行相关配置，并安装原生的XEN和KVM驱动，才能保证后续创建的云服务器正常使用。

该任务指导用户完成Linux云服务器的相关配置与驱动安装，从而创建为Linux系统盘镜像。

操作步骤

步骤1 配置云服务器。

1. 配置网络

- 检查云服务器的私有IP是否和控制台显示的私有IP一致（可以通过**ifconfig**查看）。如果不一致，请参考[清理网络规则文件](#)。
- 检查网卡属性是否为DHCP。如果云服务器网络配置为静态IP地址，请参考[设置网卡属性为DHCP（Linux）](#)。
- 检查SSH服务是否为开启状态（可以通过**service sshd status**查看）。如果未开启，请执行**service sshd start**。请确保您的云服务器防火墙（例如：Linux iptables）允许访问SSH。

2. 安装驱动

为了保证镜像创建的新云服务器的网络性能以及基本功能正常，必须在创建镜像时使用的云服务器中安装原生XEN和KVM驱动。在安装原生XEN和KVM驱动前，需要先卸载PV driver。

📖 说明

建议您禁用任何防病毒软件或入侵检测软件，安装原生的XEN和KVM驱动完成后，您可以再次启用这些软件。

- 卸载PV driver，请参考[在Linux系统中卸载PV driver](#)。
- 安装原生的XEN和KVM驱动，请参考[怎样安装原生的XEN和KVM驱动](#)。
- 驱动安装完成后需要清除日志文件、历史记录等，请参考[清除日志文件](#)。

3. 配置文件系统

- 修改grub文件的磁盘标识方式为UUID，请参考[修改grub文件磁盘标识方式为UUID](#)。
- 修改fstab文件的磁盘标识方式为UUID，请参考[修改fstab文件磁盘标识方式为UUID](#)。
- 清除“/etc/fstab”中非系统盘的自动挂载信息，避免对后续挂载数据盘可能带来影响。请参考[卸载云服务器的数据盘](#)。

4. （可选）配置增值功能

- 安装并配置Cloud-Init，请参考[安装Cloud-Init工具](#)和[配置Cloud-Init工具](#)。
- 开启网卡多队列，请参考[如何设置镜像的网卡多队列属性？](#)。
- 配置IPv6地址，请参考[如何配置IPv6地址](#)。

步骤2 创建Linux系统盘镜像。

具体操作请参考[通过云服务器创建Linux系统盘镜像](#)。

----结束

后续操作

系统盘镜像创建成功后，请及时删除临时云服务器，避免继续占用计算资源空间。

3.13 快速导入镜像文件

3.13.1 流程概览

如果外部镜像文件大小大于128GB，可以使用快速导入功能进行导入。

约束与限制

- 镜像文件仅支持raw和zvhd2格式。
- 镜像文件大小不能超过1TB。

导入方案

请根据您的镜像文件格式来选择合适的导入方案：

- 若文件格式为**zvhd2**，则导入方案为：镜像文件优化 > 上传至OBS桶 > 在云平台注册
- 若文件格式为**raw**，则导入方案为：镜像文件优化 > 生成镜像文件的位表文件 > 将镜像文件和位表文件一同上传至OBS桶 > 在云平台注册
- 若文件为**zvhd2**和**raw**以外的格式，导入时区分以下两种情况：
 - 镜像文件优化 > 转换镜像文件格式为zvhd2 > 将镜像文件上传至OBS桶 > 在云平台注册
 - 镜像文件优化 > 转换镜像文件格式为raw，并生成镜像文件的位表文件 > 将镜像文件和位表文件一同上传至OBS桶 > 在云平台注册

📖 说明

- 快速导入也称大文件导入。大文件在导入时依赖延迟加载特性，zvhd2格式已支持该特性；raw格式不支持，需要依赖位表文件，因此在上传时，需要连同位表文件一起上传。
- 镜像文件优化的指导参考：[优化过程（Windows）](#) 或者 [优化过程（Linux）](#)，根据镜像文件的操作系统类型来选择。

导入流程

本节以外部镜像文件格式为**zvhd2**和**raw**以外的场景为例，介绍完整的导入流程。

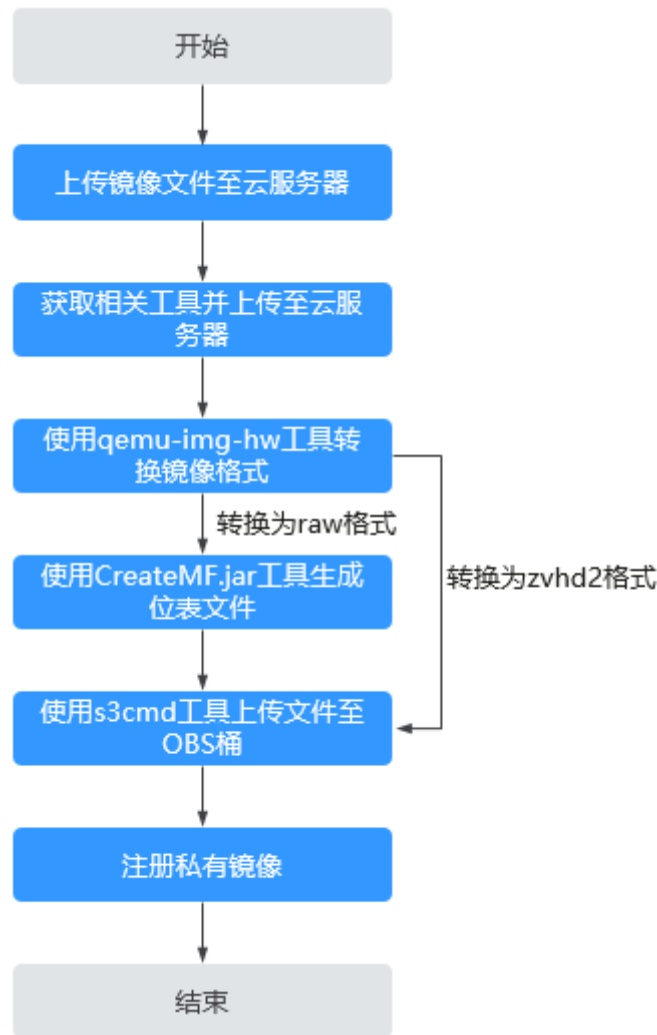
转换镜像格式时，可以使用qemu-img-hw，也可以使用开源工具qemu-img。qemu-img-hw仅适用于Linux操作系统，因此本节提供了基于Linux和Windows操作系统环境的导入指导。

📖 说明

快速导入工具中包含两个子工具，分别实现两项功能：qemu-img-hw（镜像格式转换）、CreateMF.jar（生成位表文件）。

- 基于Linux操作系统环境
推荐使用云平台的EulerOS云服务器作为操作的环境，流程如[图3-36](#)所示。

图 3-36 导入流程（基于 Linux 操作系统环境）



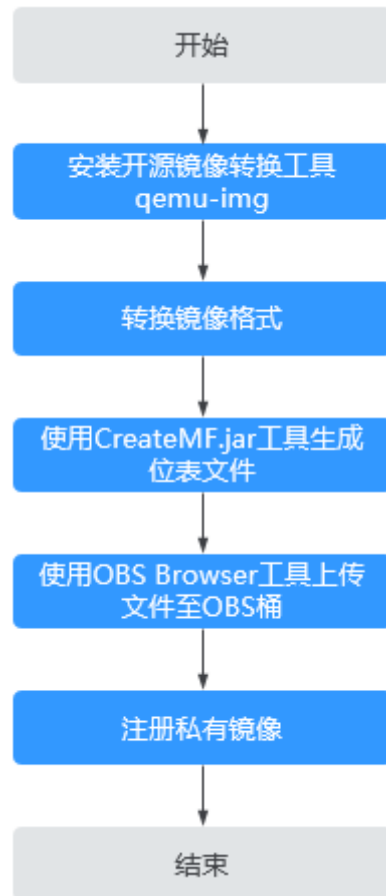
详细操作请参见[快速导入镜像文件（基于Linux环境）](#)。

- 基于Windows操作系统环境
推荐使用Windows系统本地PC作为操作的环境，流程如[图3-37](#)所示。

📖 说明

开源镜像转换工具qemu-img不支持将镜像文件转换为zvhd2格式，因此需要转换为raw格式，再使用CreateMF.jar生成位表文件。

图 3-37 导入流程（基于 Windows 操作系统环境）



详细操作请参见[快速导入镜像文件（基于Windows环境）](#)。

3.13.2 快速导入镜像文件（基于 Linux 环境）

操作场景

本节指导您基于Linux操作系统环境完成镜像文件快速导入，推荐使用云平台的EulerOS云服务器作为转换镜像格式和生成位表文件的环境。

Linux操作系统环境下，建议使用qemu-img-hw工具进行镜像格式转换。

前提条件

- 已完成镜像文件优化，详细操作请参考[优化过程（Windows）](#)或[优化过程（Linux）](#)；同时需要确保镜像文件符合[表3-6](#)或[表3-10](#)中的限制条件。

📖 说明

根据镜像文件的操作系统类型来选择所参考内容。

- 已在管理控制台创建EulerOS操作系统的弹性云服务器，并确保云服务器已绑定弹性公网IP。
- 已在管理控制台创建OBS桶。

操作步骤

步骤1 上传镜像文件至云服务器。

- 本地主机为Linux系统：

通过scp命令将镜像文件上传至云服务器。以将“image01.qcow2”文件上传至云服务器的“/usr/”目录下为例。

```
scp /var/image01.qcow2 root@xxx.xxx.xx.xxx:/usr/
```

其中，xxx.xxx.xx.xxx为云服务器的弹性公网IP。

- 本地主机为Windows系统：

使用文件传输工具（例如WinSCP）将镜像文件上传至云服务器。

步骤2 获取快速导入工具并上传至云服务器，然后解压工具包。

请联系管理员获取快速导入工具。

步骤3 使用qemu-img-hw工具转换镜像格式。

1. 进入qemu-img-hw存放目录，以存放在“/usr/quick-import-tools/qemu-img-hw”为例：

```
cd /usr/quick-import-tools/qemu-img-hw
```

2. 执行以下命令修改权限：

```
chmod +x qemu-img-hw
```

3. 执行qemu-img-hw命令将镜像文件转为zvhd2或raw格式（推荐转为zvhd2格式）。

qemu-img-hw命令格式：

```
./qemu-img-hw convert -p -O {目标镜像格式} {待转换镜像文件} {目标镜像文件}
```

以将“image01.qcow2”格式文件转换成“image01.zvhd2”格式为例：

```
./qemu-img-hw convert -p -O zvhd2 image01.qcow2 image01.zvhd2
```

– 若转换后为zvhd2格式文件，请执行[步骤5](#)。

– 若转换后为raw格式文件，请使用CreateMF.jar工具，生成raw格式镜像文件的位表文件。请执行[步骤4](#)。

步骤4 使用CreateMF.jar工具生成位表文件。

1. 请确保当前云服务器已安装jdk。

执行以下命令查看是否已安装jdk：

```
source /etc/profile
```

```
java -version
```

如果回显信息中显示java版本信息，证明当前云服务器已安装jdk。

2. 执行以下命令进入CreateMF.jar程序所在的目录。

```
cd /usr/quick-import-tools/createMF
```

3. 执行以下命令生成位表文件。

```
java -jar CreateMF.jar /原raw文件路径 /生成的mf文件路径
```

例如：

```
java -jar CreateMF.jar image01.raw image01.mf
```

注意

- 生成的.mf位表文件和raw格式镜像文件必须是相同名称。例如镜像文件名称为：image01.raw，那么生成的位表文件名称为：image01.mf。

步骤5 使用s3cmd工具上传文件至OBS桶。

1. 安装s3cmd工具。

若云服务器已安装该工具可跳过此步骤，直接进行第二步配置s3cmd：

- a. 执行以下命令，安装setuptools。

```
yum install python-setuptools
```

- b. 执行以下命令，安装wget。

```
yum install wget
```

- c. 执行以下命令，获取s3cmd软件包。

```
wget https://github.com/s3tools/s3cmd/archive/master.zip
mv master.zip s3cmd-master.zip
```

- d. 执行以下命令，安装s3cmd。

```
unzip s3cmd-master.zip
cd s3cmd-master
python setup.py install
```

2. 配置s3cmd工具。

执行以下命令配置s3cmd工具。

```
s3cmd --configure
Access Key: 输入AK
Secret Key: 输入SK
Default Region: 输入所在Region
S3 Endpoint: 可参考OBS的Endpoint
DNS-style bucket+hostname:port template for accessing a bucket: 输入带桶名的Server地址, 例如
mybucket.obs.myclouds.com
Encryption password: 不填, 直接按回车
Path to GPG program: 不填, 直接按回车
Use HTTPS protocol: 是否使用HTTPS, Yes/No
HTTP Proxy server name: 代理地址, 连接云平台需要连外网 (如不需要直接回车)
HTTP Proxy server port: 代理端口, 连接云平台需要连外网 (如不需要直接回车)
Test access with supplied credentials? y
(如果显示: Success. Your access key and secret key worked fine :-)) 则表示连接成功)
Save settings? y (是否保存配置)
```

说明

配置完成后，信息会保存在用户目录下“/root/.s3cfg”，如果还想修改，可以重新执行s3cmd --configure命令，或者直接编辑.s3cfg文件（执行vi .s3cfg命令进行编辑）。

3. 使用s3cmd命令上传zvhd2格式镜像文件到OBS桶，或者上传raw格式镜像文件及其位表文件到OBS桶。

```
s3cmd put image01.zvhd2 s3://mybucket/
```

注意

.mf位表文件必须和其对应的raw格式镜像文件在同一个OBS桶中。

步骤6 注册私有镜像。

您可以通过控制台方式、API接口方式，将转换后的zvhd2或raw文件注册为私有镜像。

方法一：通过控制台创建私有镜像

1. 登录IMS控制台。
 - a. 登录管理控制台。
 - b. 选择“计算 > 镜像服务”。
进入镜像服务页面。
2. 单击右上角的“创建私有镜像”。
3. 在“镜像类型和来源”区域，选择镜像的创建方式为“系统盘镜像”或“数据盘镜像”。
4. 镜像源选择“镜像文件”，从列表中选择转换为zvhd2或raw文件的桶，再选择转换后的镜像文件。如果是raw格式镜像文件，需要确保同名位表文件已上传。
5. 勾选快速通道栏的“开启快速通道”，并确认已优化镜像文件，然后勾选“镜像文件准备”栏的内容。
6. 根据界面提示填写配置信息。
具体的配置参数说明，请参见[注册镜像（Linux）](#)。

注意

- 操作系统必须要和镜像文件所含的操作系统一致。
- 系统盘大小必须大于镜像文件的大小。

通过qemu-img-hw工具可查询镜像文件大小：

```
qemu-img-hw info test.zvhd2
```

方法二：通过API方式创建私有镜像

您可以通过接口POST /v2/cloudimages/quickimport/action，实现镜像文件快速导入功能。

该接口的具体调用方法，请参见《[镜像服务接口参考](#)》的“[镜像文件快速导入](#)”章节。

----结束

附 1：qemu-img-hw 常用命令

- 镜像文件格式转换：`qemu-img-hw convert -p -O {目标镜像格式} {待转换镜像文件} {目标镜像文件}`

上述命令中各参数对应的说明如下：

-p: 标识转换的进度条

-O: （必须是大写）后面的参数为转换出来的镜像格式 + 源镜像文件名称 + 目标镜像文件名称

示例：将qcow2格式转为zvhd2格式

```
qemu-img-hw convert -p -O zvhd2 test.qcow2 test.zvhd2
```

- 查询镜像文件信息：`qemu-img-hw info {镜像文件}`
示例：`qemu-img-hw info test.zvhd2`
- 查看帮助：`qemu-img-hw -help`

附 2：执行 `qemu-img-hw` 常见报错

- 问题描述：

执行`qemu-img-hw`命令时回显信息如下：

```
./qemu-img-hw: /lib64/libc.so.6: version `GLIBC_2.14' not found (required by ./qemu-img-hw)
```

解决方法：

执行`strings /lib64/libc.so.6 | grep GLIBC`查看GLIBC版本，若由于版本过低造成，可安装高版本即可。依次执行下述命令：

```
wget http://ftp.gnu.org/gnu/glibc/glibc-2.15.tar.gz
```

```
wget http://ftp.gnu.org/gnu/glibc/glibc-ports-2.15.tar.gz
```

```
tar -xvf glibc-2.15.tar.gz
```

```
tar -xvf glibc-ports-2.15.tar.gz
```

```
mv glibc-ports-2.15 glibc-2.15/ports
```

```
mkdir glibc-build-2.15
```

```
cd glibc-build-2.15
```

```
../glibc-2.15/configure --prefix=/usr --disable-profile --enable-add-ons --with-headers=/usr/include --with-binutils=/usr/bin
```

📖 说明

此命令若报错“`configure: error: no acceptable C compiler found in $PATH`”，请先执行：`yum -y install gcc`

```
make
```

```
make install
```

- 问题描述：

执行`qemu-img-hw`命令时回显信息如下：

```
./qemu-img-hw: error while loading shared libraries: libaio.so.1: cannot open shared object file: No such file or directory
```

解决方法：请先执行命令`yum install libaio`

3.13.3 快速导入镜像文件（基于 Windows 环境）

操作场景

本节指导您基于Windows操作系统环境完成镜像文件快速导入，推荐使用Windows系统本地PC作为转换镜像格式和生成位表文件的环境。

Windows操作系统环境下，必须使用开源`qemu-img`工具进行镜像格式转换，`qemu-img`支持`vhd`、`vmdk`、`qcow2`、`raw`、`vhdx`、`qcow`、`vdi`或`qed`格式的镜像的相互转换，因此需要转换为`raw`格式，再使用`CreateMF.jar`工具生成位表文件。

前提条件

- 已完成镜像文件优化，详细操作请参考[优化过程（Windows）](#)或[优化过程（Linux）](#)；同时需要确保镜像文件符合[表3-6](#)或[表3-10](#)中的限制条件。

📖 说明

根据镜像文件的操作系统类型来选择所参考内容。

- 已在管理控制台创建OBS桶，并下载OBS Browser+工具。

操作步骤

步骤1 安装开源镜像转换工具qemu-img。

步骤2 通过cmd命令进入qemu-img安装目录，执行qemu-img命令将镜像文件转换为raw格式。

以将“image.qcow2”格式文件转换成“image.raw”格式为例：

```
qemu-img convert -p -O raw image.qcow2 image.raw
```

步骤3 使用CreateMF.jar工具生成位表文件。

1. 获取工具并解压。
请联系管理员获取工具。
2. 请确保当前环境已安装jdk。
可通过如下方法验证：
打开cmd.exe，执行**java -version**，若显示java版本信息，证明已安装jdk。
3. 进入CreateMF.jar程序所在的路径。
以将CreateMF.jar程序下载到“D:/test”目录为例，执行以下命令进入CreateMF.jar程序所在的路径：

D:

```
cd test
```

4. 执行以下命令，生成raw格式镜像文件对应的位表文件。
java -jar CreateMF.jar D:/image01.raw D:/image01.mf

⚠️ 注意

- 生成的.mf位表文件和raw格式镜像文件必须是相同名称。例如镜像文件名称为：image01.raw，那么生成的位表文件名称为：image01.mf。
-

步骤4 使用OBS Browser+工具上传文件至OBS桶。

必须将raw格式镜像文件及其位表文件上传至同一OBS桶。

步骤5 注册私有镜像。

您可以通过控制台方式、API接口方式，将转换后的zvhd2或raw文件注册为私有镜像。

方法一：通过控制台创建私有镜像

1. 登录IMS控制台。
 - a. 登录管理控制台。
 - b. 选择“计算 > 镜像服务”。
进入镜像服务页面。

2. 单击右上角的“创建私有镜像”。
3. 在“镜像类型和来源”区域，选择镜像的创建方式为“系统盘镜像”或“数据盘镜像”。
4. 镜像源选择“镜像文件”，从列表中选择转换为zvhd2或raw文件的桶，再选择转换后的镜像文件。如果是raw格式镜像文件，需要确保同名位表文件已上传。
5. 勾选快速通道栏的“开启快速通道”，并确认已优化镜像文件，然后勾选“镜像文件准备”栏的内容。
6. 根据界面提示填写配置信息。
具体的配置参数说明，请参见[注册镜像（Linux）](#)。

注意

- 操作系统必须要和镜像文件所含的操作系统一致。
- 系统盘大小必须大于镜像文件的大小。

通过qemu-img-hw工具可查询镜像文件大小：

```
qemu-img-hw info test.zvhd2
```

方法二：通过API方式创建私有镜像

您可以通过接口POST /v2/cloudimages/quickimport/action，实现镜像文件快速导入功能。

该接口的具体调用方法，请参见《[镜像服务接口参考](#)》的“[镜像文件快速导入](#)”章节。

----**结束**

4 管理私有镜像

4.1 通过镜像创建云服务器

操作场景

您可以使用公共镜像、私有镜像或共享镜像创建云服务器。使用公共镜像和私有镜像创建云服务器的区别是：

- 公共镜像：创建的云服务器包含所需操作系统和预装的公共应用，需要您自行安装应用软件。
- 私有镜像或共享镜像：创建的云服务器包含操作系统、预装的公共应用以及用户的私有应用。

操作步骤

1. 登录IMS控制台。
 - a. 登录管理控制台。
 - b. 选择“计算 > 镜像服务”。进入镜像服务页面。
2. 单击“公共镜像”、“私有镜像”或“共享镜像”页签进入对应的镜像列表。
3. 在镜像所在行的“操作”列下，单击“申请服务器”。
4. 创建云服务器的详细操作请参见《弹性云服务器用户指南》。

使用系统盘私有镜像创建云服务器时，云服务器的规格可以重新设置，系统盘的类型也可以重新选择，但系统盘的容量只能比镜像的系统盘大。

使用整机私有镜像创建云服务器时，如果整机镜像中包含了一块或多块数据盘，系统会自动设置好数据盘参数。您可以增加系统盘和数据盘容量，但不能缩小。

说明

如果整机镜像中包含多块数据盘，需要加载一段时间才能正常显示系统盘信息，请耐心等待。

4.2 修改镜像属性

操作场景

为了方便您管理私有镜像，您可以根据需要修改私有镜像的如下信息：

- 镜像的名称
- 描述信息
- 最小内存
- 最大内存
- 是否支持网卡多队列
开启网卡多队列功能可以将网卡中断分散给不同的CPU处理，实现负载均衡。了解更多信息，请参见[如何设置镜像的网卡多队列属性？](#)。
- 是否支持Cloud-Init

约束与限制

- 只有状态是“正常”的私有镜像才允许用户修改属性。

操作步骤

用户可以选择以下任意一种方式修改镜像的属性。

方式1：


1. 登录IMS控制台。
 - a. 登录管理控制台。
 - b. 选择“计算 > 镜像服务”。进入镜像服务页面。
2. 单击“私有镜像”页签进入对应的镜像列表。
3. 在镜像所在行的“操作”列下，单击“修改”。
4. 在弹出的“修改镜像”对话框中，修改镜像属性。

方式2：

1. 登录IMS控制台。
 - a. 登录管理控制台。
 - b. 选择“计算 > 镜像服务”。进入镜像服务页面。
2. 单击“私有镜像”页签进入对应的镜像列表。
3. 在镜像列表中，单击镜像名称，进入镜像详情页面。
4. 在镜像详情页面单击“修改”，在弹出的“修改镜像”对话框中，修改镜像属性。

方式3：

系统支持快速修改私有镜像的名称。

1. 登录IMS控制台。
 - a. 登录管理控制台。
 - b. 选择“计算 > 镜像服务”。
进入镜像服务页面。
2. 单击“私有镜像”页签。
3. 在私有镜像列表栏，将鼠标移动至目标镜像的“名称”列。
4. 单击 ，根据界面提示，修改镜像名称。
5. 单击“确定”，新名称生效。

4.3 导出镜像

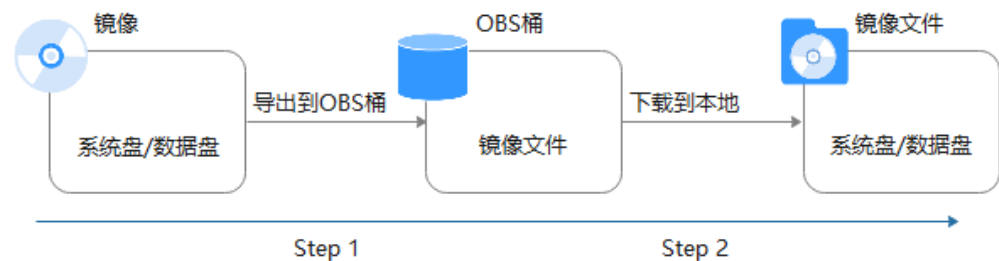
操作场景

创建私有镜像后，您可以导出镜像到OBS标准存储桶，并下载到本地使用。本文介绍导出镜像的操作步骤及相关注意事项。

背景知识

- 导出镜像的流程如下，适用于将云上的服务器系统及软件环境复制到线下集群或私有云环境使用。

图 4-1 导出镜像流程



- 导出镜像所需时间取决于私有镜像大小和当前导出任务的并发数，需要您耐心等待。
- 支持导出qcow2、vmdk、vhd和zvhd格式的镜像。创建的私有镜像默认格式为zvhd2，导出过程中，不同的格式会导致导出的镜像大小不同。


约束与限制

- 仅支持导出到存储类别为标准存储的桶，并且桶和镜像在同一区域。
- 如下类型的私有镜像不允许导出：
 - 整机镜像
 - ISO镜像
 - Windows、SUSE、Red Hat、Ubuntu、Oracle Linux公共镜像所创建的私有镜像
- 镜像大小必须小于128GB。

前提条件

- 导出镜像前请确保用户已拥有对象存储服务的Administrator权限。
- 私有镜像所在区域中已有可用的OBS桶。
如果还未创建OBS桶，请先参考《对象存储服务用户指南》进行创建，注意“存储类别”必须为“标准存储”。

操作步骤

1. 登录IMS控制台。
 - a. 登录管理控制台。
 - b. 选择“计算 > 镜像服务”。
进入镜像服务页面。
2. 在私有镜像列表中，在需要导出的镜像所在行，单击操作列的“更多 > 导出”。
3. 在“导出镜像”对话框中，填写如下参数：
 - 格式：目前支持qcow2、vmdk、vhd、zvhd格式，请根据需要进行选择。
 - 名称：输入一个方便您识别的名称。
 - 存储路径：单击  展开桶列表，选择一个用来存储镜像的OBS桶。

说明

仅支持导出到存储类别为标准存储的桶，并且桶和镜像在同一区域，否则您无法在此列表中查找到您创建的桶。

4. 单击“确定”。
在私有镜像列表上方可以查看镜像导出任务进程。

后续步骤

镜像导出成功后，您可以在对象存储服务控制台，或者客户端的指定OBS桶中下载镜像文件。

4.4 导出镜像列表信息

操作场景

用户可以通过导出镜像列表信息的操作，查看镜像详情，并以CSV文件的形式将镜像列表信息导出至本地。

系统支持选择导出该区域的公共镜像信息、或用户在该区域拥有的私有镜像信息：

- 如果导出的是**公共镜像**信息，该文件记录了：镜像的名称、镜像状态、操作系统、镜像类型、创建时间、系统盘、最小内存。
- 如果导出的是**私有镜像**信息，该文件记录了：镜像的名称、镜像ID、镜像状态、操作系统、镜像类型、创建时间、磁盘容量、共享盘信息、镜像大小、是否加密等信息。

导出私有镜像列表信息

1. 登录IMS控制台。

- a. 登录管理控制台。
 - b. 选择“计算 > 镜像服务”。
进入镜像服务页面。
2. 选择“私有镜像”页签，单击私有镜像列表上方的“导出”，按需选择导出范围。
系统会将您账号下，当前区域所选的私有镜像信息自动导出，并下载至本地。

导出公共镜像列表信息

1. 登录IMS控制台。
 - a. 登录管理控制台。
 - b. 选择“计算 > 镜像服务”。
进入镜像服务页面。
2. 选择“公共镜像”页签，单击公共镜像列表上方的“导出”，选择“导出全部数据到XLSX”。
当前区域的所有公共镜像信息自动导出，并下载至本地。

4.5 查看镜像的磁盘容量

操作场景

私有镜像创建成功后，您可以通过控制台直接查看镜像的磁盘容量。其中，整机镜像与其他私有镜像的磁盘容量查看方式略有不同：

- 查看系统盘镜像、数据盘镜像、ISO镜像的磁盘容量，请参考[查看磁盘容量（系统盘镜像/数据盘镜像/ISO镜像）](#)。
- 查看整机镜像的磁盘容量，请参考[查看磁盘容量（整机镜像）](#)。

查看磁盘容量（系统盘镜像/数据盘镜像/ISO 镜像）

对于系统盘镜像、数据盘镜像、ISO镜像的磁盘容量，可以通过控制台私有镜像列表中显示的“磁盘容量”直接查看。

1. 登录IMS控制台。
 - a. 登录管理控制台。
 - b. 选择“计算 > 镜像服务”。
进入镜像服务页面。
2. 选择“私有镜像”页签进入对应的镜像列表。
3. 在镜像所在行的“磁盘容量”列下，查看镜像的磁盘容量，单位为“GB”。

查看磁盘容量（整机镜像）

整机镜像的磁盘容量，需通过查看创建整机镜像的备份获取。用于创建整机镜像的备份中系统盘容量+数据盘容量之和，即为整机镜像的磁盘容量。

1. 登录IMS控制台。
 - a. 登录管理控制台。

- b. 选择“计算 > 镜像服务”。
进入镜像服务页面。
2. 选择“私有镜像”页签进入对应的镜像列表。
在镜像所在行的“磁盘容量”列下，发现整机镜像的磁盘容量显示为“--”，未显示磁盘容量值。
3. 单击整机镜像的名称，进入详情页。
4. 选择镜像详情下方的“备份详情”页签，查看备份中系统盘、数据盘的容量。整机镜像的磁盘容量即为备份中系统盘容量与数据盘容量之和。
整机镜像的磁盘容量=备份中系统盘的磁盘容量+备份中数据盘的磁盘容量
假设备份详情中显示：
 - 系统盘的磁盘容量为40GB，没有数据盘，则整机镜像的磁盘容量为40GB。
 - 系统盘的磁盘容量为40GB，数据盘的磁盘容量为40GB，则整机镜像的磁盘容量为80GB。

4.6 删除镜像

操作场景

您可以删除不需要的私有镜像。

- 删除私有镜像后，将无法找回，请谨慎操作。
- 删除私有镜像后，不能再使用该镜像创建云服务器或云硬盘。
- 删除私有镜像后，已使用该镜像创建的云服务器仍可正常使用，并会继续产生费用，但是无法重装操作系统，也不能创建相同配置的云服务器。
- 删除复制镜像的源镜像，对复制后的镜像没有影响；反之亦然。
- 删除正在制作的整机镜像可能会导致备份残留继续收费，如果不需要此备份，请在云备份或云服务器备份控制台进行删除。

操作步骤

1. 登录IMS控制台。
 - a. 登录管理控制台。
 - b. 选择“计算 > 镜像服务”。
进入镜像服务页面。
2. 单击“私有镜像”页签进入对应的镜像列表。
3. 在镜像所在行的“操作”列下，单击“更多 > 删除”。

说明

- 支持批量删除多个镜像，方法如下：
1. 在镜像列表中勾选需要删除的镜像。
 2. 单击镜像列表上方的“删除”。
4. （可选）选择是否勾选“删除整机镜像关联的云备份”。
待删除的私有镜像中包含整机镜像时，该参数可见。
系统支持在删除整机镜像时，同步删除其关联的云备份。

📖 说明

执行删除镜像操作后，可能出现整机镜像删除成功、关联的云备份删除失败的情况。这可能是由于云备份还在创建中，暂不支持删除。请根据报错提示单击“去删除”，根据界面引导继续进行删除操作。

5. 确认删除镜像，单击“是”。

4.7 共享镜像

4.7.1 共享镜像概述

共享镜像是将自己已经创建好的私有镜像共享给其他用户使用。共享后，接受者可以使用该共享镜像快速创建运行同一镜像环境的云服务器。

⚠️ 注意

云平台无法保证共享镜像的完整性和安全性，使用共享镜像时请确认镜像来源于可信的共享者。

约束与限制

- 镜像支持共享到同一区域内的其他租户。
- 系统盘镜像和数据盘镜像最多可以共享给128个租户，整机镜像最多可以共享给10个租户。

共享过程

用户A作为共享镜像提供者，用户B作为共享镜像接受者时，用户A将私有镜像共享给用户B的具体流程如下：

1. 用户B提供自己的项目ID给用户A。
2. 用户A共享指定的镜像给用户B。
3. 用户B确认接受用户A的共享镜像。
用户B可以使用用户A共享的镜像，完成创建云服务器等操作。

相关问题

还有其他关于共享镜像的疑问，请参考[镜像共享FAQ](#)。

4.7.2 获取项目 ID

操作场景

用户A将私有镜像共享给用户B之前，用户B须将自己的项目ID提供给用户A。本节指导用户B获取自己的项目ID。

操作步骤

1. 用户B登录IMS控制台。
 - a. 登录管理控制台。
 - b. 选择“计算 > 镜像服务”。
进入镜像服务页面。
2. 单击右上角的用户名，选择下拉列表中的“我的凭证”。
在“我的凭证”页面的项目列表中查看项目ID。
镜像仅支持共享到同一区域内的其他租户，请务必获取同区域的项目ID。

4.7.3 共享指定镜像

操作场景

用户A获取用户B的项目ID之后，可以将指定的私有镜像共享给用户B。共享镜像可以分为批量镜像共享和单个镜像共享两种方式，用户可以按照需要进行选择。

前提条件

- 用户A已获取用户B的项目ID。
- 共享镜像前，请确认私有镜像已清除敏感数据和文件。

操作步骤

- **批量镜像共享**
 - a. 用户A登录IMS控制台。
 - i. 登录管理控制台。
 - ii. 选择“计算 > 镜像服务”。
进入镜像服务页面。
 - b. 单击“私有镜像”页签进入镜像列表页面。
 - c. 勾选需要共享的私有镜像，单击镜像列表上方的“共享”。
 - d. 在“共享镜像”窗口，输入用户B的项目ID。
如果需要添加多个镜像接受者，请使用英文逗号间隔。

说明

- 一次最多可以输入100个项目ID。
- 仅支持共享到同一区域内的其他租户。
- 如果用户B是多项目用户，用户A可以将私有镜像共享至用户B在区域下创建的所有项目。
- e. 单击“确定”，完成镜像共享。
- **单个镜像共享**
 - a. 用户A登录IMS控制台。
 - i. 登录管理控制台。
 - ii. 选择“计算 > 镜像服务”。
进入镜像服务页面。

- b. 单击“私有镜像”页签进入镜像列表页面。
- c. 在需要共享的私有镜像所在行的“操作”列，单击“更多”，选择“共享”。
- d. 在“共享镜像”窗口，输入用户B的项目ID。
如果需要添加多个镜像接受者，请使用英文逗号间隔。

说明

- 一次最多可以输入100个项目ID。
 - 仅支持共享到同一区域内的其他租户。
 - 如果用户B是多项目用户，用户A可以将私有镜像共享至用户B在区域下创建的所有项目。
- e. 单击“确定”，完成镜像共享。

相关操作

用户A成功共享镜像后，用户B可以在管理控制台同一区域的“计算 > 镜像服务 > 共享镜像”页面接受共享镜像，详细操作请参见[接受或拒绝共享镜像](#)。

4.7.4 接受或拒绝共享镜像

操作场景

用户A完成共享指定镜像的操作后，用户B会收到是否接受镜像的提示。用户B可以选择接受全部或者部分镜像，也可以拒绝全部或者部分镜像。

说明

- 用户B所选的区域必须和用户A一致，否则看不到待接受的共享镜像。

前提条件

用户A已完成指定镜像的共享。

- 用户A已完成指定镜像的共享。
- 如果用户A共享的是整机镜像，那么B在接受前需要先创建云服务器备份存储库，作为存放整机镜像以及整机镜像所属备份的容器。注意，在创建云服务器备份存储库时，“保护类型”必须为“备份”。

操作步骤

1. 用户B登录IMS控制台。
 - a. 登录管理控制台。
 - b. 选择“计算 > 镜像服务”。
进入镜像服务页面。
2. 在左上角切换区域，将其切换为项目ID所在的区域。
3. 单击“共享镜像”页签进入镜像列表页面。
共享镜像列表上方提示用户B是否接受用户A共享的镜像。
 - 接受全部的共享镜像：单击右上角“全部接受”。

- 接受部分镜像：勾选接受的共享镜像，单击“接受”。
- 拒绝部分镜像：勾选拒绝的共享镜像，单击“拒绝”。

📖 说明

如果在“共享镜像”页签未查看到镜像共享信息，请检查选择的区域是否正确。

4. （可选）在“接受整机镜像”页面，选择一个保护类型为“备份”的云服务器备份存储库，并单击“确定”。

当您接受的共享镜像为整机镜像时，该页面可见。

由于备份共享功能的要求，整机镜像在接受共享前，需要先指定用于存储备份的存储库，且存储库容量应不小于整机镜像所关联云备份的磁盘容量（系统盘和数据盘容量总和）。

📖 说明

更多关于存储库的信息，请参见《云备份用户指南》。

执行结果

- 待确认：对于用户B未及时处理（接受或拒绝）的共享镜像，该共享镜像处于待确认状态。
待确认的共享镜像不会显示在共享镜像列表中。
- 接受：接受后，该镜像将显示在共享镜像列表中。用户B可以使用共享镜像列表中的镜像创建云服务器。
- 拒绝：拒绝后，该镜像不会显示在共享镜像列表中。您可以通过单击“已拒绝镜像”，查看被拒绝的共享镜像。被拒绝的镜像支持再次被接受。

后续操作

用户B接受用户A共享的系统盘镜像后，用户B可以进行以下操作：

- 使用共享镜像创建一台或多台云服务器，在选择镜像时，选择“共享镜像”即可。具体操作请参见《弹性云服务器用户指南》“创建弹性云服务器”章节。
- 使用共享镜像为已有云服务器更换操作系统。具体操作请参见《弹性云服务器用户指南》“切换操作系统”章节。

用户B接受用户A共享的数据盘镜像后，用户B可以使用该镜像申请新的云硬盘，在共享镜像的操作列单击“申请数据盘”即可。

4.7.5 拒绝已经接受的共享镜像

操作场景

用户接受了其他用户共享的镜像后，如果不再需要关注该共享镜像，可以将该共享镜像从自己的可使用共享镜像列表拒绝。

拒绝后，该镜像不再显示在共享镜像列表中。

前提条件

已接受其他用户共享的镜像。

操作步骤

1. 登录IMS控制台。
 - a. 登录管理控制台。
 - b. 选择“计算 > 镜像服务”。
进入镜像服务页面。
2. 单击“共享镜像”页签进入镜像列表页面。
3. 请选择拒绝镜像方式：
 - 拒绝批量镜像：勾选需要拒绝的镜像，单击列表上方的“拒绝”，并在弹出的“拒绝镜像”对话框中，确认拒绝的共享镜像信息，单击“是”。
 - 拒绝单个镜像：在需要拒绝的私有镜像所在行的“操作”列，单击“拒绝”“更多 > 拒绝”，并在弹出的“拒绝镜像”对话框中，确认拒绝的共享镜像信息，单击“是”。

4.7.6 接受已经拒绝的共享镜像

操作场景

用户拒绝了其他用户共享的镜像后，如果再需要使用该共享镜像，可以将该共享镜像从自己的已拒绝镜像的列表中重新接受。

前提条件

- 已拒绝其他用户共享的镜像。
- 其它用户未取消共享该镜像。

操作步骤

1. 登录IMS控制台。
 - a. 登录管理控制台。
 - b. 选择“计算 > 镜像服务”。
进入镜像服务页面。
2. 单击“共享镜像”页签进入镜像列表页面。
3. 单击“已拒绝镜像”，弹出“已拒绝镜像”列表。
4. 选择需要再次接受的镜像，单击“再次接受”。完成已拒绝共享镜像的重新接受。
5. 在“共享镜像”的镜像列表中可以查看步骤4中重新接受的镜像。

4.7.7 取消共享镜像

操作场景

用户可以取消共享给其他用户的镜像。取消后，存在如下影响：

- 接受者无法通过管理控制台或API查询到该镜像。
- 接受者无法使用该镜像创建云服务器或云硬盘，或者为已有云服务器切换操作系统。

- 接受者使用共享镜像创建的云服务器无法重装操作系统，也不能创建相同配置的云服务器。

前提条件

用户已共享私有镜像给其他用户。

操作步骤

1. 登录IMS控制台。
 - a. 登录管理控制台。
 - b. 选择“计算 > 镜像服务”。
进入镜像服务页面。
2. 单击“私有镜像”页签进入镜像列表页面。
3. 在需要取消共享的私有镜像所在行的操作列中，单击“更多”，选择“共享”。
4. 在共享镜像的对话框中单击“取消共享”页签。
5. 勾选需要取消共享的项目ID，单击“确定”。

4.7.8 添加镜像的共享租户

操作场景

对于共享镜像，用户可以查看该镜像已共享的租户，并为共享镜像添加新的共享租户，供其他用户使用。

前提条件

- 用户有已共享的私有镜像。
- 用户已获取要添加的共享租户的项目ID。

操作步骤

1. 登录IMS控制台。
 - a. 登录管理控制台。
 - b. 选择“计算 > 镜像服务”。
进入镜像服务页面。
2. 单击“私有镜像”页签进入镜像列表页面。
3. 单击镜像名称，进入镜像详情页面。
4. 单击“添加租户”。
5. 在“添加租户”窗口，输入新增共享租户的项目ID，单击“确定”。
如果需要添加多个共享租户，请输入新增共享租户的项目ID，并使用英文逗号间隔。单击“确定”，完成共享租户的添加。

说明

- 仅支持共享到同一区域内的其他租户。
- 获取的项目ID为接受共享的用户（用户B）的同区域项目ID。比如，用户A要将“区域1”的项目共享给用户B，请获取用户B“区域1”的项目ID。如果获取非用户B或者非“区域1”的项目ID，将无法共享并提示“在当前Region没有发现该成员信息。”。

4.7.9 删除镜像的共享租户

操作场景

用户可以编辑已共享的私有镜像成员列表，删除共享镜像的租户。

前提条件

- 用户有已共享的私有镜像。
- 用户已获取要删除的共享租户的项目ID。

操作步骤

1. 登录IMS控制台。
 - a. 登录管理控制台。
 - b. 选择“计算 > 镜像服务”。
进入镜像服务页面。
2. 单击“私有镜像”页签进入镜像列表页面。
3. 单击镜像名称，进入镜像详情页面。
4. 查看共享租户。
5. 在租户列表中，选择删除成员的方式。
 - 删除单个成员：单击需要删除成员所在行的“删除”。
 - 删除全部成员：单击租户列表上方的“全部删除”。
6. 单击“是”。

4.7.10 复制共享镜像

操作场景

您可以通过复制共享镜像功能，将共享镜像创建为私有镜像。复制后的镜像将显示在私有镜像列表中，和其他私有镜像一样，支持创建云服务器、导出、共享、复制等功能。

约束与限制

- 当前仅支持数据盘镜像和系统盘镜像复制。
- 当前仅支持区域内复制。
- 复制的镜像大小不超过128GB。

操作步骤

1. 登录IMS控制台。
 - a. 登录管理控制台。
 - b. 选择“计算 > 镜像服务”。
进入镜像服务页面。
2. 在“镜像服务”列表页面，选择“共享镜像”页签。
系统显示已被接受的共享镜像。

3. 选中待复制的共享镜像，单击“操作”列下的“更多 > 复制”。
4. 在“复制镜像”对话框中，输入新镜像的“名称”和“描述”。
5. 单击“确定”。

您可以选择“私有镜像”页签，在私有镜像列表栏，查看复制后镜像的创建状态。当镜像的状态为“正常”时，表示创建完成。

4.8 复制镜像

操作场景

用户可以通过复制镜像功能使镜像具备一些高级特性（如快速发放）。用户复制镜像的场景如下：

- 优化系统盘镜像使其支持云服务器快速发放
系统盘镜像支持快速发放特性，可有效缩短云服务器实例的下发时间。存量系统盘镜像可能不支持快速发放，需要使用复制镜像功能进行优化。假设A镜像不支持快速发放，复制为copy_A镜像后，就可以使用copy_A镜像快速发放云服务器实例了。

约束与限制

- 整机镜像不支持复制。
- 使用ISO文件创建的私有镜像不支持区域内复制。

前提条件

复制镜像时，使用的私有镜像的状态为“正常”。

操作步骤

1. 登录IMS控制台。
 - a. 登录管理控制台。
 - b. 选择“计算 > 镜像服务”。
进入镜像服务页面。
2. 在私有镜像列表中，在需要复制的镜像所在行，单击操作列下的“更多 > 复制”。
3. 在“复制镜像”对话框中，填写如下参数：
 - 名称：输入一个方便您识别的镜像名称。
 - 描述：可选参数，对镜像进行描述。
4. 单击“确定”。
返回私有镜像列表，在“状态”栏可以看到任务进度，显示“正常”时表示复制成功。

4.9 优化私有镜像（Windows）

4.9.1 优化过程（Windows）

云服务器的正常运行依赖于XEN Guest OS driver（PV driver）和KVM Guest OS driver（virtio驱动），为了同时支持XEN虚拟化和KVM虚拟化，以及提升云服务器网络性能，需要确保镜像安装了PV driver和virtio驱动。

1. 将待优化的Windows私有镜像创建为云服务器，并开机登录该云服务器。
2. 在云服务器云服务器上安装最新版本PV driver。
具体操作请参见[安装PV driver](#)。
3. 安装在KVM虚拟化资源池创建云服务器所需的virtio驱动。
具体操作请参见[安装virtio驱动](#)。
4. 在云服务器操作系统中打开“控制面板 > 电源选项”，在左侧单击“选择关闭显示器的时间”，将“关闭显示器”设置为“从不”，然后保存修改。
5. 清除系统日志，然后关闭云服务器。
具体操作请参见[清除系统日志](#)。
6. 通过云服务器创建Windows私有镜像。

4.9.2 查看 Windows 操作系统云服务器虚拟化类型

您可以在cmd窗口输入以下命令，查看当前云服务器的虚拟化类型。

systeminfo

查看回显信息中System Manufacturer和BIOS版本显示为XEN，说明当前云服务器为XEN虚拟化类型，如果需要同时支持KVM虚拟化，请参考本章节操作优化Windows私有镜像。

📖 说明

如果查出来的虚拟化类型为KVM，也建议您优化私有镜像，避免最终发放的云服务器出现一些不可预知的异常。

图 4-2 查看 Windows 云服务器虚拟化类型

```

>systeminfo
Host Name:                ECS-E54F
OS Name:                  Microsoft Windows Server 2012 R2 Datacenter
OS Version:               6.3.9600 N/A Build 9600
OS Manufacturer:        Microsoft Corporation
OS Configuration:        Standalone Server
OS Build Type:             Multiprocessor Free
Registered Owner:         Windows User
Registered Organization:
Product ID:                00253-50000-00000-AA442
Original Install Date:    11/2/2015, 21:05:21
System Root Time:         8/2/2018, 10:31:04
System Manufacturer:      Xen
System model:              HVM domU
System type:               x64-based PC
Processor(s):              1 Processor(s) Installed.
                           [01]: Intel64 Family 6 Model 62 Stepping 4 GenuineIN
                           tel ~2000 Mhz
BIOS Version:              Xen 4.1.2_115-908.762., 3/21/2018
Windows Directory:        C:\Windows
System Directory:          C:\Windows\system32
Boot Device:               \Device\HarddiskVolume1
System Locale:              en-us;English (United States)
Input Locale:              en-us;English (United States)
Time Zone:                 (UTC+08:00) Beijing, Chongqing, Hong Kong, Urumqi
Total Physical Memory:     1,016 MB
Available Physical Memory: 226 MB
Virtual Memory: Max Size: 1,336 MB
Virtual Memory: Available: 476 MB
Virtual Memory: In Use:    860 MB
    
```

4.9.3 相关软件及获取方式

PV driver

Windows私有镜像优化过程中，使用的PV driver软件包及获取路径如表4-1所示。

表 4-1 PV driver 软件包及获取路径

软件包	适用操作系统	获取方式
pvdri- win2008R2-64bit.zi p	Windows Server 2008 R2 64bit	<a href="https://ecs-instance-driver.obs.cn-north-1.myhuaweicloud.com/pvdri-
win2008R2-64bit.zip">https://ecs-instance- driver.obs.cn- north-1.myhuaweicloud.com/ pvdri- win2008R2-64bit.zip
pvdri- win2012-64bit.zip	Windows Server 2012 64bit	<a href="https://ecs-instance-driver.obs.cn-north-1.myhuaweicloud.com/pvdri-
win2012-64bit.zip">https://ecs-instance- driver.obs.cn- north-1.myhuaweicloud.com/ pvdri- win2012-64bit.zip
pvdri- win2012R2-64bit.zi p	Windows Server 2012 R2 64bit	<a href="https://ecs-instance-driver.obs.cn-north-1.myhuaweicloud.com/pvdri-
win2012R2-64bit.zip">https://ecs-instance- driver.obs.cn- north-1.myhuaweicloud.com/ pvdri- win2012R2-64bit.zip
pvdri- win2016-64bit.zip	Windows Server 2016 64bit	<a href="https://ecs-instance-driver.obs.cn-north-1.myhuaweicloud.com/pvdri-
win2016-64bit.zip">https://ecs-instance- driver.obs.cn- north-1.myhuaweicloud.com/ pvdri- win2016-64bit.zip

virtio 驱动

Windows私有镜像优化过程中，使用的virtio驱动的获取路径如下：

<https://fedorapeople.org/groups/virt/virtio-win/direct-downloads/archive-virtio/>

您可以按需选择适合的版本下载。

4.9.4 安装 PV driver

本章节适用于虚拟化类型为XEN架构的云服务器，目前仅部分老用户在使用XEN架构并逐步切换至KVM架构，新用户将不会再购买到XEN架构的云服务器。如果当前创建的私有镜像用于非已停售的实例规格，请省略此步骤。

操作场景

使用弹性云服务器或者外部镜像文件创建私有镜像时，必须确保操作系统中已安装PV driver，使新发放的云服务器支持XEN虚拟化，同时也可以提升云服务器的I/O处理性能、实现对云服务器硬件的监控和其他高级功能。

须知

如果不安装PV driver，云服务器的网络性能很差，并且安全组和防火墙也不会生效。因此，请您务必安装。

使用公共镜像创建的云服务器默认已安装PV driver，您可以通过以下方法验证：
按如下目录打开“version”文件，检查操作系统中安装的PV driver的版本信息。

C:\Program Files (x86)\Xen PV Drivers\bin\version

- 如果PV driver的版本高于2.5版本，无需执行安装操作。
- 如果未查到PV driver版本信息，或者PV driver的版本低于或等于2.5版本，执行[安装PV driver](#)。

前提条件

- 云服务器已安装操作系统，并且已经绑定弹性公网IP。
- 云服务器的系统磁盘的剩余空间必须大于32MB。
- 如果云服务器的操作系统为Windows 2008，需使用Administrator用户安装PV driver。
- 云服务器已下载PV driver软件包。软件包获取请参考[相关软件及获取方式](#)。
- 为了避免在云服务器上安装PV driver失败，安装前需要：
 - 先卸载第三方虚拟化平台的工具（例如：Citrix Xen Tools、VMware Tools）。相关卸载方法请参考对应的工具的官方文档。
 - 禁用任何防病毒软件或入侵检测软件，安装完成后，您可以再次启用这些软件。

安装 PV driver

1. VNC登录Windows云服务器。
登录云服务器的详细操作请参见《弹性云服务器用户指南》。

📖 说明

必须通过VNC方式登录云服务器，不可使用远程桌面登录方式，因为安装时会更新网卡驱动，远程桌面登录时网卡正在使用，从而导致安装不成功。

2. 在云服务器操作系统界面，选择“开始 > 控制面板”。
3. 单击“卸载程序”。
4. 按照提示，卸载“GPL PV Drivers for Windows x.x.x.xx”。
5. 根据[相关软件及获取方式](#)和云服务器的操作系统类型下载对应的PV driver版本。
6. 解压PV driver软件包。
7. 右键单击“GPL PV Drivers for Windows x.x.x.xx”，并选择“以管理员身份运行”，根据界面提示完成安装。
8. 根据提示重启云服务器，使PV driver生效。
对于Windows Server 2008系统的云服务器，必须重启两次。

📖 说明

安装PV driver后，云服务器网卡的配置信息会丢失，因此，若之前已配置网卡，需重新配置网卡信息。

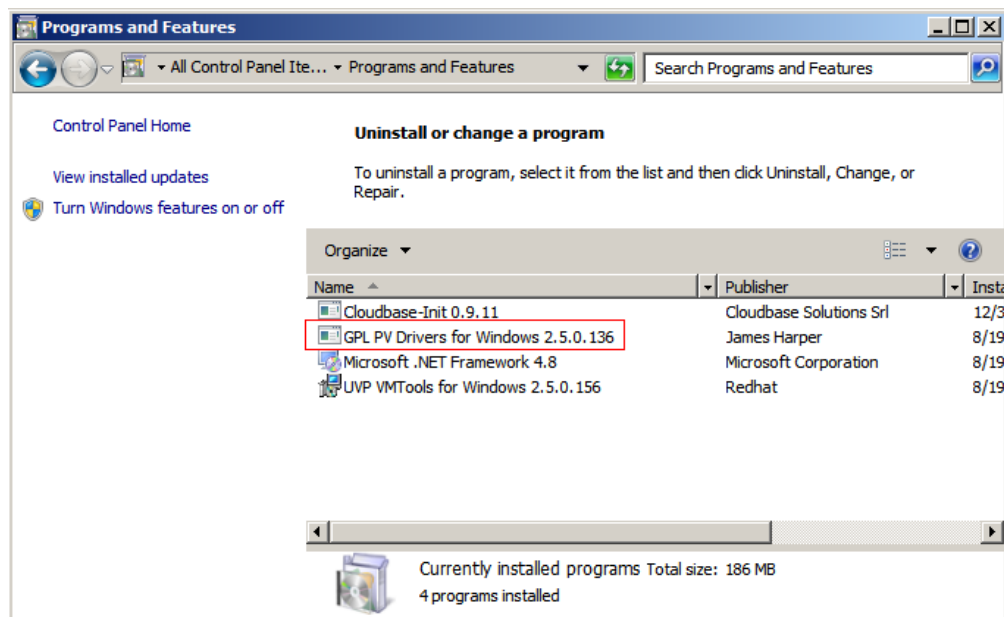
验证安装成功

如果以下步骤验证成功，则表示PV driver安装成功。

1. 选择“开始 > 控制面板 > 程序和功能”。
2. 查看是否存在PV driver信息。

如果存在则表示安装成功，如图4-3所示。

图 4-3 验证安装结果



4.9.5 安装 virtio 驱动

操作场景

virtio是一种用于虚拟化环境的标准化设备接口，旨在提高虚拟机（VM）与宿主机之间的I/O性能。关于virtio驱动程序的详细解释可以参考[官网](#)，virtio-win/kvm-guest-drivers-windows开源代码请参见<https://github.com/virtio-win/kvm-guest-drivers-windows>。

使用弹性云服务器或者外部镜像文件创建私有镜像时，必须确保操作系统中已安装virtio驱动，使新发放的云服务器支持KVM虚拟化，同时也可以提升云服务器的网络性能。

本章节介绍虚拟化类型为KVM架构的云服务器安装virtio驱动的操作步骤。

须知

如果不安装virtio驱动，云服务器的网卡可能无法检测到，无法与外部通信。因此，请您务必安装。

使用公共镜像创建的云服务器默认已安装virtio驱动。

前提条件

云服务器已经绑定弹性公网IP。

安装 virtio 驱动

以使用“virtio-win-0.1.189-1”版本中的“virtio-win-gt-x64.msi”驱动为例介绍安装步骤。

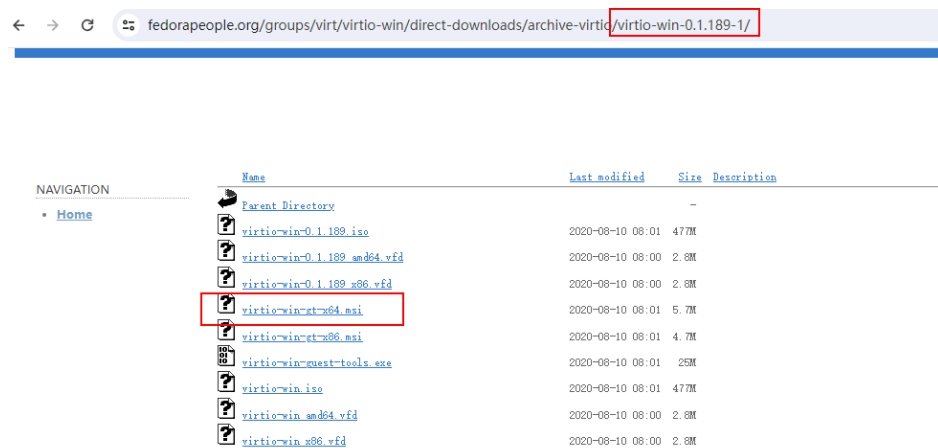
1. VNC登录Windows云服务器。
登录云服务器的详细操作请参见《弹性云服务器用户指南》。

📖 说明

必须通过VNC方式登录云服务器，不可使用远程桌面登录方式，因为安装时会更新网卡驱动，远程桌面登录时网卡正在使用，从而导致安装不成功。

2. 根据[相关软件及获取方式](#)下载virtio驱动版本。

图 4-4 下载驱动



3. 下载完成后，右键单击“virtio-win-gt-x64.msi”，并选择“以管理员身份运行”，根据界面提示完成安装。

图 4-5 单击 RUN

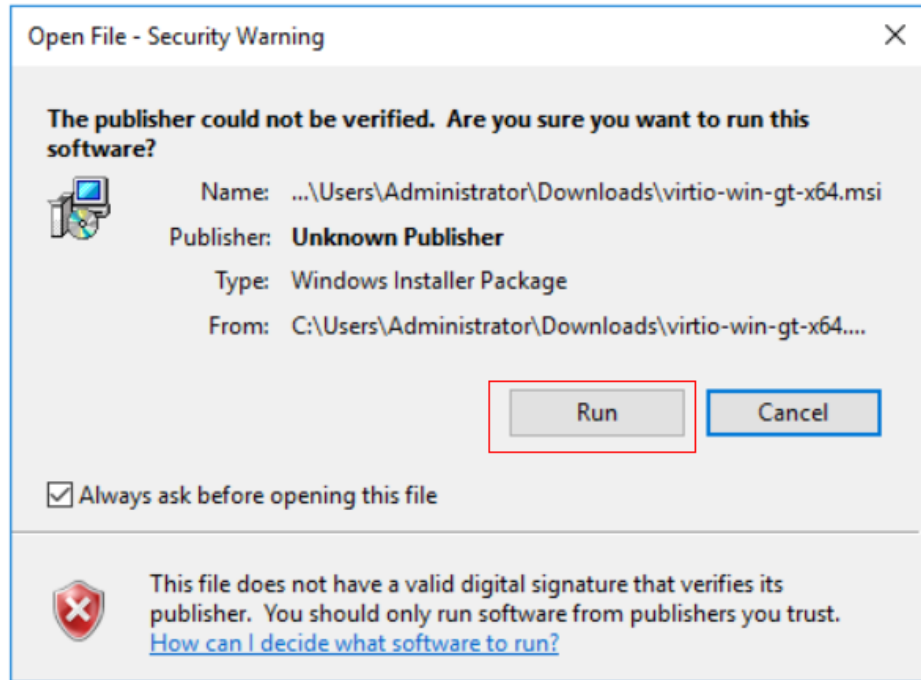


图 4-6 单击 Next

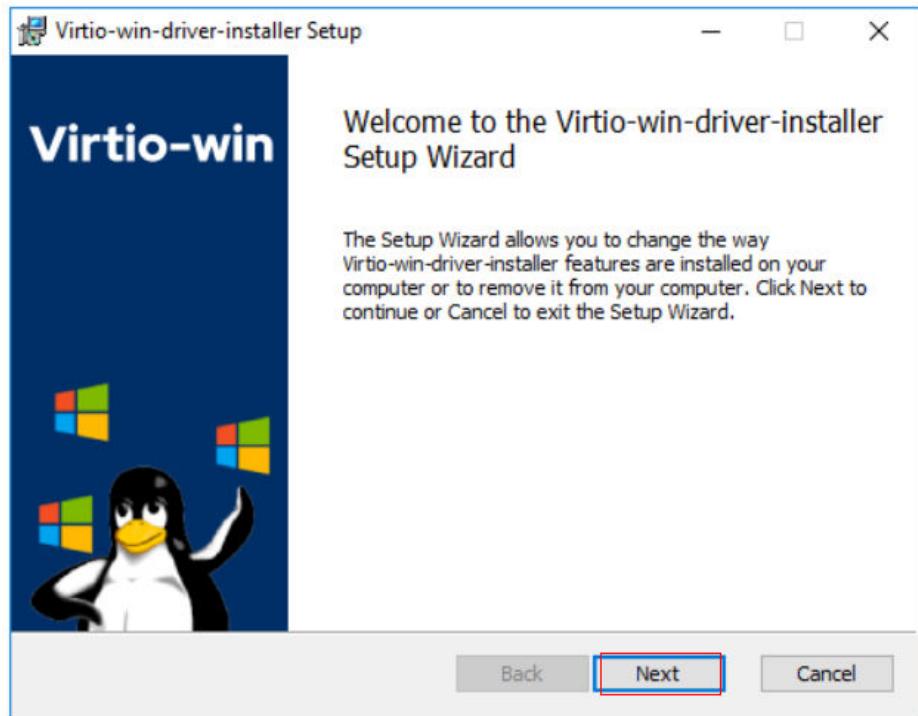
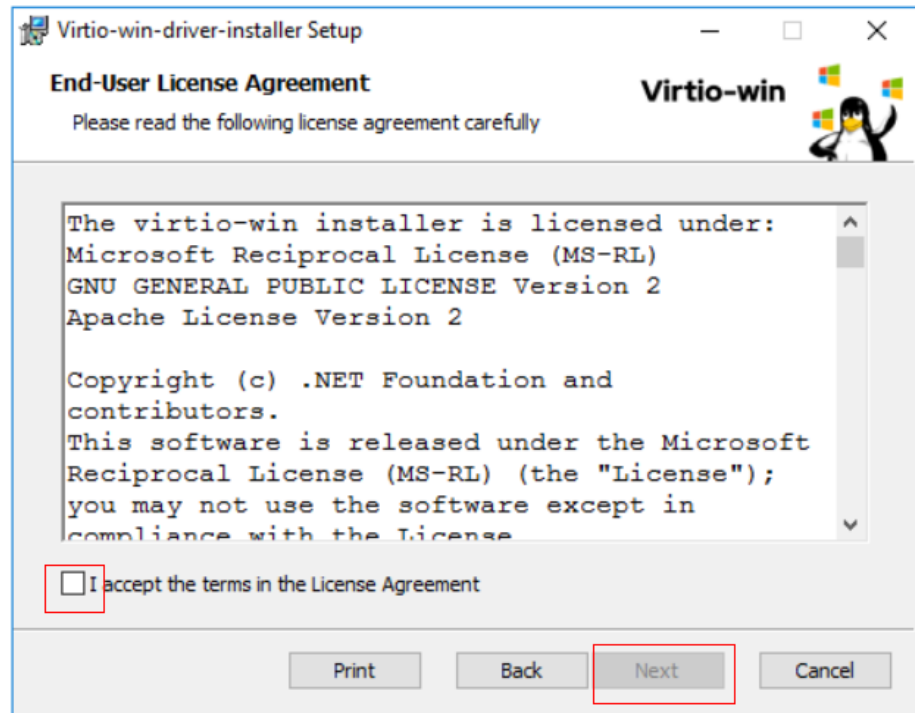


图 4-7 勾选协议



选择安装的virtio驱动。如下图，本次选择安装全部的virtio驱动，也可以选择安装所需的virtio驱动。

图 4-8 选择安装的驱动

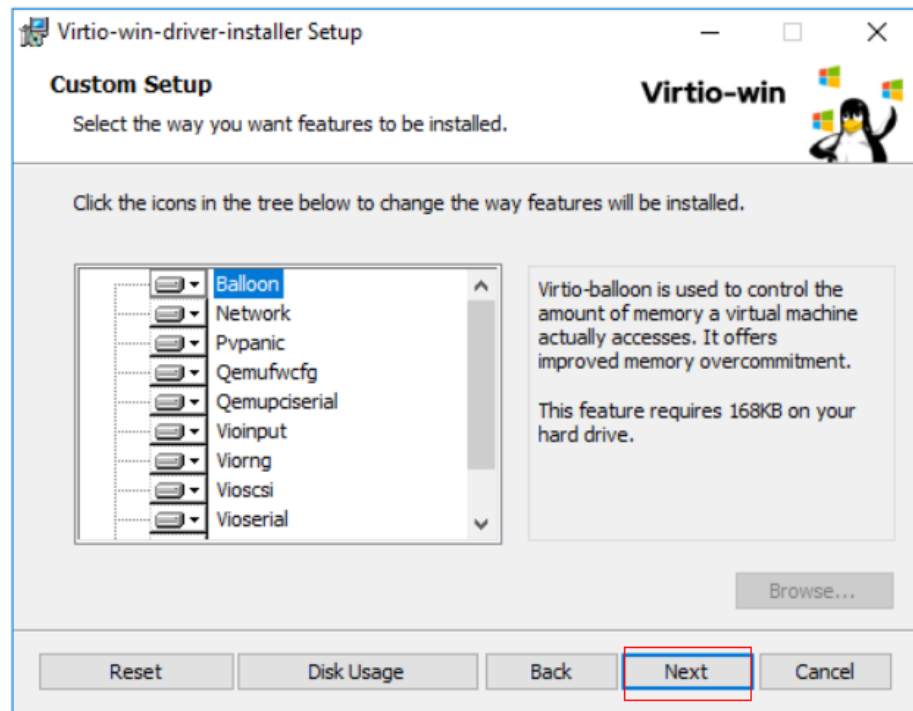
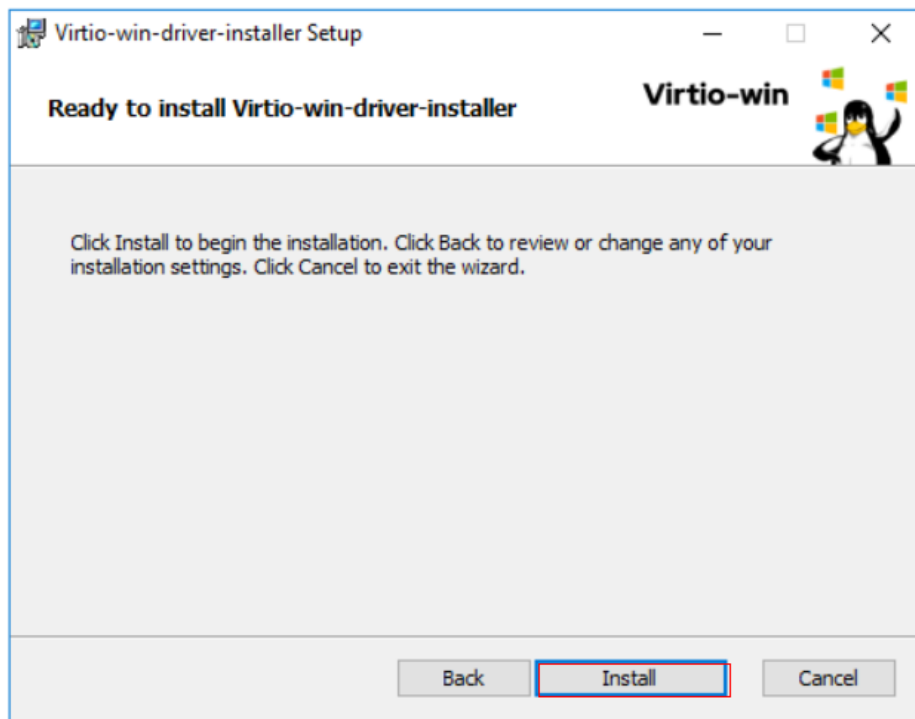
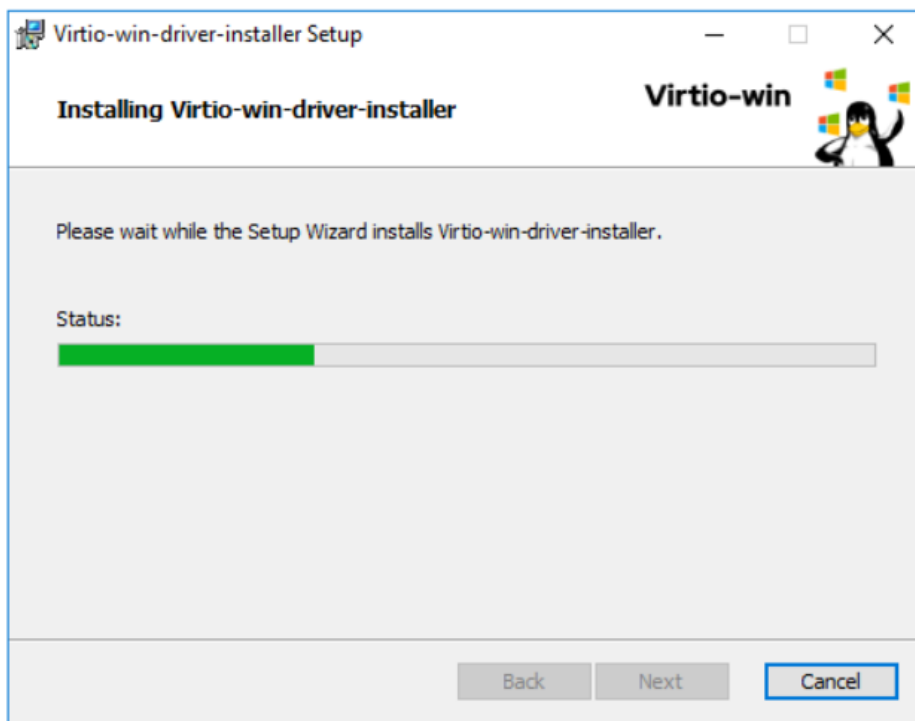


图 4-9 单击安装



4. 等待程序安装完成。

图 4-10 安装中



5. 安装完成后根据提示重启操作系统。

图 4-11 安装完成

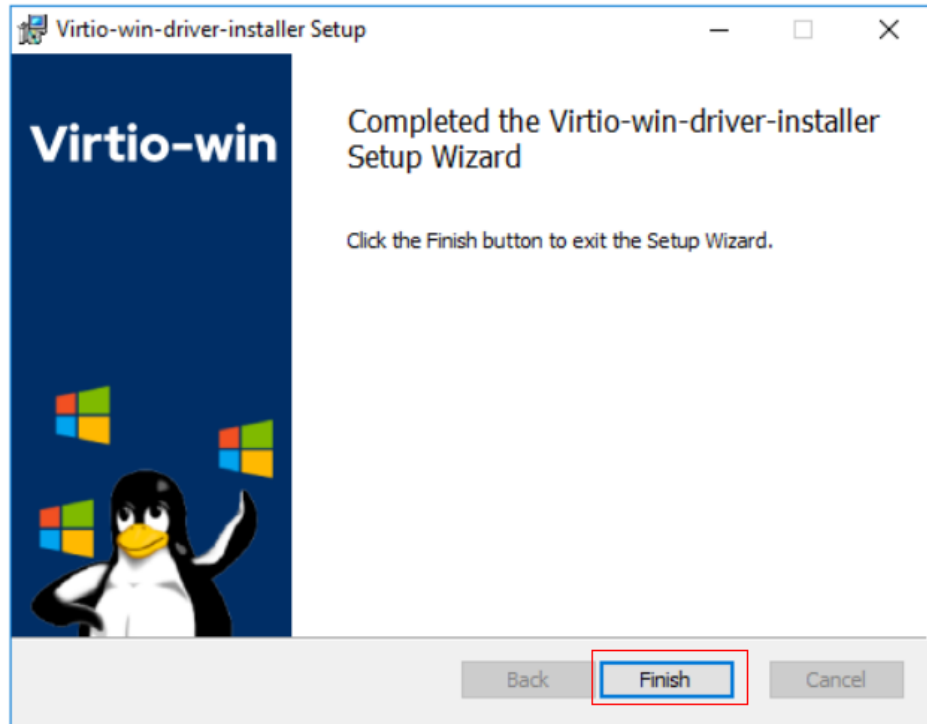
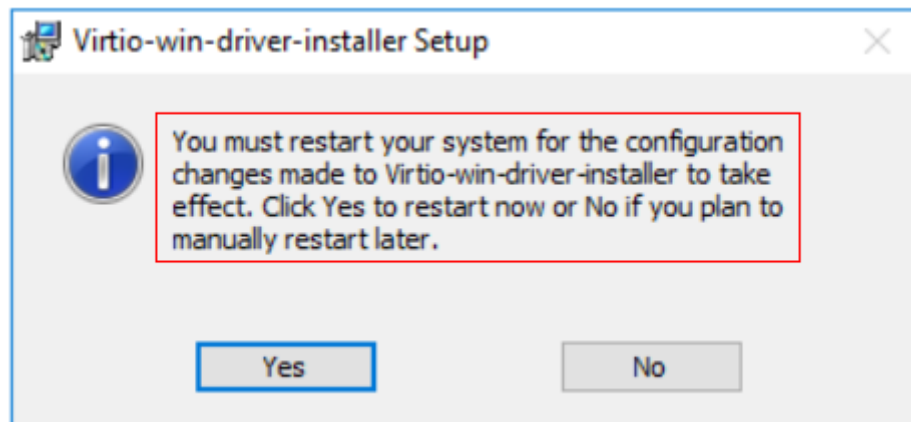


图 4-12 重启提示



6. 重启完成后请执行[验证安装成功](#)，验证virtio驱动是否安装成功。

验证安装成功

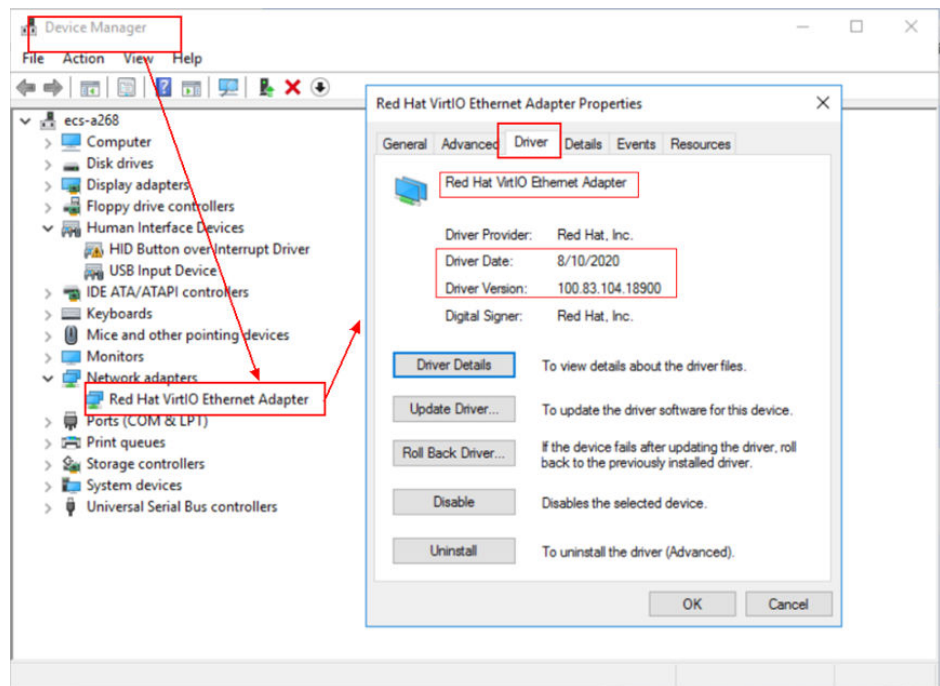
如果以下步骤验证成功，则表示virtio驱动安装成功。

1. 打开设备管理，查找virtio驱动。
2. 查看virtio驱动的版本信息，与下载的virtio驱动版本日期对比。如果一致则表示virtio驱动安装成功。

图 4-13 驱动版本的日期

Name	Last modified	Size	Description
Parent Directory		-	
virtio-win-0.1.189.iso	2020-08-10 08:01	477M	
virtio-win-0.1.189_amd64.vfd	2020-08-10 08:00	2.8M	
virtio-win-0.1.189_x86.vfd	2020-08-10 08:00	2.8M	
virtio-win-gt-x64.msi	2020-08-10 08:01	5.7M	
virtio-win-gt-x86.msi	2020-08-10 08:01	4.7M	
virtio-win-guest-tools.exe	2020-08-10 08:01	25M	
virtio-win.iso	2020-08-10 08:01	477M	
virtio-win_amd64.vfd	2020-08-10 08:00	2.8M	
virtio-win_x86.vfd	2020-08-10 08:00	2.8M	

图 4-14 查找驱动版本信息



4.9.6 清除系统日志

在安装完驱动后请执行以下操作清除系统日志。

1. Windows 2008和Windows 2012的日志路径获取方式为：右键单击“计算机”，选择“管理”。
2. 在弹出的窗口中选择“系统工具 > 事件查看器 > Windows日志”，依次清除Windows日志中5个项目的日志。
3. 清除完系统日志后关闭云服务器。

4.10 优化私有镜像（Linux）

4.10.1 优化过程（Linux）

XEN虚拟化类型的弹性云服务器正逐渐转变为KVM虚拟化类型，因此XEN实例类型的私有镜像，通常需要同时支持XEN虚拟化和KVM虚拟化。也建议您优化KVM虚拟化类型的私有镜像，避免最终发放的弹性云服务器出现一些不可预知的异常。

Linux弹性云服务器的正常运行需依赖于xen-pv驱动、virtio驱动等。因此，需要确保Linux私有镜像已完成相关配置，包括安装驱动、修改UUID等。

准备工作

1. 将待优化的Linux镜像创建为弹性云服务器，并开机登录。
2. 确认是否需要优化私有镜像。

具体操作请参见[确认是否需要优化私有镜像](#)。

请根据虚拟化类型选择对应的优化操作，不同的虚拟化类型优化过程略有不同。

私有镜像优化过程

1. 为了成功安装原生的XEN和KVM驱动，需要先卸载弹性云服务器操作系统中安装的分PV Driver。

具体操作请参见[在Linux系统中卸载PV driver](#)。

说明

KVM虚拟化类型请忽略此步骤。

2. 修改grub文件磁盘标识方式为UUID。
具体操作请参见[修改grub文件磁盘标识方式为UUID](#)。
3. 修改fstab文件磁盘标识方式为UUID。
具体操作请参见[修改fstab文件磁盘标识方式为UUID](#)。
4. 安装原生的驱动。
 - 对于XEN虚拟化类型，请安装原生的XEN驱动和KVM驱动。具体操作请参见[怎样安装原生的XEN和KVM驱动](#)。
 - 对于KVM虚拟化类型，请安装原生的KVM驱动。具体操作请参见[安装原生的KVM驱动](#)。
5. 清除日志文件、历史记录等，关闭云服务器。
具体操作请参见[清除日志文件](#)。
6. 通过弹性云服务器创建Linux私有镜像。

4.10.2 确认是否需要优化私有镜像

- 若当前云服务器为XEN虚拟化类型，需要优化。
- 若当前云服务器为KVM虚拟化类型，且未包含virtio驱动，需要优化。
- 若当前云服务器为KVM虚拟化类型，且包含virtio驱动，不需要优化。

操作步骤

1. 您可以执行以下命令，查看当前云服务器的虚拟化类型。

```
lscpu
```


- 如果回显信息中的Hypervisor vendor为XEN，说明当前云服务器为XEN虚拟化类型，请按照[私有镜像优化过程](#)对私有镜像进行优化。
- 如果回显信息中的Hypervisor vendor为KVM，说明当前云服务器为KVM虚拟化类型。请根据步骤2进一步判断是否需要优化。

图 4-15 查看 Linux 云服务器虚拟化类型

```
# lscpu
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
Byte Order:            Little Endian
CPU(s):                 4
On-line CPU(s) list:   0-3
Thread(s) per core:    1
Core(s) per socket:    4
Socket(s):              1
NUMA node(s):          1
Vendor ID:              GenuineIntel
CPU family:             6
Model:                  62
Model name:             Intel(R) Xeon(R) CPU E5-2690 v2 @ 3.00GHz
Stepping:               4
CPU MHz:                3000.079
BogoMIPS:               6000.15
Hypervisor vendor:     Xen
Virtualization type:   full
L1d cache:              32K
L1i cache:              32K
L2 cache:                256K
L3 cache:                25600K
NUMA node0 CPU(s):     0-3
You have new mail in /var/spool/mail/root
root@CSZY-home1#
```

2. 确认当前系统是否包含virtio驱动。不同操作系统执行命令有所不同，请根据系统类型选择对应的命令执行。
 - CentOS/EulerOS系列

如果引导的虚拟文件系统是initramfs，执行以下命令。

lsinitrd /boot/initramfs-`uname -r`.img | grep virtio

如果引导的虚拟文件系统是initrd，执行如下命令。

lsinitrd /boot/initrd-`uname -r` | grep virtio
 - Ubuntu/Debian系列

lsinitramfs /boot/initrd.img-`uname -r` |grep virtio
 - SUSE和openSUSE系列
 - 版本低于SUSE 12 SP1/openSUSE 13

lsinitrd /boot/initrd-`uname -r` | grep virtio
 - 版本为SUSE 12 SP1或高于SUSE 12 SP1/openSUSE 13

如果引导的虚拟文件系统是initramfs，执行以下命令。

lsinitrd /boot/initramfs-`uname -r`.img | grep virtio

如果引导的虚拟文件系统是initrd，执行如下命令。

lsinitrd /boot/initrd-`uname -r` | grep virtio

如果已经包含了virtio驱动，您可以直接导入私有镜像，无需优化私有镜像。更多信息，请参见[通过外部镜像文件创建Linux系统盘镜像](#)。

```

[root@~]# lsinitrd /boot/initramfs-$(uname -r).img | grep virtio
-rw-r--r-- 1 root root 8984 Sep 30 2021 usr/lib/modules/4.18.0-348.7.1.el8_5.x86_64/kernel/drivers/block/virtio_blk.ko.xz
-rw-r--r-- 1 root root 14836 Sep 30 2021 usr/lib/modules/4.18.0-348.7.1.el8_5.x86_64/kernel/drivers/char/virtio_console.ko.xz
-rw-r--r-- 1 root root 25740 Sep 30 2021 usr/lib/modules/4.18.0-348.7.1.el8_5.x86_64/kernel/drivers/net/virtio_net.ko.xz
-rw-r--r-- 1 root root 8604 Sep 30 2021 usr/lib/modules/4.18.0-348.7.1.el8_5.x86_64/kernel/drivers/scsi/virtio_scsi.ko.xz
    
```

如果没有包含virtio驱动，请按照[私有镜像优化过程](#)对私有镜像进行优化。

4.10.3 在 Linux 系统中卸载 PV driver

操作场景

对于XEN虚拟化类型的弹性云服务器，在优化Linux私有镜像过程中，需要安装原生的XEN和KVM驱动。

为了成功安装原生的XEN和KVM驱动，需要先卸载PV driver。

操作步骤

1. 使用VNC方式，以“root”用户登录云服务器。
2. 执行如下命令，检查操作系统中是否安装PV driver相关的驱动。

ps -ef | grep uvp-monitor

- 若回显信息如下所示，表示已安装PV driver相关的驱动。
- 若无如下回显信息，表示未安装PV driver相关的驱动，本节操作结束。

```

root 4561 1 0 Jun29 ? 00:00:00 /usr/bin/uvp-monitor
root 4567 4561 0 Jun29 ? 00:00:00 /usr/bin/uvp-monitor
root 6185 6085 0 03:04 pts/2 00:00:00 grep uvp-monitor
    
```

3. 在VNC登录窗口的云服务器操作系统界面，打开命令行终端（具体方式请查询对应操作系统的使用手册）。

进入命令行模式。

4. 执行以下命令，卸载PV driver。

/etc/.uvp-monitor/uninstall

- 回显信息如下时，表示Tools卸载成功。
The PV driver is uninstalled successfully. Reboot the system for the uninstallation to take effect.
- 回显信息如下提示不存在“.uvp-monitor”时，请执行步骤5。
-bash: /etc/.uvp-monitor/uninstall: No such file or directory

5. 执行如下操作，删除KVM虚拟化平台下不生效的uvp-monitor，防止日志溢出。

- a. 执行如下命令，查询操作系统是否安装了UVP用户态相关的监控程序。

rpm -qa | grep uvp

回显信息如下所示：

```

libxenstore_uvp3_0-3.00-36.1.x86_64
uvp-monitor-2.2.0.315-3.1.x86_64
kmod-uvpmod-2.2.0.315-3.1.x86_64
    
```

- b. 执行如下命令，删除以下三个安装包。

```

rpm -e kmod-uvpmod
rpm -e uvp-monitor
rpm -e libxenstore_uvp
    
```

4.10.4 修改 grub 文件磁盘标识方式为 UUID

操作场景

在优化Linux私有镜像过程中，需要在云服务器上修改grub文件磁盘标识方式为UUID。

修改menu.lst或者grub.cfg配置（“/boot/grub/menu.lst”或“/boot/grub/grub.cfg”或“/boot/grub2/grub.cfg”或“/boot/grub/grub.conf”或“/boot/efi/EFI/euleros/grub.cfg”，具体路径根据OS不同会有所区别），启动分区使用UUID方式配置。

说明

根据OS不同，配置文件里标记的root分区会有所不同，可能是“root=/dev/xvda”或“root=/dev/disk”，都需要将grub文件磁盘标识方式修改为UUID的形式。

操作步骤

- 以Ubuntu 14.04为例，执行**blkid**命令获取root分区对应的UUID并记录下来，编辑“/boot/grub/grub.cfg”文件，使用root分区的UUID来配置boot项。如果root分区已经使用UUID形式则不需要修改。具体操作方法如下：

- 使用root用户登录云服务器。
- 执行以下命令，列出当前系统中所有已挂载文件系统的类型以及对应设备的UUID。

blkid

回显信息如下所示：

```
/dev/xvda1: UUID="ec51d860-34bf-4374-ad46-a0c3e337fd34" TYPE="ext3"  
/dev/xvda5: UUID="7a44a9ce-9281-4740-b95f-c8de33ae5c11" TYPE="swap"
```

- 执行以下命令，查看“grub.cfg”文件：

cat /boot/grub/grub.cfg

回显信息如下所示：

```
.....menuentry 'Ubuntu Linux, with Linux 3.13.0-24-generic' --class ubuntu --class gnu-linux --  
class gnu --class os --unrestricted $menuentry_id_option 'gnulinux-3.13.0-24-generic-advanced-  
ec51d860-34bf-4374-ad46-a0c3e337fd34' {  
  recordfail  
  load_video  
  gfxmode $linux_gfx_mode  
  insmod gzio  
  insmod part_msdos  
  insmod ext2  
  if [ x$feature_platform_search_hint = xy ]; then  
  search --no-floppy --fs-uuid --set=root ec51d860-34bf-4374-ad46-a0c3e337fd34  
  else  
  search --no-floppy --fs-uuid --set=root ec51d860-34bf-4374-ad46-a0c3e337fd34  
  fi  
  echo 'Loading Linux 3.13.0-24-generic ...'  
  linux /boot/vmlinuz-3.13.0-24-generic root=/dev/xvda1 ro  
  echo 'Loading initial ramdisk ...'  
  initrd /boot/initrd.img-3.13.0-24-generic  
}
```

- 根据“/boot/grub/grub.cfg”配置文件里标记的root分区，查找是否包括“root=/dev/xvda1”或者“root=UUID=ec51d860-34bf-4374-ad46-a0c3e337fd34”信息。

- 存在“root=UUID=ec51d860-34bf-4374-ad46-a0c3e337fd34”，即root分区以UUID的表示形式，无需修改。
 - 存在“root=/dev/xvda1”，即root分区以设备名称表示的形式，请执行步骤5。
- e. 根据“root=/dev/xvda1”，即root分区对应的设备名称，以及**blkid**命令获取的分区信息，找到root分区设备名称对应的UUID。
 - f. 执行以下命令，打开“grub.cfg”文件。

vi /boot/grub/grub.cfg

- g. 按“i”进入编辑模式，将root分区改成UUID形式，本例中将“root=/dev/xvda1”修改为“root=UUID=ec51d860-34bf-4374-ad46-a0c3e337fd34”。
- h. 按“Esc”后，输入:wq，按“Enter”。保存设置并退出vi编辑器。
- i. 执行以下命令，确认修改结果。

cat /boot/grub/grub.cfg

回显信息如下所示表示修改成功，即root分区以UUID的形式表示。

```
.....menuentry 'Ubuntu Linux, with Linux 3.13.0-24-generic' --class ubuntu --class gnu-linux --
class gnu --class os --unrestricted $menuentry_id_option 'gnulinux-3.13.0-24-generic-advanced-
ec51d860-34bf-4374-ad46-a0c3e337fd34' {
  recordfail
  load_video
  gfxmode $linux_gfx_mode
  insmod gzio
  insmod part_msdos
  insmod ext2
  if [ x$feature_platform_search_hint = xy ]; then
  search --no-floppy --fs-uuid --set=root ec51d860-34bf-4374-ad46-a0c3e337fd34
  else
  search --no-floppy --fs-uuid --set=root ec51d860-34bf-4374-ad46-a0c3e337fd34
  fi
  echo 'Loading Linux 3.13.0-24-generic ...'
  linux /boot/vmlinuz-3.13.0-24-generic root=UUID=ec51d860-34bf-4374-ad46-a0c3e337fd34 ro
  echo 'Loading initial ramdisk ...'
  initrd /boot/initrd.img-3.13.0-24-generic
}
```

- 以CentOS 6.5为例，执行**blkid**命令获取root分区对应的UUID并记录下来，编辑“/boot/grub/grub.conf”文件，使用root分区的UUID来配置boot项。如果root分区已经使用UUID形式则不需要修改。具体操作步骤如下：

- a. 使用root用户登录云服务器。
- b. 执行以下命令，列出当前系统中所有已挂载文件系统的类型以及对应设备的UUID。

blkid

回显信息如下所示

```
/dev/xvda1: UUID="749d6c0c-990a-4661-bed1-46769388365a" TYPE="swap"
/dev/xvda2: UUID="f382872b-eda6-43df-9516-5a687fecdc6" TYPE="ext4"
```

- c. 执行以下命令查看“grub.conf”文件：

cat /boot/grub/grub.conf

回显信息如下所示。

```
default=0
timeout=5
splashimage=(hd0,1)/boot/grub/splash.xpm.gz
hiddenmenu
title CentOS (2.6.32-573.8.1.el6.x86_64)
```

```
root (hd0,1)
kernel /boot/vmlinuz-2.6.32-573.8.1.el6.x86_64 ro root=/dev/xvda2 rd_NO_LUKS rd_NO_LVM
LANG=en_US.UTF-8 rd_NO_MD SYSFONT=latacyrheb-sun16
crashkernel=autoKEYBOARDTYPE=pc KEYTABLE=us rd_NO_DM rhgb quiet
initrd /boot/initramfs-2.6.32-573.8.1.el6.x86_64.img
```

- d. 根据“/boot/grub/grub.conf”配置文件里标记的root分区，查找是否包括“root=/dev/xvda2”或者“root=UUID=f382872b-eda6-43df-9516-5a687fecdc6”信息。
 - 若存在“root=UUID=f382872b-eda6-43df-9516-5a687fecdc6”，即root分区以UUID的表示形式，则无需修改。
 - 若存在“root=/dev/xvda2”，即root分区以设备名称表示的形式，请执行步骤5。
- e. 根据“root=/dev/xvda2”，即root分区对应的设备名称，以及blkid命令获取的分区信息，找到root分区设备名称对应的UUID。
- f. 执行以下命令，打开“grub.conf”文件。

```
vi /boot/grub/grub.conf
```

- g. 按“i”进入编辑模式，将root分区改成UUID形式，本例中将“root=/dev/xvda2”修改为“root=UUID=f382872b-eda6-43df-9516-5a687fecdc6”。
- h. 按“Esc”后，输入:wq，按“Enter”。保存设置并退出vi编辑器。
- i. 修改后，执行以下命令确认修改结果。

```
cat /boot/grub/grub.conf
```

回显信息如下所示表示修改成功，即root分区以UUID的形式表示。

```
default=0
timeout=5
splashimage=(hd0,1)/boot/grub/splash.xpm.gz
hiddenmenu
title CentOS (2.6.32-573.8.1.el6.x86_64)
root (hd0,1)
kernel /boot/vmlinuz-2.6.32-573.8.1.el6.x86_64 ro root=UUID=f382872b-
eda6-43df-9516-5a687fecdc6 rd_NO_LUKS rd_NO_LVM LANG=en_US.UTF-8 rd_NO_MD
SYSFONT=latacyrheb-sun16 crashkernel=autoKEYBOARDTYPE=pc KEYTABLE=us rd_NO_DM
rhgb quiet
initrd /boot/initramfs-2.6.32-573.8.1.el6.x86_64.img
```

- 以CentOS 7.0为例，执行blkid命令获取root分区对应的UUID并记录下来，编辑“/boot/grub2/grub.cfg”文件，使用root分区的UUID来配置boot项。如果root分区已经使用UUID形式则不需要修改。

- a. 使用root用户登录云服务器。
- b. 执行以下命令，列出当前系统中所有已挂载文件系统的类型以及对应设备的UUID。

```
blkid
```

回显信息如下所示。

```
/dev/xvda2: UUID="4eb40294-4c6f-4384-bbb6-b8795bbb1130" TYPE="xfs"
/dev/xvda1: UUID="2de37c6b-2648-43b4-a4f5-40162154e135" TYPE="swap"
```

- c. 执行以下命令查看“grub.cfg”文件：

```
cat /boot/grub2/grub.cfg
```

回显信息如下所示。

```
.....
menuentry 'CentOS Linux (3.10.0-229.el7.x86_64) 7 (Core)' --class fedora --class gnu-linux --
class gnu --class os --unrestricted $menuentry_id_option 'gnulinux-3.10.0-229.el7.x86_64-
advanced-4eb40294-4c6f-4384-bbb6-b8795bbb1130' {
```

```
load_video
set gfxpayload=keep
insmod gzio
insmod part_msdos
insmod xfs
set root='hd0,msdos2'
if [ x$feature_platform_search_hint = xy ]; then
search --no-floppy --fs-uuid --set=root --hint='hd0,msdos2'4eb40294-4c6f-4384-bbb6-
b8795bbb1130
else
search --no-floppy --fs-uuid --set=root 4eb40294-4c6f-4384-bbb6-b8795bbb1130
fi
linux16 /boot/vmlinuz-3.10.0-229.el7.x86_64 root=/dev/xvda2 ro crashkernel=auto rhgb quiet
LANG=en_US.UTF-8
initrd16 /boot/initramfs-3.10.0-229.el7.x86_64.img
}
```

- d. 根据“/boot/grub2/grub.cfg”配置文件里标记的root分区，查找是否包括root=/dev/xvda2或者包括root=UUID=4eb40294-4c6f-4384-bbb6-b8795bbb1130信息。
- 存在“root=UUID=4eb40294-4c6f-4384-bbb6-b8795bbb1130”，即root分区以UUID的表示形式，则无需修改。
 - 存在“root=/dev/xvda2”，即root分区以设备名称表示的形式，请执行步骤5。
- e. 根据“root=/dev/xvda2”，即root分区对应的设备名称，以及blkid命令获取的分区信息，找到root分区设备名称对应的UUID。
- f. 执行以下命令，打开“grub.cfg”文件。

```
vi /boot/grub2/grub.cfg
```

- g. 按“i”进入编辑模式，将root分区改成UUID形式，本例中将“root=/dev/xvda2”修改为“root=UUID=4eb40294-4c6f-4384-bbb6-b8795bbb1130”。
- h. 按“Esc”后，输入:wq，按“Enter”。保存设置并退出vi编辑器。
- i. 执行以下命令确认修改结果。

```
cat /boot/grub2/grub.cfg
```

回显信息如下所示表示修改成功，即root分区以UUID的形式表示。

```
.....
menuentry 'CentOS Linux (3.10.0-229.el7.x86_64) 7 (Core)' --class fedora --class gnu-linux --
class gnu --class os --unrestricted $menuentry_id_option 'gnulinux-3.10.0-229.el7.x86_64-
advanced-4eb40294-4c6f-4384-bbb6-b8795bbb1130' {
load_video
set gfxpayload=keep
insmod gzio
insmod part_msdos
insmod xfs
set root='hd0,msdos2'
if [ x$feature_platform_search_hint = xy ]; then
search --no-floppy --fs-uuid --set=root --hint='hd0,msdos2'4eb40294-4c6f-4384-bbb6-
b8795bbb1130
else
search --no-floppy --fs-uuid --set=root 4eb40294-4c6f-4384-bbb6-b8795bbb1130
fi
linux16 /boot/vmlinuz-3.10.0-229.el7.x86_64 root=UUID=4eb40294-4c6f-4384-bbb6-
b8795bbb1130 ro crashkernel=auto rhgb quiet LANG=en_US.UTF-8
initrd16 /boot/initramfs-3.10.0-229.el7.x86_64.img
}
```

4.10.5 修改 fstab 文件磁盘标识方式为 UUID

操作场景

在优化Linux私有镜像过程中，需要在云服务器上修改fstab文件磁盘标识方式为UUID。

操作步骤

- 以CentOS 7.0为例，执行**blkid**命令获取所有分区对应的UUID并记录下来，编辑“/etc/fstab”文件，使用分区的UUID来配置分区自动挂载。

- 使用root用户登录云服务器。
- 执行以下命令，列出当前系统中所有已挂载文件系统的类型以及对应设备的UUID。

blkid

回显信息如下所示。

```
/dev/xvda2: UUID="4eb40294-4c6f-4384-bbb6-b8795bbb1130" TYPE="xfs"  
/dev/xvda1: UUID="2de37c6b-2648-43b4-a4f5-40162154e135" TYPE="swap"
```

- 执行以下命令，查看“fstab”文件。

cat /etc/fstab

回显信息如下所示。

```
[root@CTU1000028010 ~]# cat /etc/fstab  
/dev/xvda2 / xfs defaults 0 0  
/dev/xvda1 swap swap defaults 0 0
```

- 查看“fstab”文件中的磁盘的表示形式为设备名称。
 - 若为UUID的表示形式，无需修改。
 - 若为设备名称的表示形式，请执行步骤5。

- 执行以下命令，打开“fstab”文件。

vi /etc/fstab

- 按“i”进入编辑模式，将fstab中的磁盘表示形式修改为UUID的形式。

- 以CentOS 7.1为例，执行**blkid**命令获取所有分区对应的UUID并记录下来，编辑“/etc/fstab”文件，使用分区的UUID来配置分区自动挂载。

- 使用root用户登录云服务器。
- 执行以下命令，列出当前系统中所有已挂载文件系统的类型以及对应设备的UUID。

blkid

```
/dev/xvda2: UUID="4eb40294-4c6f-4384-bbb6-b8795bbb1130" TYPE="xfs"  
/dev/xvda1: UUID="2de37c6b-2648-43b4-a4f5-40162154e135" TYPE="swap"
```

修改前：

```
[root@CTU1000028010 ~]# cat /etc/fstab  
/dev/xvda2 / xfs defaults 0 0  
/dev/xvda1 swap swap defaults 0 0
```

修改后：

```
[root@CTU1000028010 ~]# cat /etc/fstab  
UUID=4eb40294-4c6f-4384-bbb6-b8795bbb1130 / xfs defaults 0 0  
UUID=2de37c6b-2648-43b4-a4f5-40162154e135 swap swap defaults 0 0
```

- 按“Esc”后，输入:wq，按“Enter”。保存设置并退出vi编辑器。

4. 修改后，执行以下命令确认修改结果。

```
cat /etc/fstab
```

回显信息如下所示，即磁盘的表示形式为UUID，表示修改成功。

```
[root@CTU1000028010 ~]# cat /etc/fstab
UUID=4eb40294-4c6f-4384-bbb6-b8795bbb1130 / xfs defaults 0 0
UUID=2de37c6b-2648-43b4-a4f5-40162154e135 swap swap defaults 0 0
```

4.10.6 安装原生的 XEN 和 KVM 驱动

操作场景

对于XEN虚拟化类型的弹性云服务器，在优化Linux私有镜像过程中，需要安装原生的XEN和KVM驱动。

本节介绍安装原生XEN和KVM驱动的具体方法。

注意

对于XEN实例：如果不安装XEN驱动，弹性云服务器的网络性能很差，并且安全组和防火墙也不会生效；

对于KVM实例：如果不安装KVM驱动，弹性云服务器的网卡可能无法检测到，无法与外部通信。因此，请您务必安装XEN和KVM驱动。

前提条件

- 请先确认您当前的云服务器为XEN构架。
- 对于使用Linux系统原生的XEN和KVM驱动的Linux云服务器，其内核版本必须高于2.6.24。
- 建议您禁用任何防病毒软件或入侵检测软件，安装原生的XEN和KVM驱动完成后，您可以再次启用这些软件。

操作步骤

请根据操作系统版本，修改不同的配置文件：

- CentOS/EulerOS系列操作系统
以CentOS 7.0为例，请修改“/etc/dracut.conf”文件，在add_drivers项中添加xen-pv以及virtio的驱动（xen-pv驱动：xen-blkfront、xen-netfront；virtio驱动：virtio_blk、virtio_scsi、virtio_net、virtio_pci、virtio_ring、virtio）。驱动名之间以空格隔开，保存并退出“/etc/dracut.conf”文件，执行**dracut -f**命令，重新生成initrd。
操作方法可参见[CentOS/EulerOS系列操作系统相关操作](#)。
- Ubuntu/Debian系列系统
请修改“/etc/initramfs-tools/modules”文件，添加xen-pv以及virtio的驱动（xen-pv驱动：xen-blkfront、xen-netfront；virtio驱动：virtio_blk、virtio_scsi、virtio_net、virtio_pci、virtio_ring、virtio）。驱动名之间以空格隔开，保存并退出“/etc/initramfs-tools/modules”文件，执行**update-initramfs -u**命令，重新生成initrd。
操作方法可参见[Ubuntu/Debian系列操作系统相关操作](#)。

- SUSE和openSUSE系列系统，根据操作系统版本不同，修改不同的配置文件。
 - 当操作系统版本低于SUSE 12 SP1或低于openSUSE 13时，请修改“/etc/sysconfig/kernel”文件，在INITRD_MODULES=""添加xen-pv以及virtio的驱动（xen-pv驱动：xen_vnif、xen_vbd、xen_platform_pci；virtio驱动：virtio_blk、virtio_scsi、virtio_net、virtio_pci、virtio_ring、virtio）。驱动名之间以空格隔开，执行mkinitrd命令，重新生成initrd。
 - 当操作系统版本为SUSE 12 SP1时，修改“/etc/dracut.conf”文件，在add_drivers项中添加xen-pv以及virtio的驱动（xen-pv驱动：xen_vnif、xen_vbd、xen_platform_pci；virtio驱动：virtio_blk、virtio_scsi、virtio_net、virtio_pci、virtio_ring、virtio）。驱动名之间以空格隔开，执行命令dracut -f，重新生成initrd。
 - 当操作系统版本高于SUSE 12 SP1或高于openSUSE 13版本时，修改“/etc/dracut.conf”文件，在add_drivers项中添加xen-pv和virtio的驱动（xen-pv驱动：xen-blkfront、xen-netfront；virtio驱动：virtio_blk、virtio_scsi、virtio_net、virtio_pci、virtio_ring、virtio）。驱动名之间以空格隔开，保存并退出“/etc/dracut.conf”文件，执行dracut -f命令，重新生成initrd。

操作方法可参[SUSE/openSUSE系列操作系统相关操作](#)。

📖 说明

SUSE系列操作系统首先应确认OS是否已经安装了xen-kmp包（xen-pv的驱动包），执行以下命令：

```
rpm -qa |grep xen-kmp
```

回显类似如下：

```
xen-kmp-default-4.2.2_04_3.0.76_0.11-0.7.5
```

如果没有安装xen-kmp的包，请到ISO装机文件中获取并安装。

如果误将built-in形式的驱动添加到initrd或initramfs文件中，不会影响云服务器正常使用。

CentOS/EulerOS 系列操作系统相关操作

1. 执行以下命令，打开“/etc/dracut.conf”文件。
vi /etc/dracut.conf
2. 按“i”进入编辑模式，在“add_drivers”项中添加xen-pv和virtio的驱动（具体格式要根据操作系统本身的要求来决定）。

```
[root@CTU10000xxxx ~]# vi /etc/dracut.conf
# additional kernel modules to the default
add_drivers+="xen-blkfront xen-netfront virtio_blk virtio_scsi virtio_net virtio_pci virtio_ring virtio"
.....
```
3. 按“Esc”后，输入:wq，按“Enter”。保存设置并退出“/etc/dracut.conf”文件。
4. 执行以下命令，重新生成initrd。
dracut -f /boot/initramfs-2.6.32-573.8.1.el6.x86_64.img
如果引导的虚拟文件系统不是默认的initramfs，则命令为：**dracut -f 实际使用的initramfs文件名或者initrd文件名**。“实际使用的initramfs文件名或者initrd文件名”可在grub.cfg配置（“/boot/grub/grub.cfg”或“/boot/grub2/grub.cfg”或“/boot/grub/grub.conf”，具体路径根据OS不同会有所区别）中获取。
5. 如果引导的虚拟文件系统是initramfs，执行以下命令，检查是否已经成功装载了原生的XEN和KVM驱动相应模块。
lsinitrd /boot/initramfs-`uname -r`.img | grep xen

lsinitrd /boot/initramfs-`uname -r`.img | grep virtio

如果引导的虚拟文件系统是initrd，执行如下命令，检查是否已经成功装载了原生的XEN和KVM驱动相应模块。

lsinitrd /boot/initrd-`uname -r` | grep xen

lsinitrd /boot/initrd-`uname -r` | grep virtio

以引导的虚拟文件系统是initramfs为例，回显信息如下所示：

```
[root@CTU10000xxxxx home]# lsinitrd /boot/initramfs-`uname -r`.img | grep xen
-rwxr--r-- 1 root root 54888 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/drivers/
block/xen-blkfront.ko
-rwxr--r-- 1 root root 45664 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/
drivers/net/xen-netfront.ko

[root@CTU10000xxxxx home]# lsinitrd /boot/initramfs-`uname -r`.img | grep virtio
-rwxr--r-- 1 root root 23448 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/drivers/
block/virtio_blk.ko
-rwxr--r-- 1 root root 50704 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/
drivers/net/virtio_net.ko
-rwxr--r-- 1 root root 28424 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/drivers/
scsi/virtio_scsi.ko
drwxr-xr-x 2 root root 0 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/drivers/
virtio
-rwxr--r-- 1 root root 14544 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/drivers/
virtio/virtio.ko
-rwxr--r-- 1 root root 21040 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/drivers/
virtio/virtio_pci.ko
-rwxr--r-- 1 root root 18016 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/drivers/
virtio/virtio_ring.ko
```

📖 说明

如果误将built-in形式存在内核中的驱动添加到initrd或initramfs文件中，不会影响云服务器正常使用，这里全写进去只是为了修改的方便，但是使用lsinitrd命令无法检查到。可使用如下方法确定这些驱动是否以built-in形式存在内核中，例如：

```
cat /boot/config-`uname -r` | grep CONFIG_VIRTIO | grep y
cat /boot/config-`uname -r` | grep CONFIG_XEN | grep y
```

Ubuntu/Debian 系列操作系统相关操作

1. 执行以下命令，打开“modules”文件。

```
vi /etc/initramfs-tools/modules
```

2. 按“i”进入编辑模式，修改“/etc/initramfs-tools/modules”文件，添加xen-pv以及virtio的驱动（具体格式要根据操作系统本身的要求来决定）。

```
[root@CTU10000xxxxx ~]# vi /etc/initramfs-tools/modules
.....
# Examples:
#
# raid1
# sd_mOd
xen-blkfront
xen-netfront
virtio_blk
virtio_scsi
virtio_net
virtio_pci
virtio_ring
virtio
```

3. 按“Esc”后，输入:wq，按“Enter”。保存设置并退出“/etc/initramfs-tools/modules”文件。
4. 执行以下命令，重新生成initrd。

update-initramfs -u

5. 执行以下命令，检查是否已经成功装载了原生的XEN和KVM驱动相应模块。

```
lsinitramfs /boot/initrd.img-`uname -r` |grep xen
```

```
lsinitramfs /boot/initrd.img-`uname -r` |grep virtio
```

```
[root@ CTU10000xxxxx home]# lsinitramfs /boot/initrd.img-`uname -r` |grep xen
lib/modules/3.5.0-23-generic/kernel/drivers/net/ethernet/qlogic/netxen
lib/modules/3.5.0-23-generic/kernel/drivers/net/ethernet/qlogic/netxen/netxen_nic.ko
lib/modules/3.5.0-23-generic/kernel/drivers/net/xen-netback
lib/modules/3.5.0-23-generic/kernel/drivers/net/xen-netback/xen-netback.ko
lib/modules/3.5.0-23-generic/kernel/drivers/block/xen-blkback
lib/modules/3.5.0-23-generic/kernel/drivers/block/xen-blkback/xen-blkback.ko
```

```
[root@ CTU10000xxxxx home]# lsinitramfs /boot/initrd.img-`uname -r` |grep virtio
lib/modules/3.5.0-23-generic/kernel/drivers/scsi/virtio_scsi.ko
```

📖 说明

如果误将built-in形式存在内核中的驱动添加到initrd或initramfs文件中，不会影响云服务器器正常使用，这里全写进去只是为了修改方便，但是使用lsinitrd命令无法检查到。可使用如下方法确定这些驱动是否以built-in形式存在内核中，例如：

```
[root@ CTU10000xxxxx home]# cat /boot/config-`uname -r` | grep CONFIG_VIRTIO | grep y
CONFIG_VIRTIO_BLK=y
CONFIG_VIRTIO_NET=y
CONFIG_VIRTIO=y
CONFIG_VIRTIO_RING=y
CONFIG_VIRTIO_PCI=y
CONFIG_VIRTIO_MMIO_CMDLINE_DEVICES=y
[root@ CTU10000xxxxx home]# cat /boot/config-`uname -r` | grep CONFIG_XEN | grep y
CONFIG_XEN_BLKDEV_FRONTEND=y
CONFIG_XEN_NETDEV_FRONTEND=y
```

SUSE/openSUSE 系列操作系统相关操作

当操作系统版本低于SUSE 12 SP1或低于openSUSE 13时，请修改“/etc/sysconfig/kernel”文件，操作步骤请参考[第一种情况](#)。

当操作系统版本为SUSE 12 SP1时，修改“/etc/dracut.conf”文件，添加xen-pv以及virtio的驱动，操作步骤请参考[第二种情况](#)。

当操作系统版本高于SUSE 12 SP1或高于openSUSE 13版本时，修改“/etc/dracut.conf”文件，在add_drivers项中添加xen-pv和virtio的驱动，操作步骤请参考[第三种情况](#)。

- 当操作系统版本低于SUSE 12 SP1或低于openSUSE 13时，操作步骤如下。

📖 说明

SUSE系列操作系统首先应确认OS是否已经安装了xen-kmp包（xen-pv的驱动包），执行如下命令：

```
rpm -qa |grep xen-kmp
```

回显类似如下：

```
xen-kmp-default-4.2.2_04_3.0.76_0.11-0.7.5
```

如果没有安装xen-kmp的包，请到安装ISO中获取并安装。

- a. 执行如下命令，修改“/etc/sysconfig/kernel”文件。

```
vi /etc/sysconfig/kernel
```

- b. 在INITRD_MODULES=""中添加xen-pv以及virtio的驱动（具体格式要根据OS本身的要求来决定）。

```
SIA10000xxxxx:~ # vi /etc/sysconfig/kernel
# (like drivers for scsi-controllers, for lvm or reiserfs)
```

```
#
INITRD_MODULES="ata_piix ata_generic xen_vnif xen_vbd xen_platform_pci virtio_blk
virtio_scsi virtio_net virtio_pci virtio_ring virtio"
```

- c. 执行mkinitrd命令，重新生成initrd。

📖 说明

如果引导的虚拟文件系统不是默认的initramfs或者initrd，则命令为：**dracut -f** 实际使用的initramfs或者initrd文件名。“实际使用的initramfs或者initrd文件名”可在menu.lst或者grub.cfg配置（“/boot/grub/menu.lst”或“/boot/grub/grub.cfg”或“/boot/grub2/grub.cfg”）中获取。

以SUSE 11 SP4为例，如下所示：

```
default 0
timeout 10
gfxmenu (hd0,0)/boot/message
title sles11sp4_001_[_VMX_]
root (hd0,0)
kernel /boot/linux.vmx vga=0x314 splash=silent console=ttyS0,115200n8 console=tty0
net.ifnames=0 NON_PERSISTENT_DEVICE_NAMES=1 showopts
initrd /boot/initrd.vmx
title Failsafe_sles11sp4_001_[_VMX_]
root (hd0,0)
kernel /boot/linux.vmx vga=0x314 splash=silent ide=nodma apm=off noresume edd=off
powersaved=off nohz=off hires=off processor.max+cstate=1 nomodeset x11failsafe
console=ttyS0,115200n8 console=tty0 net.ifnames=0 NON_PERSISTENT_DEVICE_NAMES=1
showopts
initrd /boot/initrd.vmx
```

其中，initrd所在行的/boot/initrd.vmx为实际使用的initrd文件，执行的时候请按照**dracut -f /boot/initrd.vmx**执行。如果initrd所在行的initrd文件不包含/boot目录，如/initramfs-xxx，请在执行dracut命令时增加boot目录，例如：**dracut -f /boot/initramfs-xxx**。

- d. 执行如下命令，检查是否已经成功装载了XEN的PVOPS或者KVM的virtio相应模块。

```
lsinitrd /boot/initrd-`uname -r` | grep xen
```

```
lsinitrd /boot/initrd-`uname -r` | grep virtio
```

```
SIA10000xxxx:~ # lsinitrd /boot/initrd-`uname -r` | grep xen
-rwxr--r-- 1 root root 42400 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/block/
xen-blkfront.ko
-rwxr--r-- 1 root root 44200 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/net/
xen-netfront.ko
```

```
SIA10000xxxx:~ # lsinitrd /boot/initrd-`uname -r` | grep virtio
-rwxr--r-- 1 root root 19248 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/scsi/
virtio_scsi.ko
-rwxr--r-- 1 root root 23856 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/block/
virtio_blk.ko
drwxr-xr-x 2 root root 0 Jul 12 14:53 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio
-rwxr--r-- 1 root root 15848 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio/
virtio_ring.ko
-rwxr--r-- 1 root root 20008 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio/
virtio_pci.ko
-rwxr--r-- 1 root root 12272 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio/
virtio.ko
-rwxr--r-- 1 root root 38208 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/net/
virtio_net.ko
```

- e. 重启云服务器。
f. 重启完毕后，修改“/boot/grub/menu.lst”文件，增加xen_platform_pci.dev_unplug=all和修改root的配置。

修改前如下所示：

```
###Don't change this comment -YaST2 identifier: Original name: linux###
title SUSE Linux Enterprise Server 11SP4 - 3.0.76-0.11 (default)
```

```
root (hd0,0)
kernel /boot/vmlinuz-3.0.76-0.11-default root=UUID=4eb40294-4c6f-4384-bbb6-
b8795bbb1130 splash=silentcrashkernel=256M-:128M showopts vga=0x314
initrd /boot/initrd-3.0.76-0.11-default
```

修改后如下所示：

```
###Don't change this comment -YaST2 identifier: Original name: linux###
title SUSE Linux Enterprise Server 11SP4 - 3.0.76-0.11 (default)
root (hd0,0)
kernel /boot/vmlinuz-3.0.76-0.11-default root=UUID=4eb40294-4c6f-4384-bbb6-
b8795bbb1130 splash=silentcrashkernel=256M-:128M showopts vga=0x314
xen_platform_pci.dev_unplug=all
initrd /boot/initrd-3.0.76-0.11-default
```

说明

- 确保磁盘root分区为UUID的表示形式。
- xen_platform_pci.dev_unplug=all该参数的添加是为了屏蔽qemu设备。
- SUSE 11 SP1 64bit ~ SUSE 11 SP4 64bit系统需要在“menu.lst”文件添加 xen_platform_pci.dev_unplug=all，SUSE 12以后版本默认启用此功能，无需配置。

g. 执行如下命令确认initrd中是否存在XEN驱动。

```
lsinitrd /boot/initrd-`uname -r` | grep xen
```

```
lsinitrd /boot/initrd-`uname -r` | grep virtio
```

```
SIA10000xxxx:~ # lsinitrd /boot/initrd-`uname -r` | grep xen
-rwxr--r-- 1 root root 42400 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/block/
xen-blkfront.ko
-rwxr--r-- 1 root root 44200 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/net/
xen-netfront.ko
```

```
SIA10000xxxx:~ # lsinitrd /boot/initrd-`uname -r` | grep virtio
-rwxr--r-- 1 root root 19248 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/scsi/
virtio_scsi.ko
-rwxr--r-- 1 root root 23856 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/block/
virtio_blk.ko
drwxr-xr-x 2 root root 0 Jul 12 14:53 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio
-rwxr--r-- 1 root root 15848 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio/
virtio_ring.ko
-rwxr--r-- 1 root root 20008 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio/
virtio_pci.ko
-rwxr--r-- 1 root root 12272 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio/
virtio.ko
-rwxr--r-- 1 root root 38208 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/net/
virtio_net.ko
```

说明

如果误将built-in形式存在内核中的驱动添加到initrd或initramfs文件中，不会影响云服务器正常使用，这里全写进去只是为了修改方便，但是使用lsinitrd命令无法检查到。可使用如下方法确定这些驱动是否以built-in形式存在内核中，例如：

```
cat /boot/config-`uname -r` | grep CONFIG_VIRTIO | grep y
```

```
cat /boot/config-`uname -r` | grep CONFIG_XEN | grep y
```

- 当操作系统版本为SUSE 12 SP1时，操作步骤如下。

a. 执行以下命令，打开“/etc/dracut.conf”文件。

```
vi /etc/dracut.conf
```

b. 按“i”进入编辑模式，在“add-drivers”项中添加xen-pv和virtio的驱动（具体格式要根据操作系统本身的要求来决定）。

```
[root@CTU10000xxxx ~]# vi /etc/dracut.conf
# additional kernel modules to the default
add_drivers+="ata_piix ata_generic xen_vnif xen_vbd xen_platform_pci virtio_blk virtio_scsi
virtio_net virtio_pci virtio_ring virtio"
```

- c. 按“Esc”后，输入:**wq**，按“Enter”。保存设置并退出“/etc/dracut.conf”文件。
- d. 执行以下命令，重新生成initrd。

dracut -f /boot/initramfs-文件名

如果引导的虚拟文件系统不是默认的initramfs，则命令为：**dracut -f 实际使用的initramfs文件名或者initrd文件名**。“实际使用的initramfs文件名或者initrd文件名”可在grub.cfg配置（“/boot/grub/grub.cfg”或“/boot/grub2/grub.cfg”或“/boot/grub/grub.conf”，具体路径根据OS不同会有所不同）中获取。

- e. 如果引导的虚拟文件系统是initramfs，执行以下命令，检查是否已经成功装载了原生的XEN和KVM驱动相应模块。

lsinitrd /boot/initramfs-`uname -r`.img | grep xen

lsinitrd /boot/initramfs-`uname -r`.img | grep virtio

如果引导的虚拟文件系统是initrd，执行如下命令，检查是否已经成功装载了原生的XEN和KVM驱动相应模块。

lsinitrd /boot/initrd-`uname -r` | grep xen

lsinitrd /boot/initrd-`uname -r` | grep virtio

- 当操作系统版本高于SUSE 12 SP1或高于openSUSE 13版本时，操作步骤如下。以SUSE Linux Enterprise Server 12 SP2 (x86_64)为例，如下所示：

- a. 执行以下命令，打开“/etc/dracut.conf”文件。

vi /etc/dracut.conf

- b. 按“i”进入编辑模式，在“add_drivers”项中添加xen-pv和virtio的驱动（具体格式要根据操作系统本身的要求来决定）。

```
[root@CTU10000xxxx ~]# vi /etc/dracut.conf
# additional kernel modules to the default
add_drivers+="ata_piix ata_generic xen-blkfront xen-netfront virtio_blk virtio_scsi
virtio_net virtio_pci virtio_ring virtio"
```

- c. 按“Esc”后，输入:**wq**，按“Enter”。保存设置并退出“/etc/dracut.conf”文件。
- d. 执行以下命令，重新生成initrd。

dracut -f /boot/initramfs-文件名

如果引导的虚拟文件系统不是默认的initramfs，则命令为：**dracut -f 实际使用的initramfs文件名或者initrd文件名**。“实际使用的initramfs文件名或者initrd文件名”可在grub.cfg配置（“/boot/grub/grub.cfg”或“/boot/grub2/grub.cfg”或“/boot/grub/grub.conf”，具体路径根据OS不同会有所区别）中获取。

- e. 如果引导的虚拟文件系统是initramfs，执行以下命令，检查是否已经成功装载了原生的XEN和KVM驱动相应模块。

lsinitrd /boot/initramfs-`uname -r`.img | grep xen

lsinitrd /boot/initramfs-`uname -r`.img | grep virtio

如果引导的虚拟文件系统是initrd，执行如下命令，检查是否已经成功装载了原生的XEN和KVM驱动相应模块。

lsinitrd /boot/initrd-`uname -r` | grep xen

lsinitrd /boot/initrd-`uname -r` | grep virtio

以引导的虚拟文件系统是initrd为例，回显信息如下所示：

```
sluo-ecs-30dc:~ # lsinitrd /boot/initrd-`uname -r` | grep xen
-rw-r--r-- 1 root root 69575 Oct 26 2016 lib/modules/4.4.21-69-default/kernel/drivers/block/xen-
```



```
blkfront.ko
-rw-r--r-- 1 root root 53415 Oct 26 2016 lib/modules/4.4.21-69-default/kernel/drivers/net/xen-
netfront.ko
drwxr-xr-x 2 root root 0 Sep 28 10:21 lib/modules/4.4.21-69-default/updates/pvdriver/xen-hcall
-rwxr-xr-x 1 root root 8320 Sep 28 10:21 lib/modules/4.4.21-69-default/updates/pvdriver/xen-
hcall/xen-hcall.ko

sluo-ecs-30dc:~ # lsinitrd /boot/initrd-`uname -r` | grep virtio
-rw-r--r-- 1 root root 29335 Oct 26 2016 lib/modules/4.4.21-69-default/kernel/drivers/block/
virtio_blk.ko
-rw-r--r-- 1 root root 57007 Oct 26 2016 lib/modules/4.4.21-69-default/kernel/drivers/net/
virtio_net.ko
-rw-r--r-- 1 root root 32415 Oct 26 2016 lib/modules/4.4.21-69-default/kernel/drivers/scsi/
virtio_scsi.ko
drwxr-xr-x 2 root root 0 Sep 28 10:21 lib/modules/4.4.21-69-default/kernel/drivers/virtio
-rw-r--r-- 1 root root 19623 Oct 26 2016 lib/modules/4.4.21-69-default/kernel/drivers/virtio/
virtio.ko
-rw-r--r-- 1 root root 38943 Oct 26 2016 lib/modules/4.4.21-69-default/kernel/drivers/virtio/
virtio_pci.ko
-rw-r--r-- 1 root root 24431 Oct 26 2016 lib/modules/4.4.21-69-default/kernel/drivers/virtio/
virtio_ring.ko
```

说明

如果误将built-in形式存在内核中的驱动添加到initrd或initramfs文件中，不会影响云服务器正常使用，这里全写进去只是为了修改方便，但是使用lsinitrd命令无法检查到。可使用如下方法确定这些驱动是否以built-in形式存在内核中，例如：

```
cat /boot/config-`uname -r` | grep CONFIG_VIRTIO | grep y
cat /boot/config-`uname -r` | grep CONFIG_XEN | grep y
```

4.10.7 安装原生的 KVM 驱动

操作场景

在优化Linux私有镜像过程中，需要在云服务器上安装原生的KVM驱动。若云服务器已安装原生的KVM驱动，请忽略本章节。

注意

如果不安装KVM驱动，弹性云服务器的网卡可能无法检测到，无法与外部通信。因此，请您务必安装KVM驱动。

前提条件

- 请先确认您当前的云服务器是否需要优化，操作步骤详见[确认是否需要优化私有镜像](#)。
- 对于使用Linux系统原生KVM驱动的Linux云服务器，其内核版本必须高于2.6.24。
- 建议您禁用任何防病毒软件或入侵检测软件，安装原生的KVM驱动完成后，您可以再次启用这些软件。

操作步骤

请根据操作系统版本，修改不同的配置文件，如[表4-2](#)所示。

表 4-2 不同 OS 的修改方案

操作系统	配置简介	操作指导
CentOS/EulerOS 系列	<p>以CentOS 7.0为例:</p> <ol style="list-style-type: none"> 1. 请修改“/etc/dracut.conf”文件，在 add_drivers项中添加virtio的驱动（virtio驱动：virtio_blk、virtio_scsi、virtio_net、virtio_pci、virtio_ring、virtio）。驱动名之间以空格隔开。 2. 保存并退出“/etc/dracut.conf”文件，执行dracut -f命令，重新生成initrd。 	CentOS/EulerOS系列操作指导
Ubuntu/Debian 系列	<ol style="list-style-type: none"> 1. 请修改“/etc/initramfs-tools/modules”文件，添加virtio的驱动（virtio驱动：virtio_blk、virtio_scsi、virtio_net、virtio_pci、virtio_ring、virtio）。驱动名之间以空格隔开。 2. 保存并退出“/etc/initramfs-tools/modules”文件，执行update-initramfs -u命令，重新生成initrd。 	Ubuntu/Debian系列操作指导
SUSE和 openSUSE系列	<p>版本低于SUSE 12 SP1或低于openSUSE 13时:</p> <ol style="list-style-type: none"> 1. 请修改“/etc/sysconfig/kernel”文件，在INITRD_MODULES=""添加 virtio的驱动（virtio驱动：virtio_blk、virtio_scsi、virtio_net、virtio_pci、virtio_ring、virtio）。驱动名之间以空格隔开。 2. 执行mkinitrd命令，重新生成initrd。 	SUSE/openSUSE系列操作指导（版本低于SUSE 12 SP1/openSUSE 13）
	<p>版本为SUSE 12 SP1时:</p> <ol style="list-style-type: none"> 1. 请修改“/etc/dracut.conf”文件，在 add_drivers项中添加virtio的驱动（virtio驱动：virtio_blk、virtio_scsi、virtio_net、virtio_pci、virtio_ring、virtio）。驱动名之间以空格隔开。 2. 执行命令dracut -f，重新生成initrd。 	SUSE/openSUSE系列操作指导（版本为SUSE 12 SP1）

操作系统	配置简介	操作指导
	<p>版本高于SUSE 12 SP1或高于openSUSE 13版本时:</p> <ol style="list-style-type: none"> 1. 请修改“/etc/dracut.conf”文件，在 add_drivers项中添加virtio的驱动（virtio驱动：virtio_blk、virtio_scsi、virtio_net、virtio_pci、virtio_ring、virtio）。驱动名之间以空格隔开。 2. 保存并退出“/etc/dracut.conf”文件，执行dracut -f命令，重新生成initrd。 	<p>SUSE/openSUSE系列操作指导（版本高于SUSE 12 SP1/openSUSE 13）</p>

CentOS/EulerOS 系列操作指导

1. 执行以下命令，打开“/etc/dracut.conf”文件。
2. 按“i”进入编辑模式，在“add_drivers”项中添加virtio的驱动（具体格式要根据操作系统本身的要求来决定）。

```
[root@CTU10000xxxx ~]# vi /etc/dracut.conf
# additional kernel modules to the default
add_drivers+="virtio_blk virtio_scsi virtio_net virtio_pci virtio_ring virtio"
.....
```

3. 按“Esc”后，输入:wq，按“Enter”。保存设置并退出“/etc/dracut.conf”文件。
4. 参考以下命令，重新生成initrd。

dracut -f /boot/initramfs-2.6.32-573.8.1.el6.x86_64.img

如果引导的虚拟文件系统不是默认的initramfs，则命令为：**dracut -f 实际使用的initramfs文件名或者initrd文件名**。“实际使用的initramfs文件名或者initrd文件名”可在grub.cfg配置（“/boot/grub/grub.cfg”或“/boot/grub2/grub.cfg”或“/boot/grub/grub.conf”，具体路径根据OS不同会有所区别）中获取。

5. 如果引导的虚拟文件系统是initramfs，执行以下命令，检查是否已经成功装载了原生的KVM驱动相应模块。

lsinitrd /boot/initramfs-`uname -r`.img | grep virtio

如果引导的虚拟文件系统是initrd，执行如下命令，检查是否已经成功装载了原生的KVM驱动相应模块。

lsinitrd /boot/initrd-`uname -r` | grep virtio

以引导的虚拟文件系统是initramfs为例，回显信息如下所示：

```
[root@CTU10000xxxx home]# lsinitrd /boot/initramfs-`uname -r`.img | grep virtio
-rwxr--r-- 1 root root 23448 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/drivers/
block/virtio_blk.ko
-rwxr--r-- 1 root root 50704 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/
drivers/net/virtio_net.ko
-rwxr--r-- 1 root root 28424 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/drivers/
scsi/virtio_scsi.ko
drwxr-xr-x 2 root root 0 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/drivers/
virtio
-rwxr--r-- 1 root root 14544 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/drivers/
virtio/virtio.ko
-rwxr--r-- 1 root root 21040 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/drivers/
```

```
virtio/virtio_pci.ko
-rwxr--r-- 1 root root 18016 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/drivers/
virtio/virtio_ring.ko
```

📖 说明

如果误将built-in形式存在内核中的驱动添加到initrd或initramfs文件中，不会影响云服务器器正常使用，这里全写进去只是为了修改的方便，但是使用lsinitrd命令无法检查到。可使用如下方法确定这些驱动是否以built-in形式存在内核中，例如：

```
cat /boot/config-`uname -r` | grep CONFIG_VIRTIO | grep y
```

Ubuntu/Debian 系列操作指导

1. 执行以下命令，打开“modules”文件。

```
vi /etc/initramfs-tools/modules
```

2. 按“i”进入编辑模式，修改“/etc/initramfs-tools/modules”文件，添加virtio的驱动（具体格式要根据操作系统本身的要求来决定）。

```
[root@CTU10000xxxx ~]#vi /etc/initramfs-tools/modules
.....
# Examples:
#
# raid1
# sd_mOd
virtio_blk
virtio_scsi
virtio_net
virtio_pci
virtio_ring
virtio
```

3. 按“Esc”后，输入:wq，按“Enter”。保存设置并退出“/etc/initramfs-tools/modules”文件。
4. 执行以下命令，重新生成initrd。

```
update-initramfs -u
```

5. 执行以下命令，检查是否已经成功装载了原生的KVM驱动相应模块。

```
lsinitramfs /boot/initrd.img-`uname -r` |grep virtio
```

```
[root@ CTU10000xxxx home]# lsinitramfs /boot/initrd.img-`uname -r` |grep virtio
lib/modules/3.5.0-23-generic/kernel/drivers/scsi/virtio_scsi.ko
```

📖 说明

如果误将built-in形式存在内核中的驱动添加到initrd或initramfs文件中，不会影响云服务器器正常使用，这里全写进去只是为了修改方便，但是使用lsinitrd命令无法检查到。可使用如下方法确定这些驱动是否以built-in形式存在内核中，例如：

```
[root@ CTU10000xxxx home]# cat /boot/config-`uname -r` | grep CONFIG_VIRTIO | grep y
CONFIG_VIRTIO_BLK=y
CONFIG_VIRTIO_NET=y
CONFIG_VIRTIO=y
CONFIG_VIRTIO_RING=y
CONFIG_VIRTIO_PCI=y
CONFIG_VIRTIO_MMIO_CMDLINE_DEVICES=y
```

SUSE/openSUSE 系列操作指导（版本低于 SUSE 12 SP1/openSUSE 13）

当操作系统版本低于SUSE 12 SP1或低于openSUSE 13时，请修改“/etc/sysconfig/kernel”文件，操作步骤如下：

1. 执行如下命令，修改“/etc/sysconfig/kernel”文件。

```
vi /etc/sysconfig/kernel
```

2. 在INITRD_MODULES=""中添加virtio的驱动（具体格式要根据OS本身的要求来决定）。

```
SIA10000xxxxx:~ # vi /etc/sysconfig/kernel
# (like drivers for scsi-controllers, for lvm or reiserfs)
#
INITRD_MODULES="ata_piix ata_generic virtio_blk virtio_scsi virtio_net virtio_pci virtio_ring
virtio"
```

3. 执行mkinitrd命令，重新生成initrd。

📖 说明

如果引导的虚拟文件系统不是默认的initramfs或者initrd，则命令为：**dracut -f** 实际使用的initramfs或者initrd文件名。“实际使用的initramfs或者initrd文件名”可在menu.lst或者grub.cfg配置（“/boot/grub/menu.lst”或“/boot/grub/grub.cfg”或“/boot/grub2/grub.cfg”）中获取。

以SUSE 11 SP4为例，如下所示：

```
default 0
timeout 10
gfxmenu (hd0,0)/boot/message
title sles11sp4_001_[_VMX_]
root (hd0,0)
kernel /boot/linux.vmx vga=0x314 splash=silent console=ttyS0,115200n8 console=tty0 net.ifnames=0
NON_PERSISTENT_DEVICE_NAMES=1 showopts
initrd /boot/initrd.vmx
title Failsafe_sles11sp4_001_[_VMX_]
root (hd0,0)
kernel /boot/linux.vmx vga=0x314 splash=silent ide=nodma apm=off noresume edd=off
powersaved=off nohz=off highres=off processor.max+cstate=1 nomodeset x11failsafe
console=ttyS0,115200n8 console=tty0 net.ifnames=0 NON_PERSISTENT_DEVICE_NAMES=1 showopts
initrd /boot/initrd.vmx
```

其中，initrd所在行的/boot/initrd.vmx为实际使用的initrd文件，执行的时候请按照**dracut -f /boot/initrd.vmx**执行。如果initrd所在行的initrd文件不包含/boot目录，如/initramfs-xxx，请在执行dracut命令时增加boot目录，例如：**dracut -f /boot/initramfs-xxx**。

4. 执行如下命令，检查是否已经成功装载了KVM的virtio相应模块。

lsinitrd /boot/initrd-`uname -r` | grep virtio

```
SIA10000xxxxx:~ # lsinitrd /boot/initrd-`uname -r` | grep virtio
-rwxr--r-- 1 root root 19248 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/scsi/
virtio_scsi.ko
-rwxr--r-- 1 root root 23856 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/block/
virtio_blk.ko
drwxr-xr-x 2 root root 0 Jul 12 14:53 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio
-rwxr--r-- 1 root root 15848 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio/
virtio_ring.ko
-rwxr--r-- 1 root root 20008 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio/
virtio_pci.ko
-rwxr--r-- 1 root root 12272 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio/
virtio.ko
-rwxr--r-- 1 root root 38208 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/net/
virtio_net.ko
```

5. 重启云服务器。
6. 执行如下命令确认initrd中是否存在KVM驱动。

lsinitrd /boot/initrd-`uname -r` | grep virtio

```
SIA10000xxxxx:~ # lsinitrd /boot/initrd-`uname -r` | grep virtio
-rwxr--r-- 1 root root 19248 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/scsi/
virtio_scsi.ko
-rwxr--r-- 1 root root 23856 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/block/
virtio_blk.ko
drwxr-xr-x 2 root root 0 Jul 12 14:53 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio
-rwxr--r-- 1 root root 15848 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio/
virtio_ring.ko
```

```
-rwxr--r-- 1 root root 20008 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio/  
virtio_pci.ko  
-rwxr--r-- 1 root root 12272 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio/  
virtio.ko  
-rwxr--r-- 1 root root 38208 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/net/  
virtio_net.ko
```

📖 说明

如果误将built-in形式存在内核中的驱动添加到initrd或initramfs文件中，不会影响云服务器器正常使用，这里全写进去只是为了修改方便，但是使用lsinitrd命令无法检查到。可使用如下方法确定这些驱动是否以built-in形式存在内核中，例如：

```
cat /boot/config-`uname -r` | grep CONFIG_VIRTIO | grep y
```

SUSE/openSUSE 系列操作指导（版本为 SUSE 12 SP1）

当操作系统版本为SUSE 12 SP1时，请修改“/etc/dracut.conf”文件，添加virtio的驱动，操作步骤如下：

1. 执行以下命令，打开“/etc/dracut.conf”文件。

```
vi /etc/dracut.conf
```

2. 按“i”进入编辑模式，在“add-drivers”项中添加virtio的驱动（具体格式要根据操作系统本身的要求来决定）。

```
[root@CTU10000xxxx ~]# vi /etc/dracut.conf  
# additional kernel modules to the default  
add_drivers+="ata_piix ata_generic virtio_blk virtio_scsi virtio_net virtio_pci virtio_ring virtio"
```

3. 按“Esc”后，输入:wq，按“Enter”。保存设置并退出“/etc/dracut.conf”文件。
4. 执行以下命令，重新生成initrd。

```
dracut -f /boot/initramfs-文件名
```

如果引导的虚拟文件系统不是默认的initramfs，则命令为：**dracut -f 实际使用的initramfs文件名或者initrd文件名**。“实际使用的initramfs文件名或者initrd文件名”可在grub.cfg配置（“/boot/grub/grub.cfg”或“/boot/grub2/grub.cfg”或“/boot/grub/grub.conf”，具体路径根据OS不同会有所不同）中获取。

5. 如果引导的虚拟文件系统是initramfs，执行以下命令，检查是否已经成功装载了原生的KVM驱动相应模块。

```
lsinitrd /boot/initramfs-`uname -r`.img | grep virtio
```

如果引导的虚拟文件系统是initrd，执行如下命令，检查是否已经成功装载了原生的KVM驱动相应模块。

```
lsinitrd /boot/initrd-`uname -r` | grep virtio
```

SUSE/openSUSE 系列操作指导（版本高于 SUSE 12 SP1/openSUSE 13）

当操作系统版本高于SUSE 12 SP1或高于openSUSE 13版本时，请修改“/etc/dracut.conf”文件，在add_drivers项中添加virtio的驱动，操作步骤如下：

以SUSE Linux Enterprise Server 12 SP2 (x86_64)为例，如下所示：

1. 执行以下命令，打开“/etc/dracut.conf”文件。

```
vi /etc/dracut.conf
```

2. 按“i”进入编辑模式，在“add_drivers”项中添加virtio的驱动（具体格式要根据操作系统本身的要求来决定）。

```
[root@CTU10000xxxx ~]# vi /etc/dracut.conf  
# additional kernel modules to the default  
add_drivers+="ata_piix ata_generic virtio_blk virtio_scsi virtio_net virtio_pci virtio_ring virtio"
```

- 按“Esc”后，输入:wq，按“Enter”。保存设置并退出“/etc/dracut.conf”文件。

- 执行以下命令，重新生成initrd。

dracut -f /boot/initramfs-文件名

如果引导的虚拟文件系统不是默认的initramfs，则命令为：**dracut -f 实际使用的initramfs文件名或者initrd文件名**。“实际使用的initramfs文件名或者initrd文件名”可在grub.cfg配置（“/boot/grub/grub.cfg”或“/boot/grub2/grub.cfg”或“/boot/grub/grub.conf”，具体路径根据OS不同会有所区别）中获取。

- 如果引导的虚拟文件系统是initramfs，执行以下命令，检查是否已经成功装载了原生的KVM驱动相应模块。

lsinitrd /boot/initramfs-`uname -r`.img | grep virtio

如果引导的虚拟文件系统是initrd，执行如下命令，检查是否已经成功装载了原生的KVM驱动相应模块。

lsinitrd /boot/initrd-`uname -r` | grep virtio

以引导的虚拟文件系统是initrd为例，回显信息如下所示：

```
sluo-ecs-30dc:~ # lsinitrd /boot/initrd-`uname -r` | grep virtio
-rw-r--r-- 1 root root 29335 Oct 26 2016 lib/modules/4.4.21-69-default/kernel/drivers/block/
virtio_blk.ko
-rw-r--r-- 1 root root 57007 Oct 26 2016 lib/modules/4.4.21-69-default/kernel/drivers/net/
virtio_net.ko
-rw-r--r-- 1 root root 32415 Oct 26 2016 lib/modules/4.4.21-69-default/kernel/drivers/scsi/
virtio_scsi.ko
drwxr-xr-x 2 root root 0 Sep 28 10:21 lib/modules/4.4.21-69-default/kernel/drivers/virtio
-rw-r--r-- 1 root root 19623 Oct 26 2016 lib/modules/4.4.21-69-default/kernel/drivers/virtio/virtio.ko
-rw-r--r-- 1 root root 38943 Oct 26 2016 lib/modules/4.4.21-69-default/kernel/drivers/virtio/
virtio_pci.ko
-rw-r--r-- 1 root root 24431 Oct 26 2016 lib/modules/4.4.21-69-default/kernel/drivers/virtio/
virtio_ring.ko
```

说明

如果误将built-in形式存在内核中的驱动添加到initrd或initramfs文件中，不会影响云服务器正常使用，这里全写进去只是为了修改方便，但是使用lsinitrd命令无法检查到。可使用如下方法确定这些驱动是否以built-in形式存在内核中，例如：

```
cat /boot/config-`uname -r` | grep CONFIG_VIRTIO | grep y
```

4.10.8 清除日志文件

清除日志文件、历史记录等，关闭云服务器。

- 执行如下命令，清除冗余key文件：

```
echo > /$path/$to/$root/.ssh/authorized_keys
```

例如：**echo > /root/.ssh/authorized_keys**

```
echo > /$path/$to/$none-root/.ssh/authorized_keys
```

例如：**echo > /home/linux/.ssh/authorized_keys**

- 执行如下命令，清空“/var/log”目录的日志文件：

```
rm -rf /var/log/*
```

说明

删除日志前，请备份需要的应用日志目录及日志文件，防止应用没有日志目录导致启动失败。例如，nginx日志默认路径为/var/log/nginx，若删除了此目录，可能导致nginx应用启动失败。

3. 执行如下命令，清空历史记录：

```
echo > /root/.bash_history
history -c
```

4.11 转换镜像格式

应用场景

云平台支持导入vhd、vmdk、qcow2、raw、vhdx、qcow、vdi、qed、zvhd或zvhd2格式镜像文件。其他镜像文件，需要转换格式后再导入。本节操作指导您使用开源qemu-img工具转换镜像格式。

方案构架

本节提供本地为Windows操作系统和Linux操作系统的转换镜像格式的操作方法。

资源和成本规划

表 4-3 资源和成本规划

资源	资源说明	成本说明
qemu-img	是一款开源的转换镜像格式的工具。 获取方式： https://qemu.weilnetz.de/w64/	免费

约束与限制

- qemu-img镜像格式转换工具支持vhd、vmdk、qcow2、raw、vhdx、qcow、vdi或qed社区格式的镜像的相互转换。
- zvhd和zvhd2是云服务内部自研格式，qemu-img工具无法识别这两种格式的镜像文件。
- vhd格式镜像在执行命令转换格式时请使用vpc代替。
例如，将CentOS 6.9镜像的vhd格式转换为qcow2格式，请执行如下命令：

```
qemu-img convert -p -f vpc -O qcow2 centos6.9.vhd centos6.9.qcow2
```

说明

如遇报错，请删除-f vpc，qemu-img工具将自动识别镜像格式。

本地为 Windows 操作系统

1. 安装qemu-img。
 - a. 下载qemu-img安装包至本地：<https://qemu.weilnetz.de/w64/>。
 - b. 双击setup文件安装qemu-img，以下操作以安装路径为“D:\Program Files\qemu”为例。

2. 配置环境变量。
 - a. 选择“开始 > 计算机”，右键单击“属性”。
 - b. 单击“高级系统设置”。
 - c. 在“系统属性”对话框里，单击“高级 > 环境变量”。
 - d. 在环境变量对话框里，在系统变量部分找到Path，并单击“编辑”。在“变量值”里，添加“D:\Program Files\qemu”，不同的变量值之间以“;”分隔。

说明

如果没有Path变量请新建，并补充Path的变量值为“D:\Program Files\qemu”。

- e. 单击“确定”，保存修改。
3. 验证安装成功。
单击“开始 > 运行”，输入“cmd”后按回车键，在“cmd”窗口输入**qemu-img --help**，如回显信息中出现qemu-img工具的版本信息，即表示安装成功。
4. 转换镜像格式。

- a. 在“cmd”窗口输入如下命令切换文件目录，以安装目录为“D:\Program Files\qemu”为例。

d:

```
cd D:\Program Files\qemu
```

- b. 执行如下命令转换镜像文件格式，以转换vmdk格式为qcow2格式的镜像为例。

```
qemu-img convert -p -f vmdk -O qcow2 centos6.9.vmdk centos6.9.qcow2
```

上述命令中各参数对应的说明如下：

- -p: 表示镜像转换的进度。
- -f后面为源镜像格式。
- -O (必须是大写) 后面的参数由如下3个部分组成：转换出来的镜像格式 + 源镜像文件名称 + 目标文件名称。

转换完成后，目标文件会出现在源镜像文件所在的目录下。

回显信息如下所示：

```
# qemu-img convert -p -f vmdk -O qcow2 centos6.9.vmdk centos6.9.qcow2  
(100.00/100%)
```

- c. 执行如下命令，查询转换后的qcow2格式镜像文件的详细信息。

```
qemu-img info centos6.9.qcow2
```

回显信息如下所示：

```
# qemu-img info centos6.9.qcow2  
image: centos6.9.qcow2  
file format: qcow2  
virtual size: 1.0G (1073741824 bytes)  
disk size: 200K  
cluster_size: 65536  
Format specific information:  
  compat: 1.1  
  lazy_refcounts: false
```

本地为 Linux 操作系统

1. 安装qemu-img。
 - Ubuntu、Debian系列操作系统，请执行如下命令：
apt install qemu-img
 - CentOS、Red Hat、Oracle系列操作系统，请执行如下命令：
yum install qemu-img
 - SUSE、openSUSE系列操作系统，请执行如下命令：
zypper install qemu-img

2. 执行如下命令，验证安装成功。

qemu-img -v

如回显信息中出现qemu-img工具的版本信息和帮助手册，即表示安装成功。以CentOS 7为例，回显信息如下所示：

```
[root@CentOS7 ~]# qemu-img -v
qemu-img version 1.5.3, Copyright (c) 2004-2008 Fabrice Bellard
usage: qemu-img command [command options]
QEMU disk image utility

Command syntax:
check [-q] [-f fmt] [--output=ofmt] [-r [leaks | all]] [-T src_cache] filename
create [-q] [-f fmt] [-o options] filename [size]
commit [-q] [-f fmt] [-t cache] filename
compare [-f fmt] [-F fmt] [-T src_cach]
```

3. 转换镜像格式，以CentOS 7操作系统中转换vmdk格式为qcow2格式的镜像为例。

- a. 执行如下命令转换镜像文件格式。

```
qemu-img convert -p -f vmdk -O qcow2 centos6.9.vmdk
centos6.9.qcow2
```

上述命令中各参数对应的说明如下：

- -p标识转换的进度条。
- -f后面为源镜像格式。
- -O（必须是大写）后面的参数为转换出来的镜像格式 + 源镜像文件名称 + 目标文件名称。

转换完成后，目标文件会出现在源镜像文件所在的目录下。

回显信息如下所示：

```
[root@CentOS7 home]# qemu-img convert -p -f vmdk -O qcow2 centos6.9.vmdk
centos6.9.qcow2
(100.00/100%)
```

- b. 执行如下命令，查询转换后的qcow2格式镜像文件的详细信息。

```
qemu-img info centos6.9.qcow2
```

回显信息如下所示：

```
[root@CentOS7 home]# qemu-img info centos6.9.qcow2
image: centos6.9.qcow2
file format: qcow2
virtual size: 1.0G (1073741824 bytes)
disk size: 200K
cluster_size: 65536
Format specific information:
  compat: 1.1
  lazy refcounts: false
```


典型应用举例

- 应用场景

用户从VMware平台导出一个pre-allocated格式的镜像文件（pre-allocated格式是VMDK monolithic Flat子格式类型），pre-allocated格式镜像依赖2个文件“xxxx.vmdk”和“xxxx-flat.vmdk”（“xxxx.vmdk”是一个配置文件，“xxxx-flat.vmdk”是实际数据文件），这两种格式的镜像文件不能直接导入到云平台，需提前转换成通用的vmdk或qcow2格式才能导入。

以下以镜像文件格式为centos6.9-64bit-flat.vmdk和centos6.9-64bit.vmdk为例，通过qemu-img工具转换镜像格式。

- 操作步骤

1. 执行如下命令，查询镜像文件的详细信息。

```
ls -lh centos6.9-64bit*
```

```
qemu-img info centos6.9-64bit.vmdk
```

```
qemu-img info centos6.9-64bit-flat.vmdk
```

回显信息如下所示：

```
[root@CentOS7 tmp]# ls -lh centos6.9-64bit*
-rw-r--r--. 1 root root 10G Jun 13 05:30 centos6.9-64bit-flat.vmdk
-rw-r--r--. 1 root root 327 Jun 13 05:30 centos6.9-64bit.vmdk
[root@CentOS7 tmp]# qemu-img info centos6.9-64bit.vmdk
image: centos6.9-64bit.vmdk
file format: vmdk
virtual size: 10G (10737418240 bytes)
disk size: 4.0K
Format specific information:
  cid: 3302005459
  parent cid: 4294967295
  create type: monolithicFlat
  extents:
    [0]:
      virtual size: 10737418240
      filename: centos6.9-64bit-flat.vmdk
      format: FLAT
[root@CentOS7 tmp]# qemu-img info centos6.9-64bit-flat.vmdk
image: centos6.9-64bit-flat.vmdk
file format: raw
virtual size: 10G (10737418240 bytes)
disk size: 0
```

说明

从回显信息中看出，centos6.9-64bit.vmdk文件的格式为vmdk，centos6.9-64bit-flat.vmdk文件的格式为raw，在执行镜像格式转换命令时，源镜像文件必须使用centos6.9-64bit.vmdk（详见步骤3）。

2. 执行如下命令，查看pre-allocated格式的镜像文件的相关配置信息。

```
cat centos6.9-64bit.vmdk
```

回显信息如下所示：

```
[root@CentOS7 tmp]# cat centos6.9-64bit.vmdk
# Disk DescriptorFile
version=1
CID=c4d09ad3
parentCID=ffffff
createType="monolithicFlat"

# Extent description
RW 20971520 FLAT "centos6.9-64bit-flat.vmdk" 0

# The Disk Data Base
```

```
#DDB
```

```
ddb.virtualHWVersion = "4"
ddb.geometry.cylinders = "20805"
ddb.geometry.heads = "16"
ddb.geometry.sectors = "63"
ddb.adapterType = "ide"
```

3. 将centos6.9-64bit-flat.vmdk和centos6.9-64bit.vmdk放在同一个目录下，执行如下命令，通过qemu-img工具转换镜像格式为qcow2。

```
[root@CentOS7 tmp]# qemu-img convert -p -f vmdk -O qcow2 centos6.9-64bit.vmdk
centos6.9-64bit.qcow2
(100.00/100%)
```

4. 执行如下命令，查询转换后的qcow2格式镜像文件的详细信息。

```
qemu-img info centos6.9-64bit.qcow2
```

回显信息如下所示：

```
[root@CentOS7 tmp]# qemu-img info centos6.9-64bit.qcow2
image: centos6.9-64bit.qcow2
file format: qcow2
virtual size: 10G (10737418240 bytes)
disk size: 200K
cluster_size: 65536
Format specific information:
  compat: 1.1
  lazy refcounts: false
```

5 Windows 操作系统相关操作

5.1 设置网卡属性为 DHCP（Windows）

操作场景

通过云服务器或者外部镜像文件创建私有镜像时，如果云服务器或镜像文件所在虚拟机的网络配置是静态IP地址时，您需要修改网卡属性为DHCP，以使私有镜像发放的新云服务器可以动态获取IP地址。

本节以Windows Server 2008 R2操作系统为例。其他操作系统配置方法略有区别，请参考对应操作系统的相关资料进行操作，文档中不对此进行详细说明。

说明

使用外部镜像文件创建私有镜像时，设置网卡属性操作需要在虚拟机内部完成，建议您在原平台的虚拟机实施修改后，再导出镜像。

前提条件

已登录创建Windows私有镜像所使用的云服务器。

登录云服务器的详细操作请参见《弹性云服务器用户指南》。

操作步骤

1. 在云服务器上选择“开始 > 控制面板”。
2. 单击“网络和Internet”。
3. 单击“网络和共享中心”，如[图5-1](#)所示。

图 5-1 网络和共享中心



4. 选择您已经设置为静态IP的连接。以单击“本地连接 2”为例，如图5-2所示。

图 5-2 本地连接 2 状态

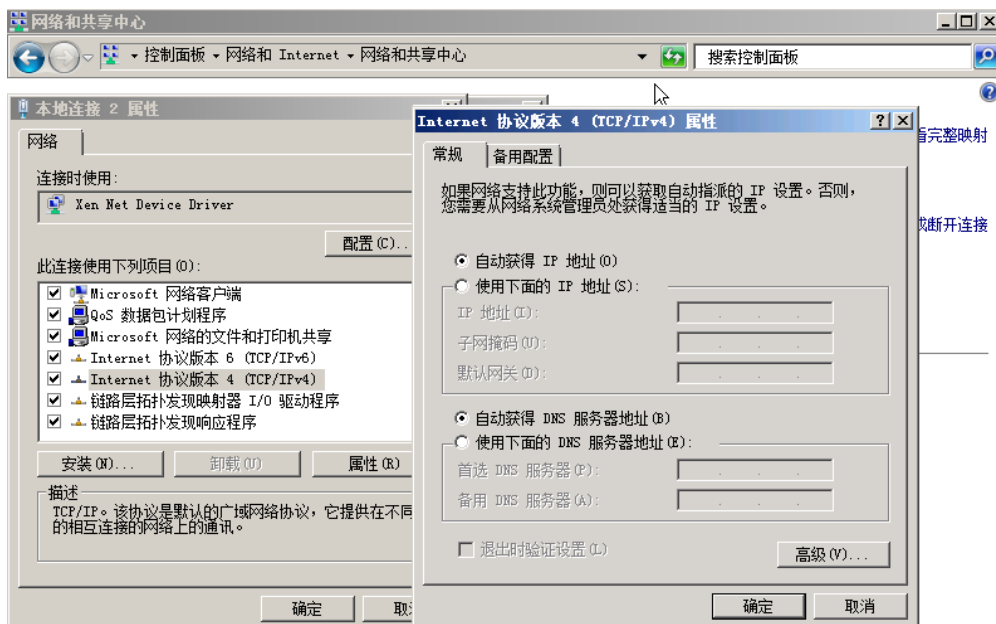


5. 单击“属性”，选择您配置的协议版本。
6. 在“常规”页签中勾选“自动获得IP地址”和“自动获得DNS服务器地址”，单击“确定”，如图5-3所示。配置后系统会自动获取IP地址。

说明

建议您记录原有网络信息，以便后续可以修改回原有配置。

图 5-3 配置网络获取 IP 方式



5.2 开启远程桌面连接功能

操作场景

对于需要使用Windows远程桌面连接方式进行访问的云服务器，需要在制作私有镜像时开启远程桌面连接功能。GPU优化型云服务器必须开启该功能。

说明

使用外部镜像文件制作私有镜像时，开启远程桌面连接操作需要在虚拟机内部完成，建议您在原平台的虚拟机实施修改后，再导出镜像。

前提条件

已登录创建Windows私有镜像所使用的云服务器。

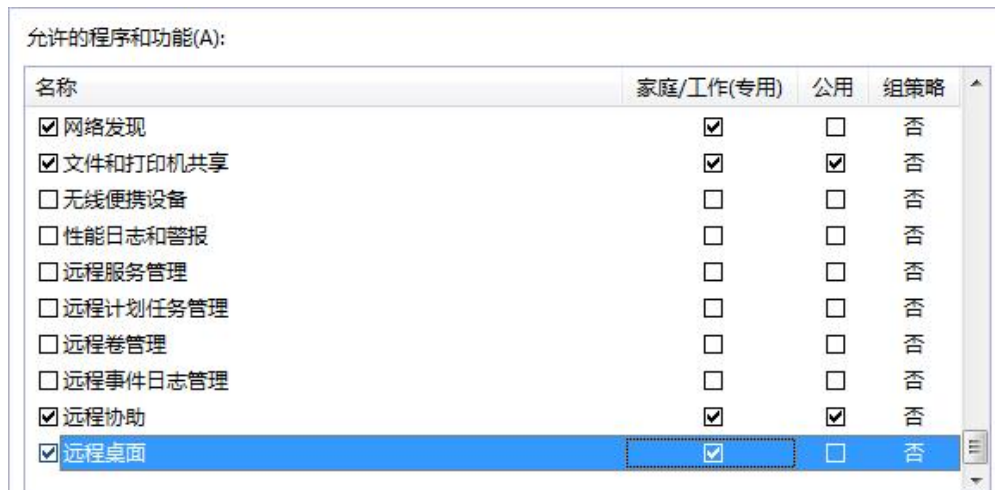
登录云服务器的详细操作请参见《弹性云服务器用户指南》。

操作步骤

1. 开启远程桌面连接功能之前，建议先将云服务器的分辨率设置为1920×1080。
设置方法：在云服务器操作系统单击“开始 > 控制面板”，在“外观和个性化”区域单击“调整屏幕分辨率”，然后在“分辨率”下拉框中选择合适的值即可。
2. 单击“开始”，右键单击“计算机”，选择“属性”，进入“计算机属性”区域框。
3. 在左侧界面中，单击“远程设置”，进入“远程桌面”区域框。
4. 选择“允许运行任意版本远程桌面的计算机连接”。
5. 单击“确定”，返回“计算机属性”界面。

6. 选择“开始 > 控制面板”，打开“Windows防火墙”。
7. 在左侧选择“允许程序或功能通过Windows防火墙”。
8. 根据用户网络的需要，配置“远程桌面”可以在哪种网络环境中通过Windows防火墙如图5-4所示，然后单击下方的“确定”完成配置。

图 5-4 配置“远程桌面”网络环境



5.3 安装并配置 Cloudbase-Init 工具

操作场景

为了保证使用私有镜像创建的新云服务器可以通过“用户数据注入”功能注入初始化自定义信息（例如为云服务器设置登录密码），建议您在创建私有镜像前安装 Cloudbase-Init 工具。

- 不安装 Cloudbase-Init 工具，将无法对云服务器进行自定义配置，只能使用镜像原有密码登录云服务器。
- 使用公共镜像创建的云服务器，默认已经安装 Cloudbase-Init，不需要执行安装及配置操作。
- 使用外部镜像文件创建的云服务器，请按照指导安装及配置 Cloudbase-Init。

📖 说明

Cloudbase-Init 为开源软件，如果安装版本有安全漏洞，建议更新至最新版本。

前提条件

- 已为云服务器绑定弹性公网 IP。
- 已登录到云服务器。
- 服务器的网卡属性为 DHCP 方式。
- 已安装一键式重置密码插件。

因为安装 Cloudbase-Init 工具的操作步骤中如果有重启云服务器的操作，可能导致密码被修改为一个随机密码，所以需要安装一键式重置密码插件重置密码。操作步骤请参考[安装一键式重置密码插件（Windows）](#)。

安装 Cloudbase-Init 工具

1. 在Windows操作系统中，单击“开始”，选择“控制面板 > 程序 > 程序和功能”查看是否安装Cloudbase-Init最新1.1.2版本。
 - 已安装并为最新版本：直接执行[配置Cloudbase-Init工具](#)。
 - 已安装但不是最新版本：卸载Cloudbase-Init工具后，执行以下安装操作步骤。
 - 未安装：执行以下安装操作步骤。
2. 操作系统是否为Windows桌面版。
 - 是，执行3。
 - 否，若操作系统为Windows Server版本，请执行4。
3. 如果操作系统是Windows桌面版，如Windows 7或者Windows 10，那么需要在安装Cloudbase-Init前确保Administrator账号未禁用。以Windows 7为例，具体操作请以实际为准。
 - a. 在操作系统中单击“开始”，选择的“控制面板 > 系统和安全 > 管理工具”。
 - b. 双击“计算机管理”。
 - c. 选择“系统工具 > 本地用户和组 > 用户”。
 - d. 右键单击“Administrator”，选择“属性”。
 - e. 确认已取消勾选“账户已禁用”选项。
4. 下载Cloudbase-Init工具安装包。

根据Windows操作系统的不同位数，您需要下载不同版本的Cloudbase-Init工具安装包。Cloudbase官网：<http://www.cloudbase.it/cloud-init-for-windows-instances/>。

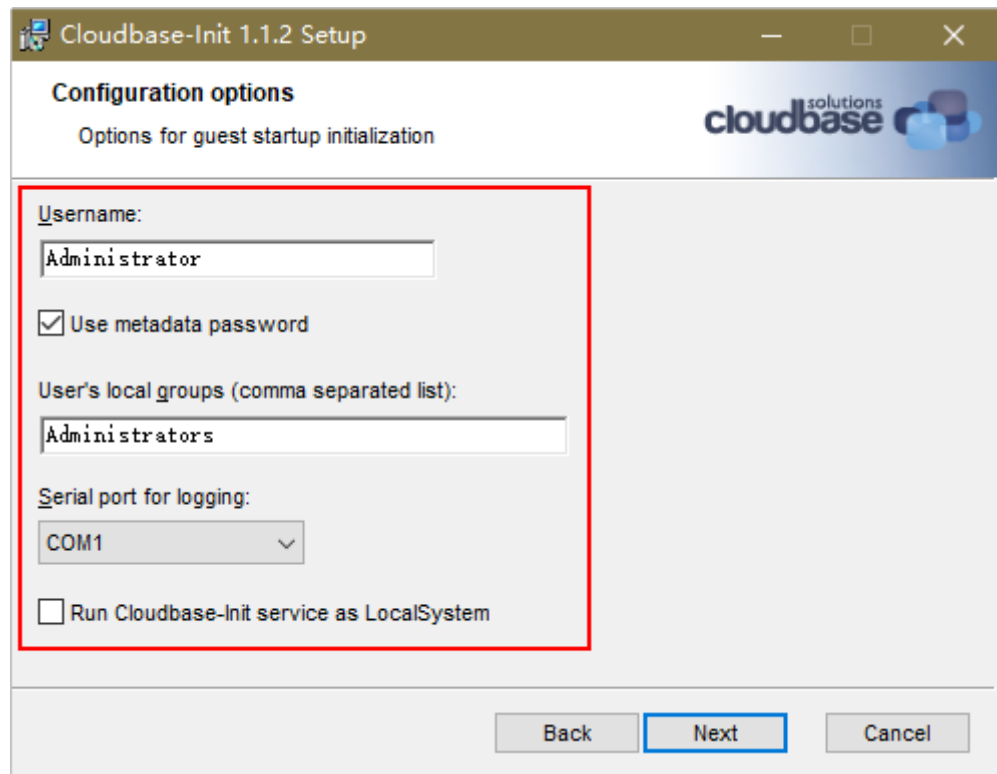
稳定版本获取路径：

 - 64位：https://www.cloudbase.it/downloads/CloudbaseInitSetup_Stable_x64.msi
 - 32位：https://www.cloudbase.it/downloads/CloudbaseInitSetup_Stable_x86.msi
5. 打开Cloudbase-Init工具安装包开始安装。
6. 单击“Next”。
7. 勾选“I accept the terms in the License Agreement”，单击“Next”。
8. 使用Cloudbase-Init默认安装路径进行安装，单击“Next”。
9. 在“Configuration options”窗口中，设置用户名为“Administrator”，日志输出串口选择“COM1”，且不勾选“Run Cloudbase-Init service as LocalSystem”。如[图5-5](#)所示。

说明

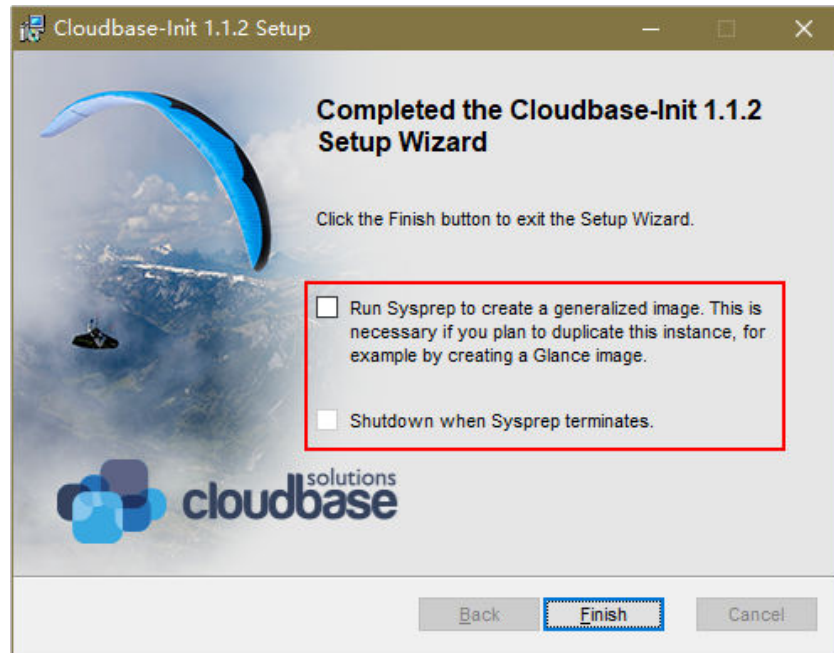
图片中的版本号仅供参考，请以实际情况为准。

图 5-5 设置参数



10. 单击“Next”。
11. 单击“Install”。
12. 在“Files in Use”窗口保留默认勾选“Close the application and attempt to restart them”，单击“OK”。
13. 操作系统是否为Windows桌面版。
 - 是，执行15。
 - 否，执行14。
14. 在“Completed the Cloudbase-Init Setup Wizard”窗口，请勿勾选“Run Sysprep to create a generalized Image. This is necessary if you plan to duplicate this instance, for example by creating a Glance image”及“Shutdown when Sysprep terminate”。如图5-6所示。

图 5-6 完成安装



说明

图片中的版本号仅供参考，请以实际情况为准。

15. 单击“Finish”。

配置 Cloudbase-Init 工具

1. 在Cloudbase-Init安装路径下的配置文件“C:\Program Files\Cloudbase Solutions\Cloudbase-Init\conf\cloudbase-init.conf”中执行以下操作：
 - a. 在配置文件最后一行，增加配置项“netbios_host_name_compatibility=false”，使Windows系统的hostname长度支持到63个字符。

说明

NetBIOS长度受Windows系统本身的限制还支持小于等于15个字符。

- b. 增加配置项“metadata_services=cloudbaseinit.metadata.services.httpservice.HttpService”，配置agent访问OpenStack数据源。
- c. 添加plugins配置项，设置要加载的模块，以“,”为分隔符，加粗的部分为模块关键字。

- 如下参数配置为官方默认加载模块，您可以根据业务需要选择配置全部或部分模块：

```
plugins=cloudbaseinit.plugins.common.localscripts.LocalScriptsPlugin,cloudbaseinit.plugins.common.mtu.MTUPlugin,cloudbaseinit.plugins.windows.createuser.CreateUserPlugin,cloudbaseinit.plugins.common.setuserpassword.SetUserPasswordPlugin,cloudbaseinit.plugins.common.sshpublickeys.SetUserSSHPublicKeysPlugin,cloudbaseinit.plugins.common.sethostname.SetHostNamePlugin,cloudbaseinit.plugins.windows.extendvolumes.ExtendVolumesPlugin,cloudbaseinit.plugins.common.userdata.UserDataPlugin,cloudbaseinit.plugins.windows.licensing.WindowsLicensingPlugin
```

各模块功能如下所述：

- LocalScriptsPlugin: 设置脚本执行
- MTUPlugin: 设置MTU网络接口
- CreateUserPlugin: 创建用户
- SetUserPasswordPlugin: 设置密码
- SetUserSSHPublicKeysPlugin: 设置密钥
- SetHostNamePlugin: 设置主机名
- ExtendVolumesPlugin: 磁盘扩容
- UserDataPlugin: 用户数据注入
- WindowsLicensingPlugin: 激活window instance

说明

若您创建实例后需要在系统内修改主机名，且业务对主机名变化敏感，不建议配置SetHostNamePlugin模块。

- 可选加载模块，请参考如下配置（可选）：
`plugins=cloudbaseinit.plugins.windows.winrmlistener.ConfigWinRMListenerPlugin,cloudbaseinit.plugins.windows.winrmcertificateauth.ConfigWinRMCertificateAuthPlugin`
 各模块功能如下所述：
 - ConfigWinRMListenerPlugin: 设置远程登录监听
 - ConfigWinRMCertificateAuthPlugin: 设置远程登录无密码认证

注意

加载模块对应的插件WinRM使用了弱密码算法，可能存在安全隐患，为保证系统安全，不建议您添加此插件。

- d. （可选）增加如下配置项，配置获取metadata的重试次数和间隔。
`retry_count=40`
`retry_count_interval=5`
- e. （可选）增加如下配置项，防止Windows添加默认路由导致metadata网络不通。
`[openstack]`
`add_metadata_private_ip_route=False`
- f. （可选）当Cloudbase-Init为0.9.12及以上版本时，用户可以自定义配置密码长度。
 操作方法：修改配置项“user_password_length”的值，完成密码长度的自定义配置。
- g. （可选）选择密码注入方式首次登录时，系统默认强制用户修改登录密码，若用户根据个人意愿，不需要修改首次登录使用的密码时，可关闭此功能。
 操作方法：增加配置项“first_logon_behaviour=no”。
- h. （可选）增加Cloudbase-Init配置项，防止虚拟机无法使用BIOS时间同步。
 操作方法：在“cloudbase-init.conf”中添加相应的配置项“real_time_clock_utc=true”。

📖 说明

BIOS时间同步功能是通过增加注册表项“RealTimeIsUniversal=1”实现。如果不添加配置项“**real_time_clock_utc=true**”，Cloudbase-Init会将注册表自动修改为默认值0，将导致windows虚拟机重启后无法使用BIOS时间同步。

2. 为了防止镜像中DHCP租期过长导致创建的云服务器无法正确的获取地址，用户需要释放当前的DHCP地址。

在Windows命令行中，执行以下命令释放当前的DHCP地址。

ipconfig /release

📖 说明

此操作会中断网络，对云服务器的使用会产生影响。当云服务器再次开机后，网络会自动恢复。

3. 使用Windows操作系统云服务器制作镜像时，需修改云服务器SAN策略为OnlineAll类型。否则可能导致使用镜像创建云服务器时磁盘处于脱机状态。

Windows操作系统SAN策略分为三种类型：OnlineAll、OfflineShared、OfflineInternal

表 5-1 Windows 操作系统 SAN 策略类型

类型	说明
OnlineAll	表示所有新发现磁盘都置于在线模式。
OfflineShared	表示所有共享总线上（比如FC、iSCSI）的新发现磁盘都置于离线模式，非共享总线上的磁盘都置于在线模式。
OfflineInternal	表示所有新发现磁盘都置于离线模式。

- a. 运行cmd.exe，执行以下命令，使用DiskPart工具来查询云服务器当前的SAN策略。

diskpart

- b. 执行以下命令查看云服务器当前的SAN策略。

san

- 如果SAN策略为OnlineAll，请执行**exit**命令退出DiskPart。
 - 否，请执行步骤**3.c**。
- c. 执行以下命令修改云服务器SAN策略为OnlineAll。

san policy=onlineall

5.4 安装一键式重置密码插件（Windows）

操作场景

为了保证使用私有镜像创建的新云服务器可以实现一键式重置密码功能，建议您在创建私有镜像前安装密码重置插件CloudResetPwdAgent。

- 使用公共镜像创建的云服务器，默认已经安装密码重置插件，不需要执行安装操作。
- 使用外部镜像文件创建的云服务器，请按照指导安装密码重置插件。

📖 说明

可选操作。如需使用控制台提供的一键式重置密码功能，请按照本节指导安装插件。

安装须知

- 用户自行决定是否安装CloudResetPwdAgent插件，使云服务器具备一键式重置密码功能。
- 支持安装一键式重置密码插件的操作系统版本如[表 支持安装一键式重置密码插件的操作系统版本](#)所示。

表 5-2 支持安装一键式重置密码插件的操作系统版本

操作系统类型	操作系统版本
Windows	Windows Server 2016 Datacenter 64bit Windows Server 2012 R2 Standard 64bit Windows Server 2012 R2 Datacenter 64bit Windows Server 2008 R2 Enterprise 64bit Windows Server 2008 SP2 Enterprise 64bit Windows Server 2008 R2 Datacenter 64bit Windows Server 2008 R2 Standard 64bit Windows Server 2008 R2 WEB 64bit

前提条件

- 云服务器的状态为“运行中”。
- 需保证C盘可写入，且剩余空间大于300MB。
- 云服务器使用的VPC网络DHCP不能禁用。
- 云服务器网络正常通行。
- 云服务器安全组出方向规则满足如下要求：

- 协议：TCP
- 端口范围：80
- 远端地址：169.254.0.0/16

如果您使用的是默认安全组出方向规则，则已经包括了如上要求，可以正常初始化。默认安全组出方向规则为：

- 协议：ALL
- 端口范围：ALL
- 远端地址：0.0.0.0/16

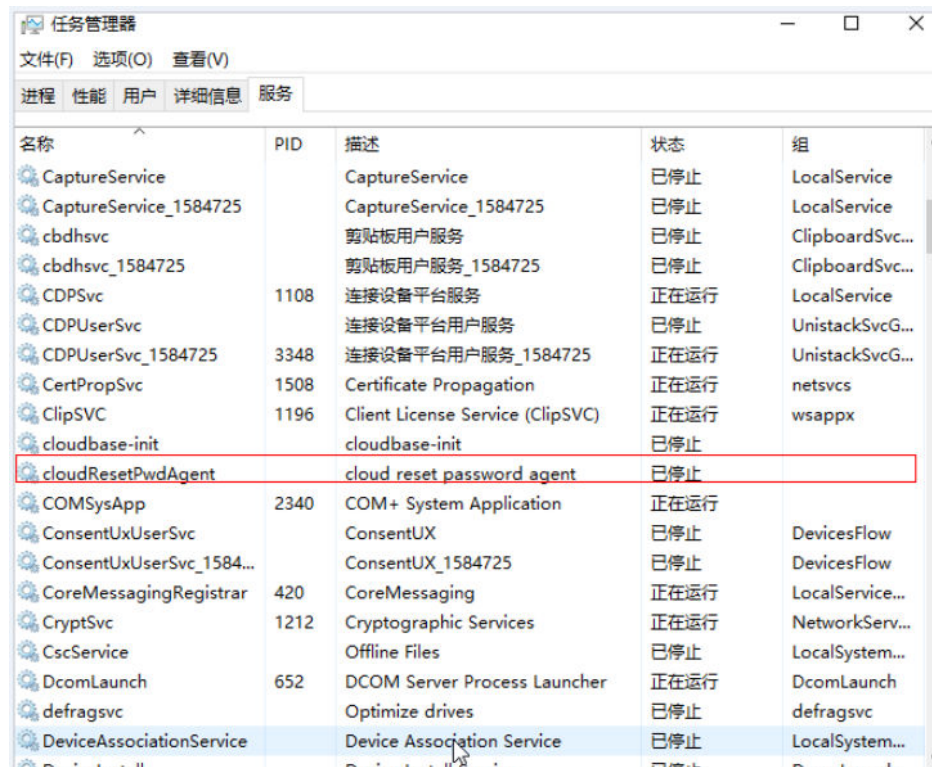
操作步骤

步骤1 登录弹性云服务器。

步骤2 检查弹性云服务器是否已安装密码重置插件CloudResetPwdAgent。检查方法如下：

查看任务管理器，如果找到cloudResetPwdAgent服务，如图 [安装插件成功](#)所示，表示弹性云服务器已安装密码重置插件。

图 5-7 安装插件成功



- 是，结束。
- 否，执行**3**。

步骤3 下载一键式重置密码插件CloudResetPwdAgent.zip。

安装一键式重置密码插件对插件的具体放置目录无特殊要求，请您自定义。

该插件放置在OBS桶中，请联系管理员获取OBS桶的路径。

步骤4 解压软件包CloudResetPwdAgent.zip。

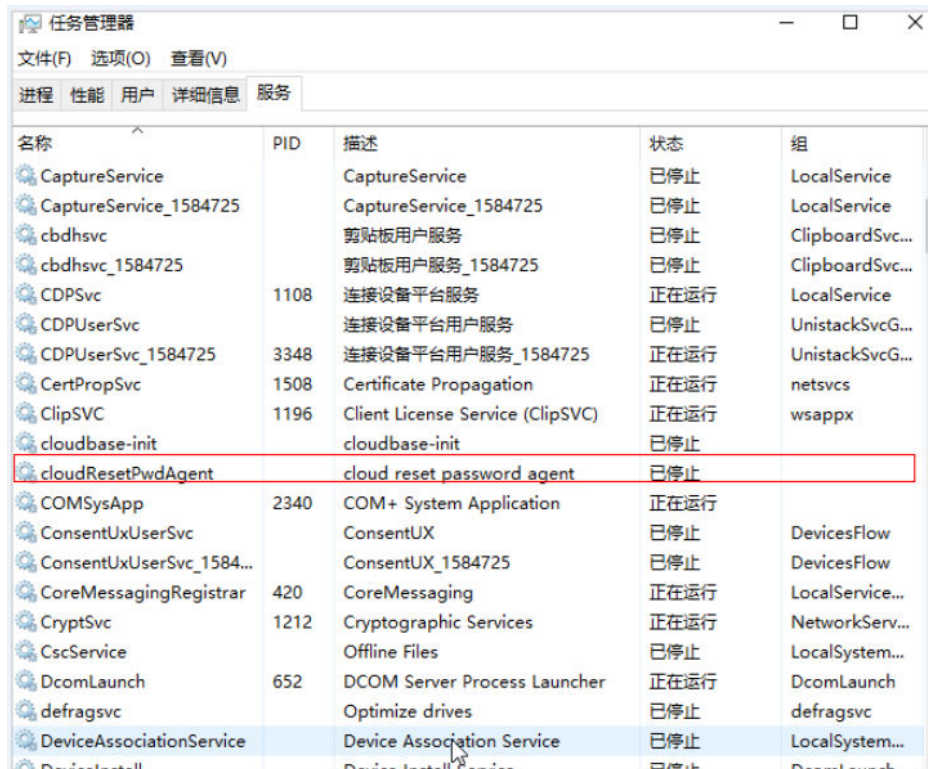
安装一键式重置密码插件对插件的解压目录无特殊要求，请您自定义。

步骤5 安装一键式重置密码插件。

1. 双击“CloudResetPwdAgent.Windows”文件夹下的“setup.bat”文件。
安装密码重置插件。
2. 查看任务管理器，检查密码重置插件是否安装成功。

如果在任务管理器中查找到了cloudResetPwdAgent服务，如图**5-8**所示，表示安装成功，否则安装失败。

图 5-8 安装插件成功



说明

如果密码重置插件安装失败，请检查安装环境是否符合要求，并重试安装操作。

----结束

5.5 执行 Sysprep

操作场景

执行Sysprep操作可以确保在云服务器加入域后SID唯一。

在Cloudbase-Init安装完成后，请确认云服务器是否需要加入域，或是否需要保证SID唯一。如果是，请按照本节指导执行Sysprep。

前提条件

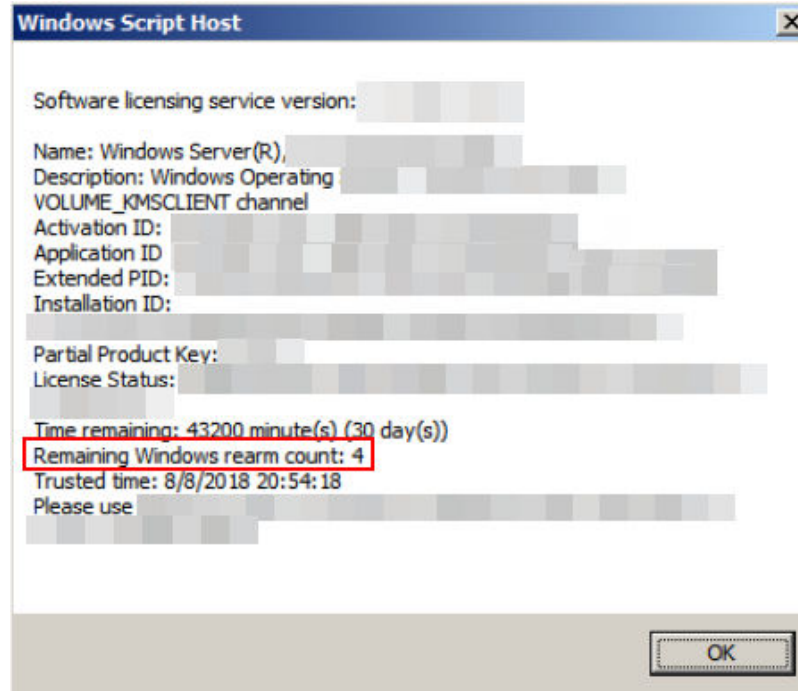
- 请使用Administrator用户执行Sysprep。
- 一个全新激活的Windows云服务器每次最多执行一次Sysprep，不能连续多次执行。
- 如果云服务器是由镜像文件创建而来，那么仅支持使用原镜像自带的Sysprep版本，且Sysprep必须始终从“%WINDIR%\system32\sysprep”目录运行。
- Windows必须保证是正版激活状态，并且必须保证剩余Windows重置计数大于等于1，否则不能执行Sysprep封装。

在Windows命令行中输入以下命令，在“Windows Script Host”弹出窗中查看还能执行Sysprep的次数。

slmgr.vbs /dlv

当“剩余Windows重置计数”的值为0时，无法执行Sysprep。

图 5-9 Windows Script Host



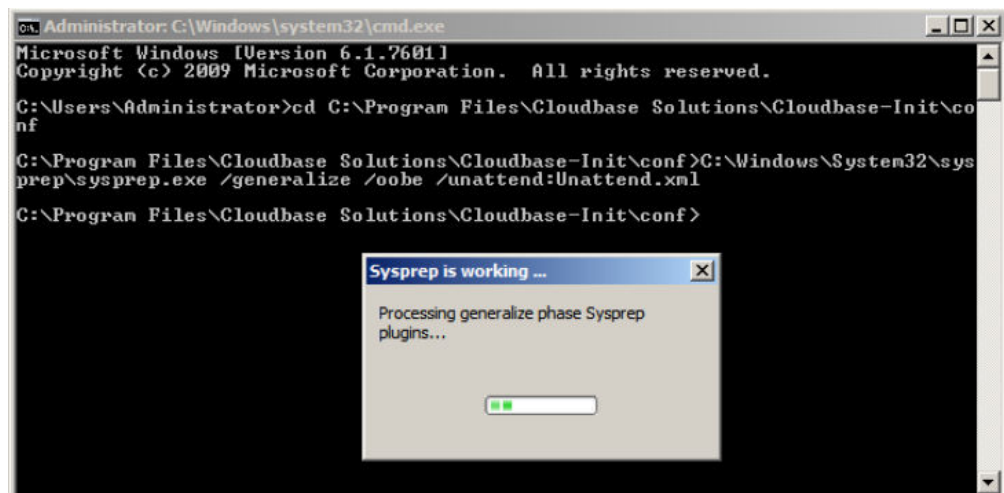
操作步骤

1. 进入Cloudbase-Init工具安装路径。
以Cloudbase-Init工具安装在“C:\Program Files\Cloudbase Solutions\”目录下为例。请切换至C盘根目录下，执行以下命令，进入安装目录。
cd C:\Program Files\Cloudbase Solutions\Cloudbase-Init\conf
2. 执行以下命令，对Windows系统进行封装。
C:\Windows\System32\sysprep\sysprep.exe /generalize /oobe /unattend:Unattend.xml

注意

- 请务必在执行该命令时，包含“/unattend:Unattend.xml”，否则您当前云服务器的用户名密码等重要配置信息会被重置，后续使用该Windows私有镜像创建的云服务器启动后仍然需要手动执行操作系统设置。
- 执行完该命令后，云服务器会自动关机。请在关机后使用该云服务器创建镜像，可以保证后续使用该镜像创建的云服务器SID唯一。如果重新启动已经执行过Sysprep操作的Windows云服务器，则执行的Sysprep操作仅对当前云服务器生效，创建镜像前需要重新执行Sysprep操作。
- 对于Windows Server 2012以及Windows Server 2012 R2操作系统，当云服务器执行完Sysprep操作后，云服务器的Administrator账号密码会被清除，请您登录云服务器后重新手动设置Administrator账号的密码，此时在管理控制台界面中设置的云服务器密码将无效，请您妥善保管重新设置的密码。
- 使用的Windows操作系统需要采用域账号登录时，请务必在创建私有镜像前执行Sysprep操作。执行Sysprep操作带来的影响请参考[Windows操作系统制作私有镜像为什么要执行Sysprep操作?](#)。
- Windows云服务器中的Cloudbase-Init账户为Cloudbase-Init代理程序的内置账户，用于云服务器启动的时候获取元数据并执行相关配置。如果用户修改、删除此账户或者卸载Cloudbase-Init代理程序会导致通过此云服务器创建的Windows私有镜像所生成新的云服务器初始化的自定义信息注入失败。因此，不建议修改或删除Cloudbase-Init账户。

图 5-10 执行 Sysprep 操作

**后续操作**

1. 使用执行完Sysprep操作后的云服务器创建私有镜像，具体操作请参考[通过云服务器创建Windows系统盘镜像](#)。
2. 使用该镜像即可批量创建弹性云服务器，且所有新创建的云服务器具有唯一的SID。

执行以下命令查询云服务器SID。

```
whoami /user
```


图 5-11 执行 Sysprep 前云服务器 SID

```
C:\Users\Administrator>whoami /user
USER INFORMATION
-----
User Name                SID
-----
ecs-330f\administrator  S-1-5-21-1324385262-1554666476-1954780781-500
```

图 5-12 执行 Sysprep 后云服务器 SID

```
C:\Users\Administrator>whoami /user
USER INFORMATION
-----
User Name                SID
-----
win-1194so2rqsk\administrator  S-1-5-21-2271228291-953988671-972520728-500
```

6 Linux 操作系统相关操作

6.1 设置网卡属性为 DHCP (Linux)

操作场景

通过云服务器或者外部镜像文件创建私有镜像时，如果云服务器或镜像文件所在虚拟机的网络配置是静态IP地址时，您需要修改网卡属性为DHCP，以使私有镜像发放的新云服务器可以动态获取IP地址。

不同操作系统配置方法略有区别，请参考对应操作系统的相关资料进行操作。

📖 说明

使用外部镜像文件创建私有镜像时，设置网卡属性操作需要在虚拟机内部完成，建议您在原平台的虚拟机实施修改后，再导出镜像。

前提条件

已登录创建Linux私有镜像所使用的云服务器。

登录云服务器的详细操作请参见《弹性云服务器用户指南》。

Ubuntu 18 及以后版本操作步骤

1. 在云服务器上执行 `vi /etc/netplan/01-netcfg.yaml` 命令，使用vi编辑器打开“/etc/netplan/01-netcfg.yaml”文件，查看网卡是否为DHCP方式，即 `dhcp4` 为 `true`。

- 信息显示所有网卡的网络配置 `dhcp4` 为 `true` 时，输入:q退出编辑器，操作结束。

```
network:
  version:2
  renderer:NetworkManager
  ethernets:
    eth0:
      dhcp4: true
```

- 信息显示网卡的网络配置的 `dhcp4` 为 `no` 并且配置了静态IP地址时，请执行下一步。

```
network:
  version:2
```

```
renderer:NetworkManager
ethernets:
  eth0:
    dhcp4: no
    addresses: [192.168.1.109/24]
    gateway4: 192.168.1.1
    nameservers:
      addresses: [8.8.8.8,114.114.114.114]
```

2. 输入“i”进入编辑模式。

删除静态IP配置，然后设置dhcp4为true。您也可以使用“#”注释静态IP设置的相关内容。

```
network:
  version:2
  renderer:NetworkManager
  ethernets:
    eth0:
      dhcp4: true #设置dhcp4为true
      #dhcp4: no #删除或使用“#”注释此部分静态IP设置
      #addresses: [192.168.1.109]
      #gateway4: 192.168.1.1
      #nameservers:
      # addresses: [8.8.8.8,114.114.114.114]
```

3. 如果您有多个网卡，请将剩余网卡按照上述方法设置dhcp4为true。

```
network:
  version:2
  renderer:NetworkManager
  ethernets:
    eth0:
      dhcp4: true
    eth1:
      dhcp4: true
    eth2:
      dhcp4: true
    eth3:
      dhcp4: true
```

4. 按“ESC”后，输入:wq，按“Enter”回车键保存设置并退出编辑器。
5. 执行netplan apply命令，使配置生效。

Ubuntu 16.04 操作步骤

1. 在云服务器上执行以下命令，使用vi编辑器打开“/etc/network/interfaces”文件，查看网卡的网络配置。

vi /etc/network/interfaces

- 信息显示所有网卡的网络配置为DHCP模式时，无需重复设置网卡属性，输入:q退出编辑器。

```
auto lo
iface lo inet loopback
auto eth0
iface eth0 inet dhcp

auto eth1
iface eth1 inet dhcp
```

- 信息显示网卡的网络配置为静态IP地址时，请执行2。

```
auto lo
iface lo inet loopback
auto eth0
#iface eth0 inet dhcp
iface eth0 inet static
address 192.168.1.109
netmask 255.255.255.0
gateway 192.168.1.1
```

- 按“i”进入编辑模式。
- 删除静态IP设置的相关内容，然后将对应的网卡设置为DHCP方式。
您也可以使用“#”注释掉静态IP设置的相关内容。

```
auto lo
iface lo inet loopback
auto eth0
iface eth0 inet dhcp
```

如果您有多个网卡，请将剩余网卡按照上述方法设置为DHCP方式。

```
auto lo
iface lo inet loopback
auto eth0
iface eth0 inet dhcp
auto eth1
iface eth1 inet dhcp
```

- 按“ESC”后，输入:wq，按“Enter”。
保存设置并退出编辑器。

相关操作

您还可以配置DHCP持续获取IP地址能力，根据云服务器操作系统选择对应的操作。

- CentOS系列/EulerOS系列：使用vi编辑器在“/etc/sysconfig/network-scripts/ifcfg-ethX”配置文件中添加PERSISTENT_DHCLIENT="y"。
- SUSE系列：使用vi编辑器将“/etc/sysconfig/network/dhcp”配置文件中的DHCLIENT_USE_LAST_LEASE设置为no。
- Ubuntu 12.04及以上版本：将dhclient升级为ISC dhclient 4.2.4以支持网卡持续DHCP获取IP地址能力。请自行安装isc-dhcp-server进行升级。

6.2 清理网络规则文件

操作场景

为了避免使用私有镜像创建的新云服务器发生网卡名称漂移，在创建私有镜像时，需要清理云服务器或镜像文件所在虚拟机的网络规则文件。

说明

使用外部镜像文件制作私有镜像时，清理网络规则文件操作需要在虚拟机内部完成，建议您在原平台的虚拟机实施修改后，再导出镜像。

前提条件

云服务器已安装操作系统和virtio驱动。

操作步骤

- 执行以下命令，查看网络规则目录下的文件。
ls -l /etc/udev/rules.d
- 执行以下命令，删除网络规则目录下，文件名同时包含**persistent**和**net**的规则文件。
例如：

```
rm /etc/udev/rules.d/30-net_persistent-names.rules
```

```
rm /etc/udev/rules.d/70-persistent-net.rules
```

以上命令中斜体部分会根据用户的实际环境有区别。

📖 说明

对于CentOS 6系列的镜像，为避免网卡名发生漂移，您需要创建一个空的rules配置文件。

示例：

```
touch /etc/udev/rules.d/75-persistent-net-generator.rules //命令中斜体部分会根据用户的实际环境有区别
```

3. 清理网络规则。

- 若操作系统使用initrd系统映像，请执行以下操作：

- i. 执行以下命令，查看initrd开头且default结尾的initrd映像文件，是否存在同时包含persistent和net的网络设备规则文件（以下命令中斜体内容请以实际操作系统版本为准）。

```
lsinitrd /boot/initrd-2.6.32.12-0.7-default |grep persistent|grep net
```

- 否，结束。
- 是，执行下一步。

- ii. 执行以下命令，备份initrd映像文件（以下命令中斜体内容请以实际操作系统版本为准）。

```
cp /boot/initrd-2.6.32.12-0.7-default /boot/initrd-2.6.32.12-0.7-default_bak
```

- iii. 执行以下命令，重新生成initrd映像文件。

```
mkinitrd
```

- 若操作系统使用initramfs系统映像（如Ubuntu），请执行以下操作：

- i. 执行以下命令，查看initrd开头且generic结尾的initramfs映像文件，是否存在同时包含persistent和net的网络设备规则文件。

```
lsinitramfs /boot/initrd.img-3.19.0-25-generic|grep persistent|grep net
```

- 否，无需清理网络规则。
- 是，执行下一步。

- ii. 执行以下命令，备份initrd映像文件。

```
cp /boot/initrd.img-3.19.0-25-generic /boot/initrd.img-3.19.0-25-generic_bak
```

- iii. 执行以下命令，重新生成initramfs映像文件。

```
update-initramfs -u
```

6.3 安装 Cloud-Init 工具

操作场景

为了保证使用私有镜像创建的新云服务器可以通过“用户数据注入”功能注入初始化自定义信息（例如为云服务器设置登录密码），请在创建私有镜像前安装Cloud-Init工具。

- 安装Cloud-Init工具时需要从官网下载并安装，因此，需要提前为云服务器绑定弹性公网IP。
- 不安装Cloud-Init工具，将无法对云服务器进行自定义配置，只能使用镜像原有密码登录云服务器。
- 使用公共镜像创建的云服务器，默认已经安装Cloud-Init，不需要执行安装及配置操作。
- 用户导入镜像创建的云服务器，请按照指导安装及配置Cloud-Init。配置Cloud-Init操作请参考[配置Cloud-Init工具](#)章节。

📖 说明

Cloud-Init为开源软件，如果安装版本有安全漏洞，建议更新至最新版本。

前提条件

- 已为云服务器绑定弹性公网IP。
- 已登录云服务器。
- 服务器的网卡属性为DHCP方式。

安装步骤说明

1. 请先检查是否已安装Cloud-Init工具。
具体操作请参考[检查是否已经安装Cloud-Init工具](#)。
2. 安装Cloud-Init工具。
Cloud-Init安装方式分为：[采用官方提供的包源安装Cloud-Init工具（优先推荐）](#)和[采用官方提供的Cloud-Init源码包通过pip方式安装Cloud-Init工具](#)。

检查是否已经安装 Cloud-Init 工具

请先执行如下步骤检查是否已安装Cloud-Init工具。在不同的操作系统下，查看是否已经安装Cloud-Init工具的方法不同。

- 若依赖python3 环境时，以Ubuntu22.0.4为例，执行以下命令查看是否安装Cloud-Init工具。

which cloud-init

- 回显类似如下，表示已经安装Cloud-Init工具，无需重复安装。

```
/usr/bin/cloud-init
```

- 回显类似如下，表示未安装Cloud-Init工具。

```
/usr/bin/which: no cloud-init in (/usr/local/bin:/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin)
```

- 若依赖python2 环境时，以CentOS 6系列为例，执行以下命令查看是否安装Cloud-Init工具。

which cloud-init

- 回显类似如下，表示已经安装Cloud-Init工具，无需重复安装。

```
cloud-init-0.7.5-10.el6.centos.2.x86_64
```

- 无回显信息表示未安装Cloud-Init工具。

📖 说明

由于Cloud-Init工具存在多种安装方法，若无回显信息，请再次使用`rpm -qa |grep cloud-init`命令进行检查。`which cloud-init`和`rpm -qa |grep cloud-init`任何一个命令显示已安装，表示操作系统已安装Cloud-Init工具。

如果已安装Cloud-Init工具，还需要执行以下操作：

- 请确认当前云服务器操作系统中用于SSH登录的证书是否继续使用。如果不再使用该证书，请删除证书。
 - root用户对应目录下的文件（如“`/$path/$to/$root/.ssh/authorized_keys`”），执行以下命令：

```
cd /root/.ssh
rm authorized_keys
```
 - 非root用户对应目录下的证书文件（如“`/$path/$to/$none-root/.ssh/authorized_keys`”），执行以下命令：

```
cd /home/centos/.ssh
rm authorized_keys
```
- 执行以下命令，清除Cloud-Init工具产生的缓存，确保使用该私有镜像创建的云服务器可以使用证书方式登录。

```
sudo rm -rf /var/lib/cloud/*
```

📖 说明

设置完成后请勿重启云服务器，否则，需重新设置。

采用官方提供的包源安装 Cloud-Init 工具（优先推荐）

在不同操作系统的云服务器上安装Cloud-Init工具的方法不同，请在root用户下执行相关安装操作。

以下将介绍SUSE、CentOS、Fedora、Debian和Ubuntu操作系统安装Cloud-Init工具的具体方法。其他操作系统类型，请安装并配置对应类型的Cloud-Init工具，例如，使用CoreOS操作系统时需安装并配置`coreos-cloudinit`。

- **SUSE操作系统**

SUSE操作系统的Cloud-Init网络安装地址：

<https://ftp5.gwdg.de/pub/opensuse/repositories/Cloud:/Tools/>

<http://download.opensuse.org/repositories/Cloud:/Tools/>

📖 说明

在上述提供的网络安装地址下选择相应操作系统版本的repo安装包进行安装。

以SUSE Enterprise Linux Server 12为例，Cloud-Init工具的安装步骤如下。

- a. 登录到创建Linux私有镜像所使用的云服务器。
- b. 执行以下命令，安装SUSE 12的网络安装源。

```
zypper ar https://ftp5.gwdg.de/pub/opensuse/repositories/Cloud:/Tools/SLE_12_SP3/Cloud:Tools.repo
```

- c. 执行以下命令，更新网络安装源。

```
zypper refresh
```

- d. 执行以下命令，安装Cloud-Init。
zypper install cloud-init
- e. 执行以下命令，设置Cloud-Init为开机自启动服务。
 - SUSE 11:
chkconfig cloud-init-local on; chkconfig cloud-init on; chkconfig cloud-config on; chkconfig cloud-final on
service cloud-init-local status; service cloud-init status; service cloud-config status; service cloud-final status
 - SUSE 12以及openSUSE 12/13/42:
systemctl enable cloud-init-local.service cloud-init.service cloud-config.service cloud-final.service
systemctl status cloud-init-local.service cloud-init.service cloud-config.service cloud-final.service

注意

对于SUSE和openSUSE操作系统，请执行以下步骤禁止动态修改云服务器名称。

1. 执行以下命令，使用vi编辑器打开“dhcp”文件。
vi etc/sysconfig/network/dhcp
2. 将“dhcp”文件中的“DHCLIENT_SET_HOSTNAME”的值修改为“no”。

• **CentOS操作系统**

CentOS操作系统的Cloud-Init网络安装地址如表6-1所示。请在提供的网络安装地址下选择相应的epel-release安装包进行安装。

表 6-1 Cloud-Init 网络安装地址

操作系统类型	版本	网络安装地址
CentOS	6系列32位	https://archives.fedoraproject.org/pub/archive/epel/6/i386/
	6系列64位	https://archives.fedoraproject.org/pub/archive/epel/6/x86_64/
	7系列64位	https://archives.fedoraproject.org/pub/archive/epel/7/x86_64/Packages/e/

- a. 执行以下命令安装Cloud-Init：
yum install ~~网络安装地址~~/epel-release-x-y.noarch.rpm
yum install cloud-init

📖 说明

“网络安装地址”表示Cloud-Init对应的epel-release安装包的地址，“x-y”表示当前操作系统对应的Cloud-Init epel-release版本号，执行命令时需参考表6-1以及实际使用的安装包版本进行替换。

- 以CentOS 6系列64位为例，当前版本号为6.8，则命令修改为：

```
yum install https://archives.fedoraproject.org/pub/archive/epel/6/x86_64/epel-release-6-8.noarch.rpm
```

- 以CentOS 7系列64位为例，当前版本号为7.14。则命令修改为：

```
yum install https://archives.fedoraproject.org/pub/archive/epel/7/x86_64/Packages/e/epel-release-7-14.noarch.rpm
```

- b. 执行以下命令，设置Cloud-Init为开机自启动服务。

```
systemctl enable cloud-init-local.service cloud-init.service cloud-config.service cloud-final.service
```

```
systemctl status cloud-init-local.service cloud-init.service cloud-config.service cloud-final.service
```

- **Fedora操作系统**

安装Cloud-Init前，请确认操作系统已经配置好对应的网络安装源地址，请查看“/etc/yum.repo.d/fedora.repo”文件中是否已配置相关软件包安装源地址，如果没有配置相关地址源，请参考相关Fedora官网信息配置软件包安装源。

- a. 执行以下命令，安装Cloud-Init。

```
yum install cloud-init
```

- b. 执行以下命令，设置Cloud-Init为开机自启动服务。

```
systemctl enable cloud-init-local.service cloud-init.service cloud-config.service cloud-final.service
```

```
systemctl status cloud-init-local.service cloud-init.service cloud-config.service cloud-final.service
```

- **Debian和Ubuntu操作系统**

安装Cloud-Init前，请确认操作系统已经配置好对应的网络安装源地址，请查看“/etc/apt/sources.list”文件中是否已配置相关软件包安装源地址，如果没有配置相关地址源，请参考Debian或者Ubuntu官网信息配置软件包安装源。

- a. 执行以下命令，安装Cloud-Init。

```
apt-get update
```

```
apt-get install cloud-init
```

- b. 执行以下命令，设置Cloud-Init为开机自启动服务。

```
systemctl enable cloud-init-local.service cloud-init.service cloud-config.service cloud-final.service
```

```
systemctl status cloud-init-local.service cloud-init.service cloud-config.service cloud-final.service
```

采用官方提供的 Cloud-Init 源码包通过 pip 方式安装 Cloud-Init 工具

以cloud-init-0.7.9版本为例，Cloud-Init工具的安装步骤如下。

1. 下载cloud-init-0.7.9.tar.gz源码包（推荐优先选用0.7.9版本），上传到云服务器指定目录“/home/”下。

cloud-init-0.7.9.tar.gz源码包下载地址：

<https://launchpad.net/cloud-init/trunk/0.7.9/+download/cloud-init-0.7.9.tar.gz>

- 在“~/pip/”目录下新建pip.conf文件，编辑内容如下。

📖 说明

“~/pip/”若不存在，可使用命令`mkdir ~/pip`命令新建。

```
[global]
index-url = https://<$mirror>/simple/
trusted-host = <$mirror>
```

📖 说明

编辑内容中<\$mirror>部分可以选择公网PyPI源进行替换。

公网PyPI源：<https://pypi.python.org/>

📖 说明

编辑内容中<\$mirror>部分可以选择公网PyPI源或教育网PyPI源进行替换。

- 公网PyPI源：<https://pypi.python.org/>
- 教育网PyPI源：<https://pypi.tuna.tsinghua.edu.cn/>
<https://pypi.mirrors.ustc.edu.cn/>

- 执行以下命令，安装本地下载的Cloud-Init源码包，安装过程中根据需要选择**--upgrade**参数。

```
pip install [--upgrade] /home/cloud-init-0.7.9.tar.gz
```

📖 说明

安装Cloud-Init源码包的更多信息，请参见[cloud-init官方文档](#)。

- 执行命令**cloud-init -v**，如回显如下类似信息表示安装Cloud-Init成功。

```
cloud-init 0.7.9
```

- 设置Cloud-Init相关服务为开机自启动。

- 若操作系统是sysvinit自启动管理服务，则执行以下命令进行设置。

```
chkconfig --add cloud-init-local; chkconfig --add cloud-init; chkconfig --add cloud-config; chkconfig --add cloud-final
```

```
chkconfig cloud-init-local on; chkconfig cloud-init on; chkconfig cloud-config on; chkconfig cloud-final on
```

```
service cloud-init-local status; service cloud-init status; service cloud-config status; service cloud-final status
```

- 若操作系统是systemd自启动管理服务，则执行以下命令进行设置。

```
systemctl enable cloud-init-local.service cloud-init.service cloud-config.service cloud-final.service
```

```
systemctl status cloud-init-local.service cloud-init.service cloud-config.service cloud-final.service
```

⚠ 注意

采用官方提供的Cloud-Init源码包通过pip方式进行安装时要注意以下两点。

1. Cloud-Init安装时需要添加syslog用户到adm组。存在syslog用户时直接添加syslog用户到adm组。不存在syslog用户时（如CentOS和SUSE），执行下列命令创建syslog用户，添加到adm组：

```
useradd syslog
groupadd adm
usermod -g adm syslog
```

2. 在“/etc/cloud/cloud.cfg”中system_info部分的distro要根据具体操作系统发行版本做相应修改（如根据具体操作系统发行版相应修改为：**distro: ubuntu**，**distro: sles**，**distro: debian**，**distro: fedora**）。

6.4 配置 Cloud-Init 工具

操作场景

Cloud-Init工具安装完成后，请参考本节操作配置Cloud-Init工具。

前提条件

- 已安装Cloud-Init工具。
- 已为云服务器绑定弹性公网IP。
- 已登录云服务器。
- 服务器的网卡属性为DHCP方式。

配置步骤说明

包含如下两步操作：

1. 配置Cloud-Init工具。
具体操作请参考[配置Cloud-Init工具](#)。
2. 检查Cloud-Init工具相关配置是否成功。
具体操作请参考[检查Cloud-Init工具相关配置是否成功](#)。

配置 Cloud-Init 工具

1. 用户可以根据需要配置登录云服务器的用户权限。使用root账户登录，需要开启root用户的ssh权限，并开启密码远程登录。
 - 若用户选择注入密码，则通过自己注入的密码进行远程SSH或noVNC登录。
 - 若用户选择注入密钥，则通过自己注入的密钥进行远程SSH登录。
2. 设置开放root密码远程登录并开启root用户的ssh权限。
以CentOS 6.7系列操作系统为例，在“/etc/cloud/cloud.cfg”设置“disable_root”值为“0”表示不禁用（部分OS的Cloud-Init配置使用“true”表示禁用，“false”表示不禁用），设置“ssh_pwauth”为“1”启用密码远程登录，“lock_passwd”设置为“False”表示不锁住用户密码。

```
users:
- name: root
  lock_passwd: False

disable_root: 0
ssh_pwauth: 1
```

3. 在配置文件“/etc/cloud/cloud.cfg”中禁用Cloud-Init接管网络。
当Cloud-Init版本等于或高于0.7.9版本时，在配置文件“/etc/cloud/cloud.cfg”中增加如下内容，禁用Cloud-Init接管网络。

图 6-1 禁用 Cloud-Init 接管网络

```
users:
- name: root
  lock_passwd: False

disable_root: 0
ssh_pwauth: 1

datasource_list: ['OpenStack']

network:
  config: disabled
```

说明

增加的内容需严格按照yaml格式进行配置。

4. 配置agent访问OpenStack数据源。
在“/etc/cloud/cloud.cfg”最后一行添加如下内容，配置agent访问OpenStack数据源。

```
datasource_list: [ OpenStack ]
datasource:
  OpenStack:
    metadata_urls: ['http://169.254.169.254']
    max_wait: 120
    timeout: 5
    apply_network_config: false
```

说明

- max_wait和timeout可由用户自定义是否需要配置，上述回显信息中max_wait和timeout的取值仅供参考。
 - 当操作系统版本低于Debian8、CentOS 5时，不支持配置agent访问OpenStack数据源。
 - CentOS、EulerOS操作系统云服务器必须要禁用默认的zeroconf路由，以便精确访问OpenStack数据源。
echo "NOZEROCONF=yes" >> /etc/sysconfig/network
 - apply_network_config: false为可选项，对于使用Cloud-Init 18.3及以上版本的用户，需添加此配置项。
5. 在配置文件“/etc/cloud/cloud.cfg”中补充如下内容。

manage_etc_hosts: localhost

防止启动云服务器时，系统长时间卡在“Waiting for cloudResetPwdAgent”状态。

图 6-2 新增 manage_etc_hosts: localhost

```
datasource_list: ['OpenStack']
manage_etc_hosts: localhost

datasource:
  OpenStack:
    # timeout: the timeout value for a request at metadata service
    timeout : 50
    # The length in seconds to wait before giving up on the metadata
    # service. The actual total wait could be up to
    # len(resolvable_metadata_urls)*timeout
    max_wait : 120
```

6. 执行 `vi /etc/ssh/sshd_config` 命令，在 vi 编辑器中打开 “/etc/ssh/sshd_config”。
- 将 “sshd_config” 中的 “PasswordAuthentication” 的值修改为 “yes”。

说明

如果是 SUSE 和 openSUSE 操作系统，需要同时配置 “sshd_config” 文件中的以下两个参数为 “yes”。

- PasswordAuthentication
 - ChallengeResponseAuthentication
7. 修改 cloud.cfg 配置文件里面的 cloud_init_modules 配置。
 - 将 ssh 从最后提前到第一位处理，提高云服务器 ssh 登录速度。
 - 开启机器名更新机制，请勿注释或删除 “ - update_hostname ” 语句。

cloud_init_modules:

```
- ssh
- migrator
- bootcmd
- write-files
- growpart
- resizefs
- set_hostname
- update_hostname
- update_etc_hosts
- rsyslog
- users-groups
```

8. 修改以下配置使得镜像创建的云服务器主机名不带 “.novalocal” 后缀且主机名称中可以带点号。

- a. 执行如下命令，修改 “__init__.py” 文件。

```
vi /usr/lib/python*/site-packages/cloudinit/sources/__init__.py
```

不同 OS 安装的 python 版本有差异，请选择对应的路径。

按 “i” 进入编辑模式，根据关键字 toks 查询，修改内容如下信息所示。

```
if toks:
    toks = str(toks).split('.')
else:
    #toks = ["ip-%s" % lhost.replace(".", "-")] #屏蔽此行
    toks = lhost.split(".novalocal") #增加此行

if len(toks) > 1:
    hostname = toks[0]
    #domain = ''.join(toks[1:]) #屏蔽此行
else:
    hostname = toks[0]

if fqdn and domain != defdomain:
    #return hostname #屏蔽此行
```

```
return "%s.%s" % (hostname, domain) #增加此行
else:
    return hostname
```

修改完成后按“ESC”退出编辑模式，输入:**wq!**保存并退出。

- b. 执行如下命令进入“cloudinit/sources”文件夹。

```
cd /usr/lib/python*/site-packages/cloudinit/sources/
```

不同OS安装的python版本有差异，请选择对应的路径。

- c. 执行如下命令，删除“__init__.pyc”文件和优化编译后的“__init__.pyo”文件。

```
rm -rf __init__.pyc
```

```
rm -rf __init__.pyo
```

- d. 执行如下命令，清理日志信息。

```
rm -rf /var/lib/cloud/*
```

```
rm -rf /var/log/cloud-init*
```

9. 执行以下命令编辑Cloud-Init日志输出路径配置文件，设置日志处理方式handlers，建议配置为cloudLogHandler。

```
vim /etc/cloud/cloud.cfg.d/05_logging.cfg
```

```
[logger_cloudinit]
level=DEBUG
qualname=cloudinit
handlers=cloudLogHandler
propagate=1
```

10. 确保删除镜像模板中已经存在的linux账户和“/home/linux”目录。

```
userdel linux
```

```
rm -fr /home/linux
```

11. 确保Cloud-Init启用。

若存在/etc/cloud/cloud-init.disabled文件，则执行**cloud-init clean --machine-id**，否则跳过。

检查 Cloud-Init 工具相关配置是否成功

执行以下命令，无错误发生，说明Cloud-Init配置成功。

```
cloud-init init --local
```

正确安装的Cloud-Init会显示Cloud-Init的版本详细信息，并且无任何错误信息。例如，正确安装的情况下，不含有缺少文件的提示信息。

📖 说明

执行如下命令，可将系统用户密码有效期设置为最大。此操作可选。

```
chage -M 99999 $user_name
```

其中，user_name为系统用户，例如root账户。

密码有效期建议设置为99999。

6.5 安装一键式重置密码插件（Linux）

操作场景

为了保证使用私有镜像创建的新云服务器可以实现一键式重置密码功能，建议您在创建私有镜像前安装密码重置插件CloudResetPwdAgent。

- 使用公共镜像创建的云服务器，默认已经安装密码重置插件，不需要执行安装操作。
- 使用外部镜像文件创建的云服务器，请按照指导安装密码重置插件。

说明

可选操作。如需使用控制台提供的一键式重置密码功能，请按照本节指导安装插件。

安装须知

- 用户自行决定是否安装CloudResetPwdAgent插件，使云服务器具备一键式重置密码功能。
- 支持安装一键式重置密码插件的操作系统版本如表6-2所示。

表 6-2 支持安装一键式重置密码插件的操作系统版本

操作系统类型	操作系统版本
CentOS	CentOS 8.3 64bit
	CentOS 8.2 64bit
	CentOS 8.1 64bit
	CentOS 8.0 64bit
	CentOS 7.3 64bit
	CentOS 7.2 64bit
	CentOS 7.0 64bit
	CentOS 7.1 64bit
	CentOS 6.9 64bit
	CentOS 6.8 64bit
	CentOS 6.8 32bit
	CentOS 6.6 32bit
	CentOS 6.6 64bit
	CentOS 6.5 64bit
	CentOS 6.4 64bit
	CentOS 6.3 64bit
	说明 CentOS 8需要先关闭selinux后再安装一键式重置密码插件，安装后再根据需要选择是否打开selinux。

操作系统类型	操作系统版本
Debian	Debian 9.0 64bit Debian 8.8 64bit Debian 8.2 64bit Debian 7.5 64bit Debian 7.5 32bit
openSUSE	openSUSE 42.2 64bit openSUSE 13.2 64bit openSUSE Leap 42.2 64bit openSUSE Leap 42.1 64bit
SUSE	SUSE 12 SP2 64bit SUSE 12 SP1 64bit SUSE 11 SP4 64bit
Ubuntu	Ubuntu 16.10 32bit Ubuntu 16.04 32bit Ubuntu Server 16.04 64bit Ubuntu Server 14.04 64bit Ubuntu Server 14.04 32bit
EulerOS	EulerOS 2.2 64bit
Fedora	Fedora 25 64bit Fedora 24 64bit
Oracle Linux	Oracle Linux 7.3 64bit Oracle Linux 6.9 64bit Oracle Linux 6.5 64bit

前提条件

- 云服务器的状态为“运行中”。
- 需保证根目录可写入，且剩余空间大于300MB。
- 对于Linux弹性云服务器，selinux会影响重置密码插件运行。若开启了selinux，请参考《弹性云服务器用户指南》的“关闭selinux”章节禁用selinux。
- 使用SUSE 11 SP4镜像创建的云服务器，内存需要大于等于4G时才能支持一键式重置密码功能。
- 云服务器使用的VPC网络DHCP不能禁用。
- 云服务器网络正常通行。
- 云服务器安全组出方向规则满足如下要求：
 - 协议：TCP
 - 端口范围：80

- 远端地址: 169.254.0.0/16

如果您使用的是默认安全组出方向规则，则已经包括了如上要求，可以正常初始化。默认安全组出方向规则为：

- 协议: ALL
- 端口范围: ALL
- 远端地址: 0.0.0.0/16

操作步骤

步骤1 检查弹性云服务器是否已安装一键式重置密码插件。

1. 以root用户登录弹性云服务器。
2. 执行以下命令，查询是否已安装CloudResetPwdAgent。

```
ls -lh /Cloud*
```

图 6-3 查询是否已安装一键式重置密码插件

```
[root@ecs-test ~]# ls -lh /Cloud*
total 20K
drwx----- 2 root root 4.0K Jun 13 14:13 bin
drwxr-xr-x 2 root root 4.0K Jun 13 11:53 conf
drwx----- 3 root root 4.0K Jun 13 11:53 depend
drwx----- 2 root root 4.0K Jun 13 11:53 lib
drwx----- 2 root root 4.0K Jun 13 14:13 logs
[root@ecs-test ~]#
[root@ecs-test ~]#
```

检查结果是否如图6-3所示。

- 是，表示已安装一键式重置密码插件，结束。
- 否，表示未安装一键式重置密码插件，请继续执行如下操作进行安装。

步骤2 下载一键式重置密码插件CloudResetPwdAgent.zip。

该插件放置在OBS桶中，请联系管理员获取OBS桶的路径。

假设获取的下载地址为：http://xxx-cloud-reset-pwd.obs.com/linux/64/reset_pwd_agent/CloudResetPwdAgent.zip

则下载插件的命令如下：

```
wget http://xxx-cloud-reset-pwd.obs.com/linux/64/reset_pwd_agent/CloudResetPwdAgent.zip
```

步骤3 执行以下命令，解压软件包CloudResetPwdAgent.zip。

安装一键式重置密码插件对插件的解压目录无特殊要求，请您自定义。

```
unzip -o -d 插件解压目录 CloudResetPwdAgent.zip
```

示例：

假设插件解压的目录为/home/linux/test，则命令行如下：

```
unzip -o -d /home/linux/test CloudResetPwdAgent.zip
```

步骤4 安装一键式重置密码插件。

1. 执行以下命令，进入文件CloudResetPwdAgent.Linux。
cd CloudResetPwdAgent/CloudResetPwdAgent.Linux
2. 执行以下命令，添加文件setup.sh的运行权限。
chmod +x setup.sh
3. 执行以下命令，安装插件。
sudo sh setup.sh

如果脚本执行成功打印“cloudResetPwdAgent install successfully.”，且未打印“Failed to start service cloudResetPwdAgent”，表示安装成功。

说明

- 您也可以根据**步骤1**，检查密码重置插件是否安装成功。
- 如果密码重置插件安装失败，请检查安装环境是否符合要求，并重试安装操作。

步骤5 修改重置密码插件的文件权限。

```
chmod 700 /CloudrResetPwdAgent/bin/cloudResetPwdAgent.script
chmod 700 /CloudrResetPwdAgent/bin/wrapper
chmod 600 /CloudrResetPwdAgent/lib/*
```

----结束

6.6 安装 Linux 特殊驱动

操作场景

对于一些类型的云服务器，如果使用私有镜像进行创建，需要在制作私有镜像时安装特殊驱动。

InfiniBand 网卡驱动

1. 如果这个私有镜像用于创建H2型云服务器，需要在镜像中安装InfiniBand网卡驱动。请根据IB类型，在Mellanox官网选择相应版本的InfiniBand网卡驱动下载（建议选择4.2-1.0.0.0的驱动版本下载使用），并根据Mellanox提供的操作指导进行安装。
 - IB网卡类型：“Mellanox Technologies ConnectX-4 Infiniband HBA (MCX455A-ECAT)”
 - Mellanox官网地址：<http://www.mellanox.com/>
 - 网卡驱动下载地址：https://network.nvidia.com/products/infiniband-drivers/linux/mlnx_ofed/
2. 如果这个私有镜像用于创建HL1型云服务器，需要在镜像中安装InfiniBand网卡驱动。请根据IB类型，在Mellanox官网选择相应版本的InfiniBand网卡驱动下载（建议选择4.2-1.0.0.0的驱动版本下载使用），并根据Mellanox提供的操作指导进行安装。
 - IB网卡类型：“Mellanox Technologies ConnectX-4 Infiniband HBA (MCX455A-ECAT)”
 - Mellanox官网地址：<http://www.mellanox.com/>

6.7 卸载云服务器的数据盘

操作场景

如果创建私有镜像所使用的原云服务器挂载了多个数据盘，可能导致由私有镜像创建的新云服务器无法使用。因此在创建私有镜像前，需要卸载原云服务器中挂载的所有数据盘。

本节介绍如何卸载云服务器中挂载的所有数据盘。

前提条件

已登录到创建Linux私有镜像所使用的云服务器。

操作步骤

1. 查询是否挂载数据盘。

执行以下命令，检查云服务器挂载的磁盘块数。

fdisk -l

- 磁盘块数 > 1，表示云服务器挂载有数据盘，执行2。
- 磁盘块数=1，表示云服务器未挂载数据盘，执行3。

2. 执行以下命令，检查当前云服务器挂载的数据分区。

mount

- 回显不包含数据盘分区的信息时，无需进行卸载数据盘的操作。
`/dev/vda1 on / type ext4 (rw,relatime,data=ordered)`
- 回显包含类似如下信息，请执行3。
`/dev/vda1 on / type ext4 (rw,relatime,data=ordered)`
`/dev/vdb1 on /mnt/test type ext4 (rw,relatime,data=ordered)`

3. 清理“fstab”文件中的配置信息。

- a. 执行以下命令编辑“fstab”文件。

vi /etc/fstab

- b. 删除相关数据盘在“fstab”文件中的配置信息。

“/etc/fstab”文件中记录了系统启动时自动挂载的文件系统和存储设备的信息，需要删除，如图6-4中最后一行是数据盘在“fstab”中的配置信息。

图 6-4 数据盘在 fstab 文件中的配置信息

```
[root@ecs-bf78 ~]# cat /etc/fstab
#
# /etc/fstab
# Created by anaconda on Wed Feb 27 06:58:16 2019
#
# Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
#
UUID=4c2c090d-4228-49fc-9cbe-3920b3bf287c / ext4 defaults 1 1
UUID=9c29104b-31b8-4421-a207-102f86ec7ae5 /mnt/test ext4 defaults 1 1
```

4. 执行如下命令，卸载已挂载的数据盘。

卸载前请确认数据盘上没有任何数据读写，否则会导致卸载失败。

umount /dev/vdb1

5. 执行以下命令，再次查看当前云服务器挂载的数据分区。

mount

回显信息中不再包含数据盘分区的信息时，则表示卸载成功。

7 标签管理

操作场景

用户可以使用“镜像标签”对镜像进行分类。您可以增加、修改或删除镜像标签，也可以在镜像列表中按标签搜索需要的镜像。

说明

- 镜像添加标签和使用标签搜索镜像时，查询预定义标签需要申请标签管理服务（Tag Management Service, TMS）的访问权限。

约束与限制


一个镜像最多添加10个标签。

镜像标签的增、删、改

1. 登录IMS控制台。
 - a. 登录管理控制台。
 - b. 选择“计算 > 镜像服务”。进入镜像服务页面。
2. 选择“私有镜像”页签，单击镜像名称查看镜像详情。
 - 修改镜像标签，请执行3。
 - 删除镜像标签，请执行4。
 - 增加镜像标签，请执行5。
3. 在镜像详情里，单击“标签”页签，在“镜像标签”列表选择一个标签，单击“编辑”，对镜像标签进行修改。
4. 在镜像详情里，单击“标签”页签，在“镜像标签”列表选择一个标签，单击“删除”，单击“确定”，完成对镜像标签的删除。
5. 在镜像详情里，单击“标签”页签，单击“添加标签”，在“添加标签”对话框中增加新的镜像标签。

按标签搜索私有镜像

1. 登录IMS控制台。

- a. 登录管理控制台。
- b. 选择“计算 > 镜像服务”。
进入镜像服务页面。
2. 选择“私有镜像”页签，单击右上角的“标签搜索”，展开查询页。
3. 输入待查询私有镜像的标签。
标签键和标签值均不能为空，当标签键和标签值全匹配时，系统可以自动查询到目标私有镜像。
4. 单击  添加标签。
系统支持不同标签的组合搜索，并取各个标签的交集。
5. 单击“搜索”。
系统根据标签键或标签值搜索目标私有镜像。


8 配额管理

什么是配额？

为防止资源滥用，平台限制了各服务资源的配额，对用户的资源数量和容量做了限制。如您最多可以创建多少台弹性云服务器、多少块云硬盘。

如果当前资源配额限制无法满足使用需要，您可以申请扩大配额。

怎样查看我的配额？

1. 登录管理控制台。
 2. 单击页面右上角的“**My Quota**”图标  。
 3. 您可以在“服务配额”页面，查看各项资源的总配额及使用情况。
- 如果当前配额不能满足业务要求，请参考后续操作，申请扩大配额。

如何申请扩大配额？

目前系统暂不支持在线调整配额大小。如您需要调整配额，请联系运营管理员。

在联系运营管理员之前，请您准备好以下信息：

- 账号名，获取方式如下：
登录云账户管理控制台，在右上角单击用户名，选择“我的凭证”，在“我的凭证”页面获取“账号名”。
- 配额信息，包括：服务名、配额类别、需要的配额值。

9 使用 CTS 审计 IMS 操作事件

9.1 IMS 支持审计的关键操作列表

操作场景

云审计服务（Cloud Trace Service, CTS），是云平台安全解决方案中专业的日志审计服务，提供对各种云资源操作记录的收集、存储和查询功能，可用于支撑安全分析、合规审计、资源跟踪和问题定位等常见应用场景。

通过云审计服务，您可以记录与镜像服务相关的操作事件，便于日后的查询、审计和回溯。

前提条件

使用云审计服务前需要先开通云审计服务，如果不开通云审计服务，则无法对资源操作进行记录。开通后CTS会自动创建一个追踪器，并将当前租户的所有操作记录在该追踪器中。CTS最多显示近7天的事件，为了长期保存操作记录，可以将事件文件保存至对象存储服务中。

支持审计的关键操作列表

表 9-1 云审计服务支持的 IMS 操作列表

操作名称	资源类型	事件名称
创建镜像	ims	createImage
修改镜像	ims	updateImage
批量删除镜像	ims	deleteImage
复制镜像	ims	copyImage
导出镜像	ims	exportImage
新增成员	ims	addMember

操作名称	资源类型	事件名称
批量修改成员	ims	updateMember
批量删除成员	ims	deleteMemeber

表 9-2 由 IMS 触发的操作与 OpenStack 原生接口的关系

操作名称	事件名称	服务类型	资源类型	归属
创建镜像	createImage	IMS	image	glance
修改镜像信息/ 上传镜像	updateImage	IMS	image	glance
删除镜像	deleteImage	IMS	image	glance
添加标签	addTag	IMS	image	glance
删除标签	deleteTag	IMS	image	glance
添加镜像成员	addMember	IMS	image	glance
修改镜像成员 信息	updateMember	IMS	image	glance
删除镜像成员	deleteMember	IMS	image	glance

9.2 查看 IMS 审计日志


操作场景



用户进入云审计服务创建管理类追踪器后，系统开始记录云服务资源的操作。云审计服务管理控制台会保存最近的操作记录。

说明

云审计控制台对用户的操作事件日志保留，过期自动删除，不支持人工删除。

查看审计事件

1. 登录管理控制台。
2. 单击左上角 ，选择“> 云审计服务 CTS”，进入云审计服务页面。
3. 单击左侧导航树的“事件列表”，进入事件列表信息页面。
4. 事件列表支持通过筛选来查询对应的操作事件。当前事件列表支持四个维度的组合查询，详细信息如下：
 - 事件类型、事件来源、资源类型和筛选类型，在下拉框中选择查询条件。

- 筛选类型按资源ID筛选时，还需手动输入某个具体的资源ID。
 - 筛选类型按事件名称筛选时，还需选择某个具体的事件名称。
 - 筛选类型按资源名称筛选时，还需选择或手动输入某个具体的资源名称。
 - 操作用户：在下拉框中选择某一具体的操作用户，此操作用户指用户级别，而非租户级别。
 - 事件级别：可选项为“所有事件级别”、“Normal”、“Warning”、“Incident”，只可选择其中一项。
 - 时间范围：可选择查询最近1小时、最近1天、最近1周的操作事件，也可以自定义最近内任意时间段的操作事件。
5. 选择完查询条件后，单击“查询”。
 6. 在事件列表页面，您还可以导出操作记录文件和刷新列表。
 - 单击“导出”按钮，云审计服务会将查询结果以CSV格式的表格文件导出，该CSV文件包含了本次查询结果的所有事件，且最多导出5000条信息。
 - 单击按钮，可以获取到事件操作记录的最新信息。
 7. 在需要查看的事件左侧，单击展开该记录的详细信息。
 8. 在需要查看的记录右侧，单击“查看事件”，会弹出一个窗口显示该操作事件结构的详细信息。

查看事件 ×

```

{
  "request": "",
  "trace_id": "676d4ae3-842b-11ee-9299-9159eee6a3ac",
  "code": "200",
  "trace_name": "createDockerConfig",
  "resource_type": "dockerlogincmd",
  "trace_rating": "normal",
  "api_version": "",
  "message": "createDockerConfig, Method: POST Url=/v2/manage/utils/secret. Reason:",
  "source_ip": " ",
  "domain_id": " ",
  "trace_type": "ApiCall",
  "service_type": "SWR",
  "event_type": "system",
  "project_id": " ",
  "response": "",
  "resource_id": "",
  "tracker_name": "system",
  "time": "2023/11/16 10:54:04 GMT+08:00",
  "resource_name": "dockerlogincmd",
  "user": {
    "domain": {
      "name": " ",
      "id": " "
    }
  }
}
            
```

9. 关于事件结构的关键字段详解，请参见《云审计服务用户指南》中的和。

10 常见问题

10.1 镜像咨询类

10.1.1 镜像怎么选？

在创建弹性云服务器实例时，必须选择一个镜像，如何在众多镜像类型和操作系统中选择合适的镜像，取决于以下因素：

- [镜像类型](#)
- [操作系统](#)

镜像类型

根据镜像来源不同，分为公共镜像、私有镜像和共享镜像。其中，私有镜像又分为系统盘镜像、数据盘镜像和整机镜像。详情请参见[什么是镜像服务](#)。

操作系统

选择操作系统时，您可能需要考虑以下方面：

- 选择架构类型

系统架构	适用内存	使用限制
32位	适用于4 GB以下内存	<ul style="list-style-type: none">• 实例规格内存大于4 GB时，无法使用32位操作系统。• 32位操作系统只能4 GB内寻址，超过4 GB的内存操作系统无法访问。
64位	适用于4 GB及以上内存	如果您的应用需要4 GB以上内存或者未来需要扩充到4 GB以上，请使用64位操作系统。

- 选择操作系统类型

操作系统类型	适用场景	使用限制
Windows	<ul style="list-style-type: none"> • 适合运行Windows下开发的程序，如.NET等。 • 支持SQL Server等数据库（需要自行安装）。 	系统盘要求40 GB及以上、内存1 GB及以上。
Linux	<ul style="list-style-type: none"> • 适合运行高性能Web等服务器应用，支持常见的PHP、Python等编程语言。 • 支持MySQL等数据库（需要自行安装）。 	系统盘要求40 GB及以上、内存512 MB及以上。

10.1.2 镜像和备份有什么区别？

云备份和镜像服务有很多功能交融的地方，有时需要搭配一起使用。镜像有时也可用来备份云服务器运行环境，作为备份来使用。

备份和镜像的区别

云备份和镜像服务区别主要有以下几点，如表10-1所示。

表 10-1 备份和镜像的区别

对比维度	云备份	镜像服务
概念	备份是将云服务器或者云硬盘某一时间节点的状态、配置和数据信息保存下来，以供故障时进行恢复，其目的是为了保证数据安全，提升高可用性。	镜像相当于云服务器的“装机盘”，它提供了启动云服务器所需的所有信息，其目的是为了创建云服务器，批量部署软件环境。系统盘镜像包含运行业务所需的操作系统、应用软件，数据盘包含业务数据。整机镜像是系统盘镜像和数据盘镜像的总和。
使用方式	<ul style="list-style-type: none"> • 数据存储位置：与服务器/磁盘数据分开存储，存储在对象存储（OBS）中。如果将创建备份的云硬盘删除，对应的备份不会被同时删除。 • 操作对象：保存云服务器/磁盘指定时刻的数据，可以设置自动备份和过期自动删除。 • 用途：备份可以恢复数据至原服务器/磁盘中，也可以直接创建新的磁盘或整机镜像。 • 是否可以导出至本地：否。 	<ul style="list-style-type: none"> • 数据存储位置：与服务器/磁盘数据分开存储，存储在对象存储（OBS）中。如果将创建镜像的服务器/磁盘删除，对应的镜像不会被同时删除。 • 操作对象：可以将服务器的系统盘和数据盘制作为私有镜像，也可以通过外部镜像文件制作私有镜像。 • 用途：系统盘镜像或整机镜像可以创建新的服务器，数据盘镜像可以创建新的磁盘，实现业务迁移。 • 是否可以导出至本地：是。整机镜像不支持导出至本地。

对比维度	云备份	镜像服务
应用场景	适用场景： <ul style="list-style-type: none"> • 数据备份与恢复 • 业务快速部署和迁移 	适用场景： <ul style="list-style-type: none"> • 服务器上云或云上迁移 • 部署特定软件环境 • 批量部署软件环境 • 服务器运行环境备份
优势	支持自动备份，可以定时定量保留服务器/磁盘某一时间节点的数据。支持备份云下VMware虚拟机，并同步至云上，在云上使用备份恢复数据至新的云服务器。	可以备份系统盘。可以将本地或者其他云平台的服务器数据盘镜像文件导入至镜像服务中。导入后，可使用该镜像创建新的云硬盘。

📖 说明

备份和镜像虽然存储在对象存储服务中，但不占用用户的资源，所以用户无法从对象存储服务中看到备份和镜像数据。

备份和镜像的联系

1. 通过云服务器备份可以创建整机镜像。
2. 为云服务器创建整机镜像时，需要先对目标云服务器进行备份。
3. 使用备份创建镜像时，镜像会对备份进行压缩，所以产生的镜像可能会比备份小。

10.1.3 可以裁剪镜像吗？

不建议您裁剪镜像。

在导入外部镜像文件时，建议您使用操作系统官方发行版本进行系统构建，避免对发行版本进行裁剪及高度定制，以减少问题发生概率。

操作系统发行版本由操作系统厂商进行不定期版本更迭，部分系统版本厂商已停止维护，不再发布问题修复及安全补丁，建议及时关注厂商公告进行系统升级更新，保证您系统的健壮性。

10.1.4 如何备份云服务器当前状态，方便以后系统故障时进行恢复？

可以考虑如下方案：

- （推荐）使用云备份为云服务器创建定时备份。当云服务器故障时，选择需要恢复的时间点所对应的备份副本，将其制作为整机镜像，然后使用该镜像申请新的云服务器，或者在已有云服务器基础上切换镜像。
- 为云服务器制作系统盘镜像。当云服务器故障时，使用该系统盘镜像申请新的云服务器，或者在已有云服务器基础上切换镜像。
- 为云服务器的系统盘创建快照。当云服务器故障时，回滚快照即可。

10.1.5 创建的私有镜像如何使用到已有的云服务器上？

- 后续可以为云服务器切换操作系统，选择私有镜像即可。切换操作系统的详细指导请参见《弹性云服务器用户指南》“切换操作系统”章节。

10.1.6 数据盘镜像中的数据可以导入到之前已有的数据盘内吗？

不能直接导入。

数据盘镜像只能用于申请新的磁盘，无法直接导入到之前已有的磁盘内。您可以按如下指导操作：

1. 使用数据盘镜像临时生成一个新磁盘。
2. 将新磁盘挂载至旧磁盘所在的云服务器。
3. 拷贝数据到旧磁盘内，再删除这个临时的磁盘。

10.1.7 能否跨账号使用私有镜像？

能。

您可以通过共享镜像功能，将私有镜像共享给其他账号使用。

详细操作请参见[共享指定镜像](#)。

10.2 镜像创建类

10.2.1 创建镜像 FAQ

如何通过云服务器创建一台相同配置的云服务器？

假设您的云服务器已经部署好应用，您可以将这台云服务器创建为私有镜像，然后通过私有镜像创建相同配置的云服务器，省去重复部署应用的时间。参考如下指导：

- [通过云服务器创建Windows系统盘镜像](#)
- [通过云服务器创建Linux系统盘镜像](#)
- [通过镜像创建云服务器](#)

一个账号最多可以创建多少个私有镜像？

在当前阶段，您在一个区域内默认最多可以创建100个私有镜像。

云服务器创建私有镜像的时候一定要关机吗？

不需要，当前支持云服务器开机制作镜像。但是在镜像制作过程中，如果原主机有数据写入，这些数据将不包含在镜像中。

在哪里查看镜像创建进度？创建镜像需要多少时间？

登录管理控制台，选择“计算 > 镜像服务”，单击“私有镜像”页签，在私有镜像列表的“状态”列查看镜像创建进度。

镜像创建过程涉及XEN、KVM虚拟化平台原生驱动的安装，OS内核的加载以及grub引导配置等环节，处理时间可能会比较长。并且网络速度、镜像文件类型、镜像所属实例的磁盘大小等因素均会影响镜像的创建时长。

在子账户做的私有镜像，主账户创建云服务器的时候可以选择到吗？

可以。

子账户做的私有镜像，主账户和其他子账户（如果有）均可以看到。

- 如果该私有镜像为系统盘镜像或整机镜像，主账户和其他子账户创建云服务器时，选择“私有镜像”，然后在下拉列表中选择该镜像。
- 如果该私有镜像为数据盘镜像，主账户和其他子账户创建云硬盘时，选择数据源为“从镜像创建”，然后在弹出的对话框中选择该镜像。

另外，主账户创建的私有镜像，所有子账户都可以看到。

10.2.2 整机镜像 FAQ

什么是整机镜像？

整机镜像是包含云服务器操作系统、应用软件和业务数据的镜像。一般适用于云服务器数据整体搬迁，例如：

- 将老旧云服务器上的数据迁移至新云服务器

为什么创建整机镜像时需要选择存储库？需要多支付费用吗？

使用云备份创建整机镜像时，必须选择一个存储库，这个存储库相当于存储容器，镜像、备份都存放在该容器中。用户需要为存储库付费。

使用云服务器备份创建整机镜像时，占用CSBS的存储空间，存储的容器未对用户开放。

因此不管哪种创建方式，都要收取一定的存储费用，选择存储库并不代表需要多付费。

创建成功的整机镜像在哪里查看数据盘信息？

整机镜像创建成功后，镜像列表和详情中只显示系统盘信息（即“磁盘容量”参数），数据盘信息可以在云服务器备份或云备份中查看。究竟在云服务器备份中查看，还是在云备份中查看，取决于整机镜像的创建方式。

以在云备份中查看为例，方法如下：

1. 在私有镜像列表中，单击整机镜像名称。
进入镜像详情页面。
2. 找到“来源”参数，单击其后的备份ID。
进入云备份详情页面。
3. 单击“磁盘级备份”页签，列表中展示了系统盘和数据盘的详细信息。

整机镜像有哪些使用限制？

- 整机镜像不支持导出，建议将系统盘和数据盘分别制作镜像，然后进行导出。

- 整机镜像不支持区域内复制。

10.2.3 使用备份创建镜像和使用云服务器创建镜像是否有区别？

没有区别。

创建整机镜像的方式：使用云服务器创建、以及使用云备份创建。

使用备份创建镜像与使用云服务器创建镜像原理一样。云服务器创建镜像时，先为云服务器创建备份，再通过备份创建镜像，中间过程为系统自动完成的。所以二者没有区别。

10.2.4 为什么创建云服务器或者为云服务器切换操作系统时选不到ISO 镜像？

- 将ISO文件注册为ISO镜像后，无法在创建云服务器页面选择到，也无法用来切换操作系统，必须完成操作系统及驱动安装做成系统盘镜像后才能使用。
- ISO镜像在镜像服务页面所创建的云服务器只建议用来装机，部分功能受限。如：不支持挂载磁盘。

参考如下链接，了解利用ISO文件创建私有镜像的完整流程。

- [通过ISO文件创建Windows系统盘镜像](#)
- [通过ISO文件创建Linux系统盘镜像](#)

10.2.5 弹性云服务器存在跨区卷时如何制作整机镜像？

使用弹性云服务器制作Windows操作系统整机镜像时，不允许弹性云服务器存在跨区卷，否则制作的整机镜像创建弹性云服务器时可能会导致数据丢失。

当弹性云服务器磁盘存在跨区卷时，请先将跨区卷的数据备份，再删除跨区卷。使用已删除跨区卷的弹性云服务器制作整机镜像。使用该整机镜像创建弹性云服务器，再根据需要使用已备份的跨区卷数据重新制作跨区卷。

说明

Linux操作系统弹性云服务器存在由多个物理卷组成的卷组，或由多个物理卷组成的逻辑卷时，请参考上述操作，先将卷组或逻辑卷数据备份，删除卷组或逻辑卷，再制作整机镜像，防止数据丢失。

10.2.6 Windows 操作系统制作私有镜像为什么要执行 Sysprep 操作？

执行 Sysprep 的作用

对于需要加入域且使用域账号登录Windows操作系统的用户，在创建私有镜像前，需要执行**Sysprep**操作。否则，镜像会包含和原云服务器相关的信息，特别是SID信息。具有多个相同SID的云服务器加入域会失败。如果您的Windows系统不需要加入域等操作，您可以选择不执行Sysprep操作。

注意

配置Sysprep前，必须确保Windows操作系统正版激活。

执行 Sysprep 的限制

由于操作系统的相关限制，只能将Sysprep用于配置Windows操作系统的全新安装。可以根据需要多次运行Sysprep以构建和配置Windows安装。但是，重置Windows激活次数最多为三次。禁止使用Sysprep来重新配置已部署的Windows现有安装。

说明

在Windows命令行中输入以下命令，在“Windows Script Host”弹窗中查看还能执行Sysprep的次数。

```
slmgr /dlv
```

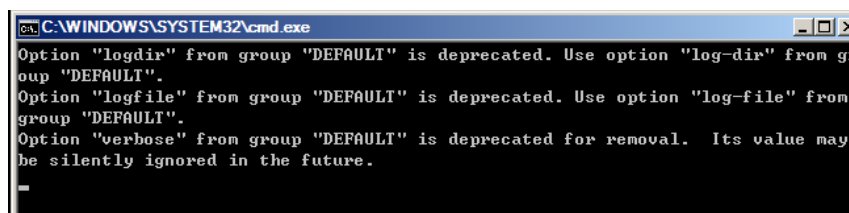
当“剩余Windows重置计数”的值为0时，无法执行Sysprep。

10.2.7 Windows 操作系统镜像执行 Sysprep 之后，使用该镜像创建的弹性云服务器启动失败怎么办？

问题描述

1. Windows操作系统镜像执行Sysprep之后，使用该镜像创建的弹性云服务器启动时出现如下图的提示信息：

图 10-1 提示信息

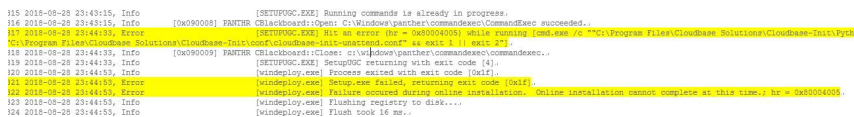


且弹出如下提示信息：

Windows无法分析或处理 pass [specialize] 的无人参与应答文件。应答文件中指定的设置无法应用。处理组件设置时检测到错误 [Microsoft-Windows-Shell-Setup]。

2. 单击提示框的“确定”，弹出如下提示信息：
计算机意外地重新启动或遇到错误。Windows安装无法继续。若要安装Windows，请单击“确定”重新启动计算机，然后重新启动安装。
3. 在“C:\Windows\Panther”路径下打开“setupact.log”文件，查看云服务器日志中是否存在如下信息。

图 10-2 查看云服务器日志



处理方法

1. 使用公共镜像创建弹性云服务器（由于执行Sysprep有次数限制，建议您使用公共镜像重新创建弹性云服务器）。
2. 您可以选择新建“Unattend.xml”文件或者修改系统自带的“Unattend.xml”。

- 你可以自行选择新建“Unattend.xml”文件（如果使用新建“Unattend.xml”文件，请确保执行Sysprep时使用的是新建的“Unattend.xml”文件），内容及书写规范请参考微软官网链接：
 - <https://docs.microsoft.com/en-us/windows-hardware/manufacture/desktop/update-windows-settings-and-scripts-create-your-own-answer-file-sxs>
 - <https://docs.microsoft.com/en-us/windows-hardware/manufacture/desktop/sysprep--system-preparation--overview>
- 修改“C:\Program Files\Cloudbase Solutions\Cloudbase-Init\conf”目录下的“Unattend.xml”文件，删除“RunSynchronous”版块。

图 10-3 删除 RunSynchronous 版块

```

- <settings pass="specialize">
- <component language="neutral" xmlns:xsl="http://www.w3.org/2001/XMLSchema-instance" xmlns:wcm="http://schemas.microsoft.com
versionScope="nonSxS" publicKeyToken="31bf3856ad364e35" processorArchitecture="amd64" name="Microsoft-Windows-Deployment">
  <RunSynchronous>
    <RunSynchronousCommand wcm:action="add">
      <Order>1</Order>
      <Path>cmd.exe /c ""C:\Program Files\Cloudbase Solutions\Cloudbase-Init\Python\Scripts\cloudbase-init.exe" --config-file
      Solutions\Cloudbase-Init\conf\cloudbase-init-unattend.conf" && exit 1 || exit 2"</Path>
      <Description>Run Cloudbase-Init to set the hostname</Description>
      <WillReboot>OnRequest</WillReboot>
    </RunSynchronousCommand>
  </RunSynchronous>
</component>
</settings>
</unattend>
    
```

3. 执行Sysprep。详细操作请参考“[执行Sysprep](#)”章节。

须知

如果使用自行新建的“Unattend.xml”文件，执行Sysprep时请注意对应的“Unattend.xml”路径，确保使用的是新建的“Unattend.xml”文件。

4. 使用执行Sysprep后的弹性云服务器重新创建镜像。

10.2.8 通过 API 无法创建 ZVHD2 格式的镜像怎么办？

问题描述

用户使用API创建ZVHD2格式镜像时，得到的镜像格式为ZVHD。

处理方法

查看用户使用的token是否带有“op_gated_ild”角色（“op_gated_ild”为公测标识，在“获取用户Token”API响应消息的body体中查看），ZVHD2格式镜像具备延迟加载特性，如果当前环境不支持延迟加载特性，或者处于公测阶段，则创建ZVHD2格式镜像不成功。

请先联系管理员，确保当前环境已支持延迟加载特性后，重新获取token，再使用新的token制作镜像。

10.3 镜像共享类

10.3.1 镜像共享 FAQ

每个镜像最多可以共享给多少个租户？

系统盘镜像和数据盘镜像为128个；整机镜像为10个。

我最多可以获得多少个共享镜像？

没有限制。

使用共享镜像是否占用我的私有镜像配额？

不占用。

共享镜像给一个账号，而这个账号没有接受或者拒绝，消耗共享配额吗？

不消耗。

别人共享过来的镜像在哪查看？

确认共享的镜像所在的区域，切换到该区域，然后在管理控制台的“服务列表 > 计算 > 镜像服务 > 共享镜像”页签下查看。

对于多项目用户，需要确认共享的镜像所在的项目，切换到该区域对应的项目，然后在管理控制台的“服务列表 > 计算 > 镜像服务 > 共享镜像”页签下查看。

需要注意，镜像在未接受时，“共享镜像”页签旁会有红点提示，并且提示用户是否接受其他租户共享的镜像；镜像在接受后，将显示在共享镜像列表中。

假如我是账号 A，我想共享系统盘镜像给账号 B，那账号 B 需要提前购买云服务器吗？

账号B不需要提前购买云服务器。账号A共享系统盘镜像后，账号B可以使用该镜像申请云服务器。

使用共享镜像创建云服务器时是否有区域限制？

有区域限制。区域A中的共享镜像只能在区域A创建云服务器。

我能把别人共享给我的镜像再共享给其他人吗？

不能直接共享，您可以先将共享镜像复制为私有镜像，再将私有镜像共享给其他人。

我把镜像共享给其他人，还能使用该镜像创建云服务器吗？

可以。您将镜像共享给其他账号后，还可以用该镜像创建云服务器，在该云服务器的基础上也可以继续创建私有镜像。

使用共享镜像创建云服务器，存在什么风险？

镜像拥有者可以查看该镜像的共享关系，也可以随时取消共享或删除该镜像。共享镜像被提供者删除后，会导致接受者使用共享镜像创建的云服务器不能重装操作系统，也不能创建相同配置的云服务器。

云平台不保证其他账号共享镜像的完整性和安全性，使用共享镜像时您需要自行承担风险，请您选择信任的账号共享的镜像。

我把镜像共享给其他账号，存在什么风险？

存在数据、文件或软件泄露的风险。在共享给其他账号之前，您需要删除敏感数据和重要文件。镜像接受方可以用这个共享镜像创建云服务器，还可以用这个云服务器创建更多私有镜像，也可能再次共享给其他人，因此，数据会不停传播，造成大范围泄露风险。

共享镜像是否支持指定区域、指定可用区？

共享镜像时只能指定项目ID，无法指定区域和可用区，即镜像只能在同一区域内共享。另外，镜像可以在一个区域的任意可用区使用。

别的账号共享过来的数据盘镜像，能恢复到我已有的数据盘中吗？

别的账号共享过来的数据盘镜像只能重新申请数据盘，不能直接恢复到已有数据盘中，您可以按照[数据盘镜像中的数据可以导入到之前已有的数据盘内吗？](#)指导进行间接恢复。

拒绝后的共享镜像如何使用？

用户B拒绝了用户A的共享镜像的情况下，如果需要使用该共享镜像，需要告知用户A按照以下步骤完成操作。

- 方法一
用户A重新将用户B添加为镜像的共享租户。请参考[添加镜像的共享租户](#)。
- 方法二
用户B重新接受已拒绝的共享镜像。请参考[接受已经拒绝的共享镜像](#)。

10.3.2 共享镜像和复制镜像的区别？

- 共享镜像：
范围为同一区域。镜像支持共享到同一区域内的其他租户。如果您需要共享到不同区域，请先复制镜像到目标区域后再共享。详情参见[共享镜像概述](#)。
- 复制镜像：
 - 复制镜像：用户可以通过复制镜像功能使镜像具备一些高级特性（如快速发放）。

共享镜像和复制镜像使用场景不同，具体如下表。

场景	操作	说明	相关文档
同区域跨账号共享	共享	共享镜像不会生成新的镜像，接受者所看到的镜像和源镜像拥有相同的ID。镜像所有者仍为共享者。	参见 共享指定镜像 。
同区域同账号复制	复制	使镜像具备一些高级特性（如快速发放）。	参见 复制镜像 。

10.3.3 创建的镜像为什么不能共享？

在镜像列表页面，有些镜像不支持共享功能，因此在“操作”列没有提供“共享”按钮。

镜像支持共享到同一区域内的其他租户。

共享镜像详细信息请参见[共享镜像](#)。

10.4 操作系统类

10.4.1 如何选择操作系统？

- Linux
适用于Linux环境下的开发平台或运营业务。当前提供CentOS（推荐）和Ubuntu等类型操作系统。
系统盘要求40GB及以上、内存512MB及以上。
- 4G以上内存操作系统选择
如果使用4GB及以上内存，请选择64位的操作系统（32位操作系统只能4GB内寻址，超过4GB的内存操作系统无法访问）。

10.4.2 UEFI 启动方式与 BIOS 启动方式有哪些区别？

表 10-2 UEFI 与 BIOS 启动方式的区别

启动方式	说明	特点
BIOS	基本输入输出系统（Basic Input Output System，简称BIOS）保存了云服务器重要的基本输入输出的程序、系统设置信息、开机后自检程序和系统自启动程序。	为云服务器提供最直接的设置和控制。
UEFI	统一可扩展固件接口（Unified Extensible Firmware Interface，简称UEFI）是一种详细描述类型接口的标准，可用于操作系统自动从预启动的操作环境加载到一种操作系统上。	缩短了启动时间和从休眠状态恢复的时间。

10.4.3 Windows 弹性云服务器如何删除多余的网络连接？

方法一

1. 按“Win+R”键打开运行对话框，输入regedit并按回车键，打开注册表编辑器。

📖 说明

修改注册表可能会导致系统启动失败，请在修改前备份注册表。

2. 打开至以下注册表键值：
“HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\NetworkList\Profiles” 中可能包含多个子项，且这些子项名称为数字与字母组成的序列。依次单击每个子项，检查右侧窗口中名为“ProfileName”这个键值所对应的“数据”列。
3. 双击“ProfileName”，将“数值数据”修改为待更改的网络名称。
4. 重启云服务器，使修改生效。

方法二

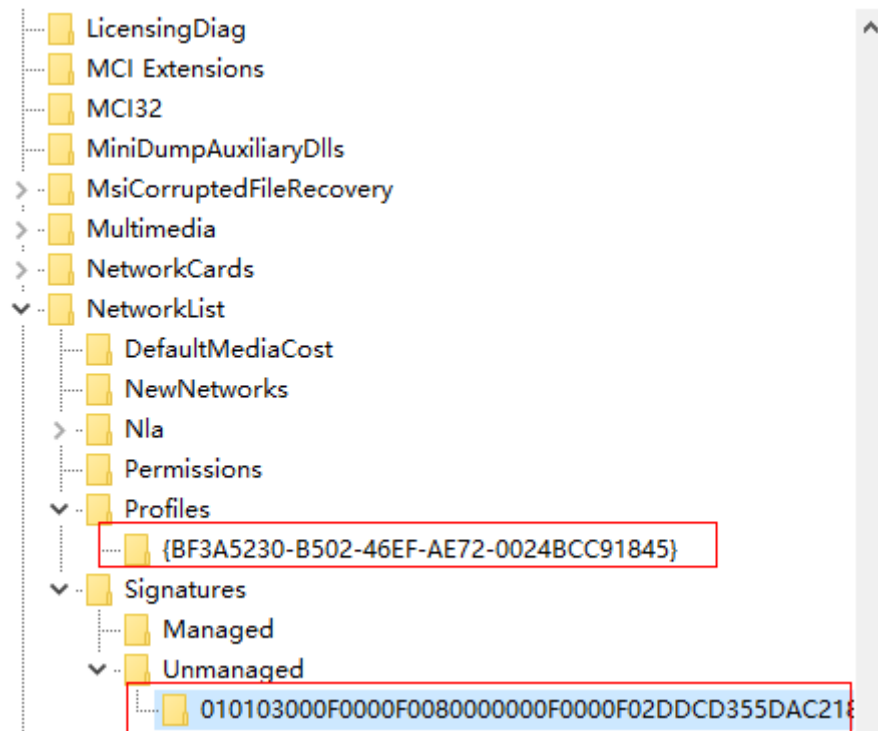
1. 按“Win+R”键打开运行对话框，输入regedit并按回车键，打开注册表编辑器。

📖 说明

修改注册表可能会导致系统启动失败，请在修改前备份注册表。

2. 打开至以下注册表键值。
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\NetworkList\Profiles
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\NetworkList\Signatures\Unmanaged
3. 删除如下注册表删除键下的目录。

图 10-4 注册表目录



📖 说明

- 如果文件最后有“exit 0”，则在文件最后的“exit 0”前一行输入以上内容。如果文件最后无“exit 0”，则在文件最后一行输入。
- 使用SUSE 11 SP4镜像创建的云服务器，内存需要大于等于4G时才能支持一键式密码重置功能。

操作步骤

- CoreOS操作系统

请执行如下命令完成一键式重置密码插件的启动：

```
cat >/etc/systemd/system/cloudResetPwdAgent.service <<EOT
[Unit]
Description=cloudResetPwdAgent service
Wants=local-fs.target
Requires=local-fs.target
[Service]
Type=simple
ExecStart=/CloudrResetPwdAgent/bin/cloudResetPwdAgent.script start
RemainAfterExit=yes
ExecStop=/CloudrResetPwdAgent/bin/cloudResetPwdAgent.script stop
KillMode=none
[Install]
WantedBy=multi-user.target
EOT
systemctl enable cloudResetPwdAgent.service
```

- SUSE、Ubuntu或Debian操作系统

a. 执行如下命令，打开rc文件。

```
vi /etc/init.d/rc
```

b. 按“i”进入编辑模式，在文件末尾补充如下内容：

```
/CloudrResetPwdAgent/bin/cloudResetPwdAgent.script start
```

c. 按“Esc”后，输入:**wq**，按“Enter”，保存设置并退出。

- CentOS 7系列操作系统

a. 执行如下命令，打开rc.local文件。

```
chmod +x /etc/rc.d/rc.local
```

```
vi /etc/rc.d/rc.local
```

b. 按“i”进入编辑模式，在文件末尾补充如下内容：

```
/CloudrResetPwdAgent/bin/cloudResetPwdAgent.script start
```

c. 按“Esc”后，输入:**wq**，按“Enter”，保存设置并退出。

- openSUSE 13系列操作系统

a. 执行如下命令，打开boot.local文件。

```
vi /etc/init.d/boot.local
```

b. 按“i”进入编辑模式，在文件末尾补充如下内容：

```
/CloudrResetPwdAgent/bin/cloudResetPwdAgent.script start
```

c. 按“Esc”后，输入:**wq**，按“Enter”，保存设置并退出。

- Debian 8系列操作系统

a. 执行如下命令，打开rc.local文件。

```
vi /etc/rc.local
```

b. 按“i”进入编辑模式，在文件末尾补充如下内容：

```
/CloudrResetPwdAgent/bin/cloudResetPwdAgent.script start
```


- c. 按“Esc”后，输入:**wq**，按“Enter”，保存设置并退出。
- Fedora 20系列操作系统
 - a. 执行如下命令，打开rc.local文件。

```
touch /etc/rc.d/rc.local
chmod +x /etc/rc.d/rc.local
vi /etc/rc.d/rc.local
```
 - b. 按“i”进入编辑模式，在文件末尾补充如下内容：

```
/CloudrResetPwdAgent/bin/cloudResetPwdAgent.script start
```
 - c. 按“Esc”后，输入:**wq**，按“Enter”，保存设置并退出。
- 其他操作系统
 - a. 执行如下命令，打开rc.local文件。

```
vi /etc/rc.d/rc
```
 - b. 按“i”进入编辑模式，在文件末尾补充如下内容：

```
/CloudrResetPwdAgent/bin/cloudResetPwdAgent.script start
```
 - c. 按“Esc”后，输入:**wq**，按“Enter”，保存设置并退出。

10.4.6 如何解决 SR-IOV 场景下 Windows 7 弹性云服务器安装 82599 网卡驱动报错？

问题描述

从Intel官网下载最新的20.4.1版本驱动包（下载地址：<https://downloadcenter.intel.com/search?keyword=Intel++Ethernet+Connections+CD>），该版本驱动包在Windows7 64位SR-IOV直通弹性云服务器上安装时会提示“找不到英特尔适配器”错误。

原因分析

Intel 82599直通网卡在未安装驱动时被操作系统识别为以太网控制器设备，20.4.1版本驱动包在安装时未能正确识别出Intel网卡设备，导致程序报错。

处理方法

在20.4.1驱动包文件夹下运行Autorun.exe。安装驱动包前先给网卡安装驱动，使网卡被系统识别为Intel 82599 VF设备，安装驱动有两种方法。

- 方法1：通过版本更新方法安装驱动
 - a. 从Intel官网下载18.6版本驱动包。
 - b. 到18.6驱动包安装文件夹下运行Autorun.exe进行安装。
 - c. 成功后到20.4.1版本驱动包文件夹下运行Autorun.exe更新驱动。
- 方法2：设备管理器手动安装驱动
 - a. 打开Windows资源管理器，右键单击“计算机”，选择“管理”，打开“设备管理器”，在设备管理器中找到网卡，未安装驱动时，网卡位于“其他设备”一栏，名字为“以太网控制器”。
 - b. 右键单击“以太网控制器”，选择“更新驱动程序软件”。
 - c. 单击“浏览”，选择驱动包所在路径，单击“下一步”。

- d. 驱动安装成功后，可以在设备管理器的“网络适配器”一栏看到网卡。
- e. 单击“Autorun.exe”安装20.4.1版本一键安装驱动包。

10.4.7 云服务器创建或切换操作系统时，为什么选不到我的私有镜像？

用户在创建云服务器或为云服务器切换操作系统时，有时会选不到自己的私有镜像。可能原因是x86与ARM架构不兼容，或者UEFI与BIOS启动方式不兼容等。详细说明如下：

- 通过x86 CPU架构的云服务器创建的私有镜像，不能用于创建鲲鹏CPU架构的云服务器，也不能在鲲鹏CPU架构云服务器切换操作系统时使用。反之亦然。
- 通过外部镜像文件创建私有镜像时，若架构类型选择“x86”，则该私有镜像不能用于创建鲲鹏CPU架构的云服务器，也不能在鲲鹏CPU架构云服务器切换操作系统时使用。反之亦然。
- 通过BIOS启动方式的云服务器创建的私有镜像，不能用于创建UEFI启动方式的云服务器，也不能在UEFI启动方式的云服务器切换操作系统时使用。反之亦然。
- 通过外部镜像文件创建私有镜像时，若启动方式选择“BIOS”，则该私有镜像不能用于创建UEFI启动方式的云服务器，也不能在UEFI启动方式的云服务器切换操作系统时使用。反之亦然。

10.5 镜像导入类

10.5.1 除了文档中支持的镜像格式类型，我可以其他镜像格式吗？

目前支持VMDK、VHD、RAW、QCOW2、VHDX、QED、VDI、QCOW、ZVHD2、ISO和ZVHD格式的镜像导入。

不支持“-flat.vmdk”格式的镜像、包含任何快照卷或者包含差分卷的镜像文件包。用户可以使用qemu-img开源工具将对应的镜像转换成支持的格式后，再进行上传。

📖 说明

Windows平台下qemu-img开源工具的安装和使用可以参考以下地址：

<https://cloudbase.it/qemu-img-windows/>

10.5.2 没有对云服务器进行相关预操作会带来什么影响？

通过云服务器或者外部镜像文件创建私有镜像时，需要对云服务器或镜像文件所在虚拟机进行相关预操作。否则，可能造成以下影响：

1. 如果您没有将云服务器网卡配置成DHCP，或者没有删除残留的udev规则。那么，通过镜像文件注册的私有镜像所创建出的云服务器仍和源镜像文件中的配置保持一致，或者出现云服务器的网卡不从eth0开始的情况。此时，您需要远程登录云服务器，对云服务器进行相关配置。
2. 如果用于创建Linux云服务器的镜像没有提前进行相关配置，在创建Linux云服务器过程中可能会导致以下三种情况出现。
 - 无法实现自定义密码注入。

- 无法注入证书。
 - 无法实现其他可能的云服务器自定义配置。
3. 如果您没有清理“fstab”文件中对用户磁盘的自动挂载检测相关信息，可能会导致创建出来的云服务器在启动时出现异常。

10.5.3 如果操作系统类型选择错误或者系统磁盘容量填写错误怎么办？

通过镜像文件注册私有镜像时，如果选择了错误的操作系统，可能会导致创建云服务器失败。

通过镜像文件注册私有镜像时，如果填写的系统磁盘容量小于镜像文件内部的系统盘容量，会导致创建镜像失败。

如果出现上述情况，请您删除使用错误参数注册的镜像，填写正确的配置参数重新创建私有镜像。

10.5.4 为什么 VHD 格式的镜像上传失败，任务中心查看报错为外部镜像文件的系统盘容量大于用户设置的系统磁盘容量？

如果用户通过外部镜像文件注册私有镜像失败，且在任务中心查看错误信息为外部镜像文件的系统盘容量大于用户设置的系统磁盘容量。此问题通常有两种情况：

1. 用户设置的系统盘大小小于原来平台虚拟机的系统盘大小，请重新确认原镜像文件的系统盘大小并重新注册。
2. 用户使用的vhd格式是通过qemu-img或者其他类似工具转换生成，产生的vhd格式的实际虚拟磁盘大小（virtual size）和用户原虚拟机系统盘大小不一致，具体原因参考：<https://bugs.launchpad.net/qemu/+bug/1490611>

此情况下可以通过qemu-img工具的qemu-img info命令检查处理。

```
[xxxx@xxxx test]$ qemu-img info 2g.vhd
image: 2g.vhd
file format: vpc
virtual size: 2.0G (2147991552 bytes)
disk size: 8.0K
cluster_size: 2097152
```

计算virtual size的bytes换算成以G为单位的值是否为整数值。如上所示，2147991552 bytes通过换算得到大小约为2.0004G，大于2G，因此上传时需要设置系统盘大小大于2G的值。

10.6 镜像导出类

10.6.1 制作的私有镜像可以下载到本地吗？支持哪些镜像格式？

用户制作的私有镜像可以通过[导出镜像](#)功能下载到本地。

目前支持导出VMDK、VHD、QCOW2和ZVHD格式的镜像。

创建的私有镜像默认格式为zvhd2，导出过程中，不同的格式会导致导出的镜像大小不同。

10.6.2 为什么镜像导出到 OBS 桶后，大小和在镜像服务中显示的不一致？

问题描述

私有镜像导出到OBS桶后，大小和在镜像服务中显示的不一致。

例如：私有镜像在镜像服务页面显示的镜像大小为1.04 GB，导出到OBS桶后，大小为2.91 GB。

原因分析

创建的私有镜像默认格式为zvhd2，在导出到OBS桶时，会先让用户选择导出的文件格式，目前支持导出VMDK、VHD、QCOW2和ZVHD格式的镜像，导出后的文件大小根据所选择的文件格式来决定。

10.6.3 公共镜像能否直接下载到本地，怎么操作？

暂不支持直接下载公共镜像，您可以先通过公共镜像创建云服务器，再将云服务器制作作为私有镜像，然后导出私有镜像至个人OBS桶，再下载至本地。

参考链接如下：

- [通过云服务器创建Windows系统盘镜像](#)或[通过云服务器创建Linux系统盘镜像](#)
- [导出镜像](#)

说明

- Windows、SUSE、Red Hat、Ubuntu、Oracle Linux操作系统的公共镜像及此类公共镜像创建的私有镜像，均不支持导出。
- 若您使用Windows、SUSE、Red Hat、Ubuntu、Oracle Linux私有镜像文件（非来源于公共镜像）通过线下导入的方式创建云服务器，基于该云服务器制作的系统盘镜像支持导出。

10.6.4 镜像为什么没有“导出”按钮？

在镜像列表页面，有些镜像不支持导出功能，因此在“操作”列没有提供“导出”按钮。以下镜像不支持导出功能：

- 公共镜像
- 整机镜像
- ISO镜像
- Windows、SUSE公共镜像所创建的私有镜像

10.7 镜像优化类

10.7.1 一定要在云服务器中安装 Guest OS driver 吗？

云服务器中安装Guest OS driver，可以增强云服务器使用时用户的体验。同时，可以保证云服务器的高可靠性以及稳定性。

- Windows操作系统必须安装PV driver和virtio。

- Linux操作系统必须使用xen-pv以及virtio原生驱动并装载到initrd。

10.7.2 Windows 操作系统为什么要安装并更新 virtio?

为什么要安装 virtio 驱动?

virtio是为弹性云服务器提供高性能磁盘和网卡的半虚拟化驱动（virtio driver）。

- 标准的Windows系统不会自带virtio driver。
- 平台提供的公共镜像默认已安装virtio。
- 私有镜像需用户自行安装virtio，安装指导请参见[安装virtio驱动](#)。

为什么要更新 virtio?

保持最新的驱动版本能有效避免触发社区发现的已知问题。

用户需要做哪些操作?

- 根据指导升级Windows私有镜像或正在运行的Windows实例中的驱动程序。
- 如有任何技术问题或疑问，请联系管理员获取帮助。

10.7.3 通过镜像文件注册私有镜像过程中，系统会对镜像做哪些修改?

建议您勾选“进行后台自动化配置”，勾选后，后台系统会自动进行以下操作。

Linux 操作系统

- 检查pvdriver相关驱动是否存在，如果存在，将删除相关驱动。
- 修改grub和syslinux配置文件，增加内核启动参数，并将磁盘分区名改成“UUID=*磁盘分区的UUID*”。
- 检查“/etc/fstab”文件中的磁盘分区名并修改成“UUID=*磁盘分区的UUID*”。
- 检查“initrd”文件是否含有XEN和IDE驱动，若不含有，则加载XEN和IDE驱动。
- 修改xwindow配置文件“/etc/X11/xorg.conf”文件，避免图形化界面显示失败。
- 删除VMware tools对应的服务。
- 记录最近一次自动化配置对镜像做的修改，记录文件位置为“/var/log/rainbow_modification_record.log”。
- Linux操作系统会拷贝自带的virtio前端驱动到initrd或initramfs中去。具体情况请参见[外部镜像文件支持的格式和操作系统类型](#)。

📖 说明

对于如下场景的镜像文件，勾选“进行后台自动化设置”后，该功能不生效：

- /usr目录被单独分区的镜像文件
- 使用xfs文件系统的Fedora 29 64bit、Fedora 30 64bit、CentOS 8.0 64bit镜像
- 使用ext4文件系统的SUSE 12 SP4 64bit镜像

Windows 操作系统

- 对IDE驱动进行修复，系统第一次启动使用IDE启动。
- 对鼠标和键盘对应的注册表键值删除，让鼠标键盘注册表键值在新的平台上重新生成，保证鼠标键盘可用。
- 对PV Driver注册表键值修复，解决系统无法安装驱动问题和XEN驱动冲突问题。
- 修复DHCP，系统会根据DHCP协议动态获取IP地址等信息。

10.7.4 创建私有镜像前云服务器或镜像文件需要完成哪些初始化配置？

镜像源为云服务器或镜像文件

表 10-3 云服务器相关配置项

操作系统	相关配置项	参考链接
Windows	<ul style="list-style-type: none"> • 设置网卡属性为DHCP • 开启远程桌面连接功能 • 安装一键式重置密码插件 • (可选) 安装Cloudbase-Init工具 • 安装Guest OS driver, 包括PV driver和virtio驱动 • 执行Sysprep操作 	通过云服务器创建Windows系统盘镜像
Linux	<ul style="list-style-type: none"> • 设置网卡属性为DHCP • (可选) 安装Linux特殊驱动 • 安装一键式重置密码插件 • (可选) 安装Cloud-Init工具 • 清理网络规则文件 • 修改grub文件的磁盘标识方式为UUID • 修改fstab文件的磁盘标识方式为UUID • 安装原生的XEN和KVM驱动 • 卸载云服务器的数据盘 	通过云服务器创建Linux系统盘镜像

表 10-4 镜像文件（用于创建云服务器）相关配置项

操作系统	相关配置项	参考链接
Windows	<ul style="list-style-type: none"> 设置网卡属性为DHCP 开启远程桌面连接功能 安装Guest OS driver，包括PV driver和virtio驱动 （可选）安装Cloudbase-Init工具 （可选）开启网卡多队列 （可选）配置IPv6地址 	准备镜像文件（Windows）
Linux	<ul style="list-style-type: none"> 清理网络规则文件 设置网卡属性为DHCP 安装原生的XEN和KVM驱动 修改grub文件的磁盘标识方式为UUID 修改fstab文件的磁盘标识方式为UUID 清除“/etc/fstab”中非系统盘的自动挂载信息 （可选）安装Cloud-Init工具 （可选）开启网卡多队列 （可选）配置IPv6地址 	准备镜像文件（Linux）

📖 说明

- 使用外部镜像文件创建私有镜像时，以上相关步骤操作需要在虚拟机内部完成，强烈建议您在原平台的虚拟机实施修改后，再导出镜像。
- 使用Windows外部镜像文件创建私有镜像时，关于Guest OS driver，在勾选“进行后台自动化配置”后，云平台会对镜像文件进行检查，如果未添加Guest OS driver，会尽力帮助客户进行自动安装。

10.7.5 Windows 外部镜像文件在导出前未完成初始化配置，怎么办？

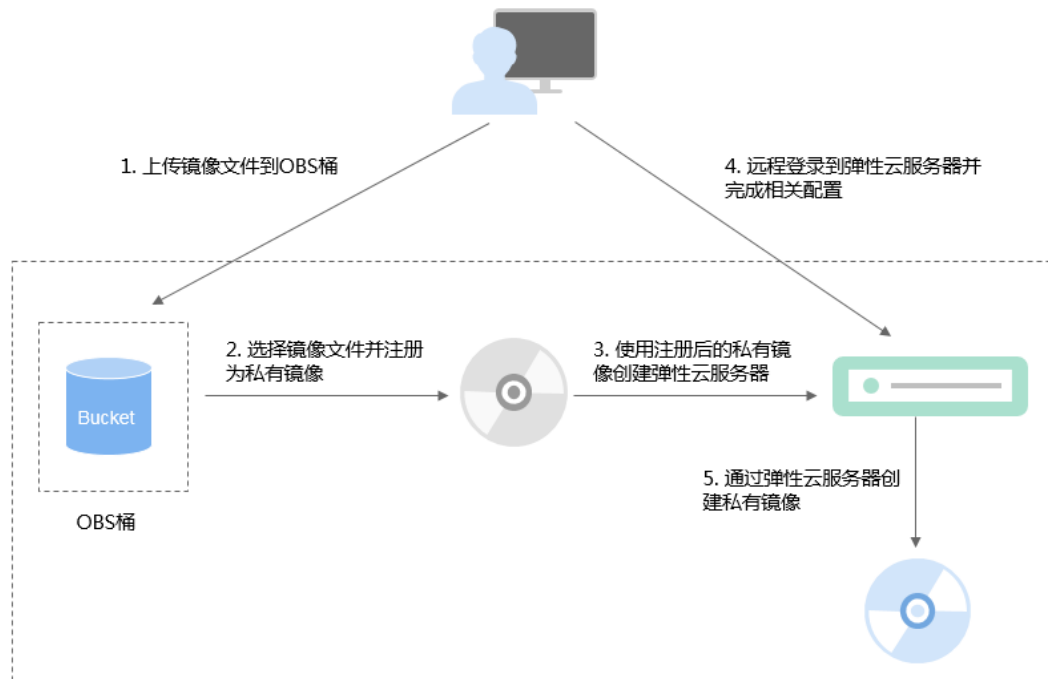
外部镜像文件在从原平台导出前，没有按照表3-5的要求完成初始化操作，推荐您使用弹性云服务器完成相关配置。流程如图10-7所示。

注意

云服务器的正常运行依赖于XEN Guest OS driver (PV driver) 和KVM Guest OS driver (virtio)，未安装会对云服务器运行时的性能产生影响，云服务器的部分功能会有缺失。请确保外部镜像文件在从原平台导出前，已安装这些驱动，否则云服务器因启动不成功而无法进行任何配置。

- 安装PV driver，请参考[安装PV driver](#)。
- 安装virtio，请参考[安装virtio驱动](#)。

图 10-7 创建过程



步骤 1：上传镜像文件

上传外部镜像文件到OBS个人桶中，请参考[上传镜像文件（Windows）](#)。

步骤 2：注册镜像

通过管理控制台选择上传的镜像文件，并将镜像文件注册为未初始化的私有镜像。请参考[注册镜像（Windows）](#)。

步骤 3：创建云服务器

1. 登录IMS控制台。
 - a. 登录管理控制台。
 - b. 选择“计算 > 镜像服务”。
进入镜像服务页面。
2. 单击“私有镜像”页签进入私有镜像列表。
3. 在未初始化的私有镜像所在行，单击“操作”列的“申请服务器”。

4. 根据界面提示完成云服务器的创建。需要关注以下几点：
 - 创建云服务器过程中需要绑定弹性公网IP，以便您能上传相关安装包或者在云服务器内部直接下载安装包。
 - 需要在云服务器所属安全组中添加对应的入方向规则，确保可以从外部访问该云服务器。
 - 如果在镜像文件系统中已经安装并配置了Cloudbase-Init工具，请按照界面提示设置密码方式登录云服务器。如果未安装，请使用镜像文件中包含的密码或证书登录。详细操作请参见《弹性云服务器用户指南》。
5. 执行以下步骤检查云服务器，验证私有镜像是否可用。
 - a. 如果云服务器启动成功，证明外部镜像文件已安装Guest OS driver，或者云平台的自动化配置功能已为您自动安装该驱动。否，请您在原平台安装Guest OS driver后，再重新创建私有镜像。
 - b. 如果能够使用用户设置的密码/密钥登录云服务器，证明已安装Cloudbase-Init工具。否，请使用原始密码/密钥登录云服务器，再请参考[安装并配置Cloudbase-Init工具](#)安装Cloudbase-Init。
 - c. 请参考[步骤4：配置云服务器](#)中的第2步检查网卡属性是否为DHCP方式。
 - d. 使用MSTSC工具远程登录成功，证明云服务器的远程桌面配置正常。否，请参考[步骤4：配置云服务器](#)中的第3步配置远程桌面。如果满足以上条件，则私有镜像可用，请直接参考[清理环境（可选）](#)清理环境。

步骤 4：配置云服务器

远程登录[步骤3：创建云服务器](#)中创建的云服务器，完成网络配置、软件安装等操作。

1. 登录云服务器。
2. 检查网卡属性是否为DHCP。如果云服务器网络配置为静态IP地址，请参考[设置网卡属性为DHCP（Windows）](#)修改为DHCP方式。
3. 对于需要使用Windows远程桌面连接方式进行访问的云服务器，需要开启远程桌面连接功能，请参考[开启远程桌面连接功能](#)。
4. （可选）配置增值功能
 - 安装并配置Cloudbase-Init，请参考[安装并配置Cloudbase-Init工具](#)。
 - 开启网卡多队列，请参考[如何设置镜像的网卡多队列属性？](#)。
 - 配置IPv6地址，请参考[如何配置IPv6地址](#)。

步骤 5：通过云服务器创建镜像

通过云服务器创建正常状态的私有镜像，请参考[通过云服务器创建Windows系统盘镜像](#)。

清理环境（可选）

以上步骤中，未初始化的镜像文件以及创建的云服务器会额外占用存储和计算资源空间，因此建议您在完成镜像注册后清理环境。

- 删除在[步骤2：注册镜像](#)中注册的未初始化镜像。
- 删除在[步骤3：创建云服务器](#)中创建的云服务器。

- 删除OBS个人桶中的镜像文件。

10.7.6 Linux 外部镜像文件在导出前未完成初始化配置，怎么办？

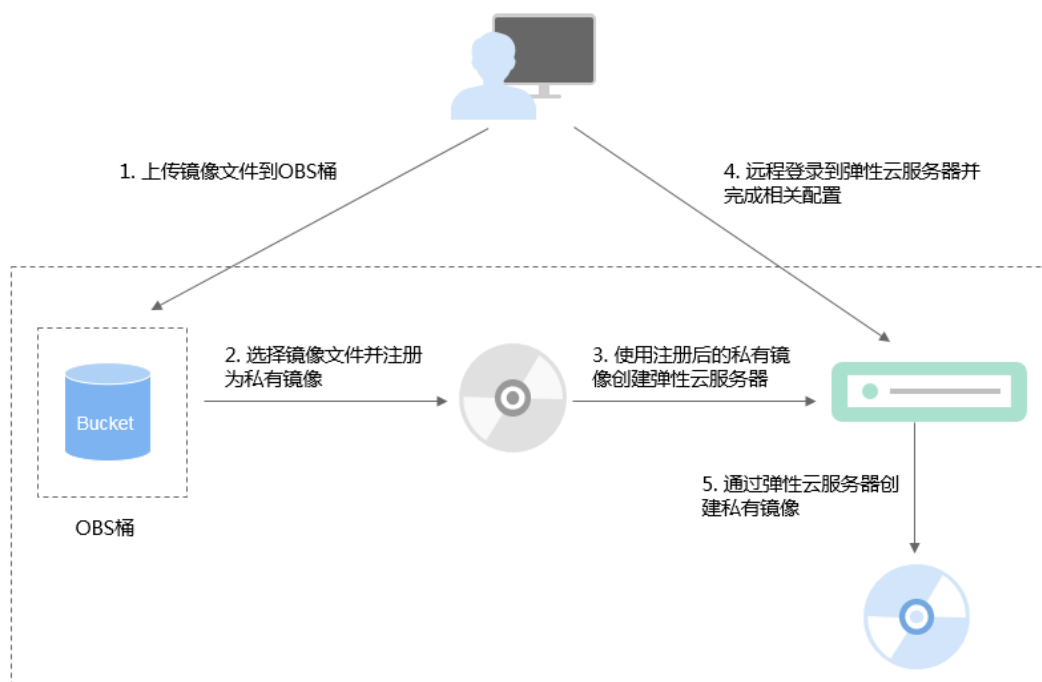
外部镜像文件在从原平台导出前，没有按照表3-9的要求完成初始化操作，推荐您使用弹性云服务器完成相关配置。流程如图10-8所示。

⚠ 注意

云服务器的正常运行依赖于XEN和KVM驱动，未安装会对云服务器运行时的性能产生影响，云服务器的部分功能会有缺失。请确保外部镜像文件在从原平台导出前，已安装这些驱动，否则云服务器因启动不成功而无法进行任何配置。

- 对于XEN虚拟化类型，请安装原生的XEN驱动和KVM驱动。具体操作请参见[怎样安装原生的XEN和KVM驱动](#)。
- 对于KVM虚拟化类型，请安装原生的KVM驱动。具体操作请参见[安装原生的KVM驱动](#)。

图 10-8 创建过程



步骤 1：上传镜像文件

上传外部镜像文件到OBS个人桶中，请参考[上传镜像文件（Linux）](#)。

步骤 2：注册镜像

通过管理控制台选择上传的镜像文件，并将镜像文件注册为未初始化的私有镜像。请参考[注册镜像（Linux）](#)。

步骤 3：创建云服务器

使用未初始化的私有镜像创建弹性云服务器。

1. 登录IMS控制台。
 - a. 登录管理控制台。
 - b. 选择“计算 > 镜像服务”。
 进入镜像服务页面。
2. 单击“私有镜像”页签进入私有镜像列表。
3. 在未初始化的私有镜像所在行，单击“操作”列的“申请服务器”。
4. 根据界面提示完成云服务器的创建。需要关注以下几点：
 - 需要在云服务器所属安全组中添加对应的入方向规则，确保可以从外部访问该云服务器。
 - 如果镜像文件中已安装了Cloud-Init工具，请按照界面提示设置登录密码；如果未安装，请使用镜像文件中包含的密码或证书登录。
 详细操作请参见“《弹性云服务器用户指南》”。
5. 请执行以下步骤检查云服务器，验证私有镜像是否可用。
 - a. 如果云服务器启动成功，证明外部镜像文件已安装原生XEN和KVM驱动，或者云平台的自动化配置功能已为您自动安装该驱动。否，请您在原平台安装原生XEN和KVM驱动后，再重新创建私有镜像。
 - b. 如果能够使用用户设置的密码/密钥登录云服务器，证明已安装Cloud-Init工具。否，请使用原始密码/密钥登录云服务器，再参考[安装Cloud-Init工具](#)安装Cloud-Init工具。
 - c. 请参考[步骤4：配置云服务器](#)检查网络相关配置。

如果满足以上条件，则私有镜像可用，请直接参考[清理环境（可选）](#)清理环境。

步骤 4：配置云服务器

远程登录[步骤3：创建云服务器](#)中创建的云服务器，完成网络配置、软件安装等操作。

1. 登录云服务器。
2. 配置网络
 - 检查云服务器的私有IP是否和控制台显示的私有IP一致（可以通过[ifconfig](#)查看）。如果不一致，请参考[清理网络规则文件](#)清理网络规则文件。
 - 检查网卡属性是否为DHCP。如果云服务器网络配置为静态IP地址，请参考[设置网卡属性为DHCP（Linux）](#)修改为DHCP方式。
 - 检查SSH服务是否为开启状态（可以通过[service sshd status](#)查看）。如果未开启，请执行[service sshd start](#)。请确保您的云服务器防火墙（例如：Linux iptables）允许访问SSH。
3. 配置文件系统
 - 修改grub文件的磁盘标识方式为UUID，请参考[修改grub文件磁盘标识方式为UUID](#)。
 - 修改fstab文件的磁盘标识方式为UUID，请参考[修改fstab文件磁盘标识方式为UUID](#)。
 - 清除“/etc/fstab”中非系统盘的自动挂载信息，避免对后续挂载数据盘可能带来影响。请参考[卸载云服务器的数据盘](#)。

4. （可选）配置增值功能
 - 安装并配置Cloud-Init，请参考[安装Cloud-Init工具](#)和[配置Cloud-Init工具](#)。
 - 开启网卡多队列，请参考[如何设置镜像的网卡多队列属性？](#)。
 - 配置IPv6地址，请参考[如何配置IPv6地址](#)。

步骤 5：通过云服务器创建镜像

通过云服务器创建正常状态的私有镜像，请参考[通过云服务器创建Linux系统盘镜像](#)。

清理环境（可选）

以上步骤中，未初始化的镜像文件以及创建的云服务器会额外占用存储和计算资源空间，因此建议您在完成镜像注册后清理环境。

- 删除在[步骤2：注册镜像](#)中注册的未初始化镜像。
- 删除在[步骤3：创建云服务器](#)中创建的云服务器。
- 删除OBS个人桶中的镜像文件。

10.7.7 如何设置镜像的网卡多队列属性？

操作场景

随着网络IO的带宽不断提升，单核CPU处理网络中断存在瓶颈，不能完全满足网卡的需求，通过开启网卡多队列功能，您可以将弹性云服务器中的网卡中断分散给不同的CPU处理，以满足网卡的需求，从而提升网络PPS和带宽性能。

网卡多队列支持列表

网卡多队列的支持情况和实例规格、虚拟化类型、镜像的操作系统有关，只有同时满足这些要求，云服务器才能开启网卡多队列功能。

- 支持网卡多队列的实例规格请参见《弹性云服务器用户指南》的“实例”章节。

说明

网卡多队列数为大于1的值，表示支持网卡多队列。

- 虚拟化类型必须为KVM，XEN类型不支持网卡多队列。
- [表10-5](#)所列的Linux公共镜像，支持网卡多队列。

说明

- Windows操作系统公共镜像暂未支持网卡多队列，如果对Windows操作系统镜像开启网卡多队列功能，可能会引起操作系统启动速度变慢等问题。
- Linux操作系统云服务器建议将操作系统内核版本升级至2.6.35及以上，否则不支持网卡多队列。
建议您使用命令**uname -r**查询内核版本，如果低于2.6.35请联系技术支持升级内核。

表 10-5 KVM 弹性云服务器网卡多队列支持列表

类别	镜像	支持度
Windows	Windows Server 2008 WEB R2 64bit	可通过私有镜像支持
	Windows Server 2008 R2 Standard/Datacenter/Enterprise 64bit	可通过私有镜像支持
	Windows Server 2012 R2 Standard/Datacenter 64bit	可通过私有镜像支持
	Windows Server 2016 Standard/Datacenter 64bit	可通过私有镜像支持
Linux	Ubuntu 14.04/16.04 Server 64bit	支持
	openSUSE 42.2 64bit	支持
	SUSE Enterprise 12 SP1/SP2 64bit	支持
	CentOS 6.8/6.9/7.0/7.1/7.2/7.3/7.4/7.5/7.6 64bit	支持
	Debian 8.0.0/8.8.0/8.9.0/9.0.0 64bit	支持
	Fedora 24/25 64bit	支持
	EulerOS 2.2 64bit	支持

操作说明

假设以下场景所述的云服务器满足规格和虚拟化类型要求：

- 使用[网卡多队列支持列表](#)中的公共镜像创建的云服务器，默认已开启网卡多队列，无需执行本节操作。
- 对于私有镜像场景，如果您的外部镜像文件的操作系统在[网卡多队列支持列表](#)范围内，需要按照如下流程开启网卡多队列：
 - a. [将外部镜像文件导入镜像服务控制台](#)。
 - b. [为镜像添加网卡多队列标签](#)。
 - c. [使用私有镜像创建云服务器](#)。
 - d. [开启网卡多队列](#)。

将外部镜像文件导入镜像服务控制台

将外部镜像文件导入镜像服务控制台，详细操作请参见[注册镜像（Linux）](#)。

为镜像添加网卡多队列标签

Windows操作系统暂未商用支持网卡多队列，如果对Windows操作系统镜像添加网卡多队列标签，开启网卡多队列功能，可能会引起操作系统启动速度变慢等问题。

用户可以选择以下任意一种方式设置镜像的网卡多队列属性。

方式1:

1. 登录IMS控制台。
 - a. 登录管理控制台。
 - b. 选择“计算 > 镜像服务”。
进入镜像服务页面。
2. 单击“私有镜像”页签，在对应镜像所在行的“操作”列下，单击“修改”。
3. 设置镜像的网卡多队列属性。

方式2:

1. 登录IMS控制台。
 - a. 登录管理控制台。
 - b. 选择“计算 > 镜像服务”。
进入镜像服务页面。
2. 单击“私有镜像”页签，在镜像列表中，单击镜像名称，进入镜像详情页面。
3. 单击右上角的“修改”，在弹出的“修改镜像”对话框中，设置镜像的网卡多队列属性。

方式3: 通过API为镜像添加网卡多队列标签hw_vif_multiqueue_enabled

1. 获取Token的方法请参考《镜像服务API参考》中的“如何调用API > 认证鉴权”。
2. “更新镜像信息”的API使用方法请参考《镜像服务API参考》中的“更新镜像信息（OpenStack原生）”章节。
3. 在请求消息头中增加“X-Auth-Token”。
“X-Auth-Token”的取值为步骤1中获取的Token。
4. 在请求消息头中增加“Content-Type”。
“Content-Type”取值为application/openstack-images-v2.1-json-patch

请求的URI格式为：

```
PATCH /v2/images/{image_id}
```

请求的body体如下所示。

```
[
  {
    "op": "add",
    "path": "/hw_vif_multiqueue_enabled",
    "value": true
  }
]
```

使用私有镜像创建云服务器

使用注册好的私有镜像创建云服务器，详细操作请参见《弹性云服务器用户指南》。在配置参数时，需要注意以下两点：

- 区域：必须选择私有镜像所在的区域。
- 镜像：选择“私有镜像”，并在下拉列表中选择需要的镜像。

开启网卡多队列

使用Windows操作系统的KVM弹性云服务器，可通过私有镜像支持网卡多队列功能。

使用Linux操作系统的弹性云服务器，本节以CentOS 7.4为例，介绍开启网卡多队列的具体操作：

步骤1 开启网卡多队列功能

1. 登录弹性云服务器。
2. 执行以下命令，查看网卡支持和已开启的队列数。

```
ethtool -l 网卡
```

3. 执行以下命令，设置网卡当前使用的队列数。

```
ethtool -L 网卡 combined 队列数
```

示例：

```
[root@localhost ~]# ethtool -l eth0 #查询网卡eth0的队列数
Channel parameters for eth0:
Pre-set maximums:
RX:          0
TX:          0
Other:       0
Combined: 4 #表示此网卡最多支持设置开启4个队列
Current hardware settings:
RX:          0
TX:          0
Other:       0
Combined: 1 #表示当前开启的是1个队列

[root@localhost ~]# ethtool -L eth0 combined 4 #设置eth0网卡开启4个队列
```

步骤2 （可选）建议开启irqbalance服务，让系统自动调整网卡中断在多个CPU核上的分配。

1. 执行以下命令，开启irqbalance服务。

```
service irqbalance start
```

2. 执行以下命令，查询irqbalance服务状态。

```
service irqbalance status
```

如果回显中“Active”字段的值包含“active (running)”，表示irqbalance服务已正常开启，如图10-9所示。

图 10-9 irqbalance 服务正常开启

```
[root@localhost ~]# service irqbalance status
Redirecting to /bin/systemctl status irqbalance.service
irqbalance.service - irqbalance daemon
Loaded: loaded (/usr/lib/systemd/system/irqbalance.service; enabled; vendor preset: enabled)
Active: active (running) since Wed 2018-08-15 10:27:30 CST; 4h 5min ago
Main PID: 850 (irqbalance)
CGroup: /system.slice/irqbalance.service
└─850 /usr/sbin/irqbalance --foreground

Aug 15 10:27:30 localhost.localdomain systemd[1]: Started irqbalance daemon.
Aug 15 10:27:30 localhost.localdomain systemd[1]: Starting irqbalance daemon...
```

步骤3 （可选）开启中断绑定。

开启irqbalance服务让中断自由调度，网络性能提升。如果提升后的网络性能仍不满足您的预期，可以手动设置弹性云服务器操作系统的中断亲和性。

具体操作如下：

执行以下脚本，使得弹性云服务器中每个CPU服务一个多队列的中断请求，一对一的对应设置。即一个队列对应一个中断号，一个中断号绑定一个CPU，一一对应。

```
#!/bin/bash
service irqbalance stop

eth_dirs=$(ls -d /sys/class/net/eth*)
if [ $? -ne 0 ];then
```

```

echo "Failed to find eth* , sleep 30" >> $secs_network_log
sleep 30
eth_dirs=$(ls -d /sys/class/net/eth*)
fi

for eth in $eth_dirs
do
cur_eth=$(basename $eth)
cpu_count=`cat /proc/cpuinfo| grep "processor"| wc -l`
virtio_name=$(ls -l /sys/class/net/"$cur_eth"/device/driver/ | grep pci |awk '{print $9}')

affinity_cpu=0
virtio_input="$virtio_name"-input"
irqs_in=$(grep "$virtio_input" /proc/interrupts | awk -F ":" '{print $1}')
for irq in ${irqs_in[*]}
do
echo $((affinity_cpu%cpu_count)) > /proc/irq/"$irq"/smp_affinity_list
affinity_cpu=$((affinity_cpu+2))
done

affinity_cpu=1
virtio_output="$virtio_name"-output"
irqs_out=$(grep "$virtio_output" /proc/interrupts | awk -F ":" '{print $1}')
for irq in ${irqs_out[*]}
do
echo $((affinity_cpu%cpu_count)) > /proc/irq/"$irq"/smp_affinity_list
affinity_cpu=$((affinity_cpu+2))
done
done
done

```

步骤4 （可选）开启XPS、RPS特性。

XPS主要是针对多队列的网卡发送时的优化，当发送一个数据包的时候，它会根据CPU来选择对应的队列。

```

#!/bin/bash
# enable XPS feature
cpu_count=$(grep -c processor /proc/cpuinfo)
dec2hex(){
echo $(printf "%x" $1)
}
eth_dirs=$(ls -d /sys/class/net/eth*)
if [ $? -ne 0 ];then
echo "Failed to find eth* , sleep 30" >> $secs_network_log
sleep 30
eth_dirs=$(ls -d /sys/class/net/eth*)
fi
for eth in $eth_dirs
do
cpu_id=1
cur_eth=$(basename $eth)
cur_q_num=$(ethtool -l $cur_eth | grep -iA5 current | grep -i combined | awk '{print $2}')
for((i=0;i<cur_q_num;i++))
do
if [ $i -eq $cpu_count ];then
cpu_id=1
fi
xps_file="/sys/class/net/${cur_eth}/queues/tx-$i/xps_cpus"
rps_file="/sys/class/net/${cur_eth}/queues/rx-$i/rps_cpus"
cpuset=$(dec2hex "$cpu_id")
echo $cpuset > $xps_file
echo $cpuset > $rps_file
let cpu_id=cpu_id*2
done
done
done

```

----结束

10.7.8 如何配置 IPv6 地址

操作场景

IPv6的使用，可以有效弥补IPv4网络地址资源有限的问题。如果当前云服务器使用IPv4，那么启用IPv6后，云服务器可在双栈模式下运行，即云服务器可以拥有两个不同版本的IP地址：IPv4地址和IPv6地址，这两个IP地址都可以进行内网/公网访问。

按照[约束与限制](#)中的网络环境要求创建的云服务器，有些不能动态获取到IPv6地址，需要进行相关配置才行。如果云服务器使用的是公共镜像，则支持情况如下：

- Windows公共镜像默认已开启IPv6动态获取功能，无需配置，文中的[Windows 2012操作系统](#)和[Windows 2008操作系统](#)部分供您验证、参考。
- Linux公共镜像开启动态获取IPv6功能时，需要先判断是否支持IPv6协议栈，再判断是否已开启动态获取IPv6。目前，所有Linux公共镜像均已支持IPv6协议栈。

约束与限制

- 请确保云服务器所在的子网已开启IPv6功能。
若云服务器所在子网未开启IPv6功能，需参考[开启云服务器的IPv6功能](#)进行开启，开启后不允许关闭。
- 请确保创建云服务器时已选择“自动分配IPv6地址”。
- 云服务器启动之后动态插拔的网卡不支持IPv6地址动态获取功能。
- 仅弹性云服务器支持IPv6双栈，裸金属服务器不支持。
- 同一个网卡上，只能绑定一个IPv6地址。

操作导航

- Windows系统：本文以Windows 2012版本、Windows 2008版本为例，介绍Windows操作系统启用IPv6的方法，如[表10-6](#)所示。
- Linux系统：本文提供了自动配置、手动配置两种方式启用IPv6，推荐您使用自动配置方法，如[表10-6](#)所示。

对于CentOS 6.x和Debian操作系统，云服务器内部配置IPv6自动获取功能之后，将该云服务器制作为私有镜像，使用该镜像在非IPv6网络环境中创建云服务器时，由于等待获取IPv6地址超时，导致云服务器启动较慢，此时您可以参考[设置云服务器获取IPv6地址超时时间](#)操作。

表 10-6 不同操作系统启用 IPv6 操作指导

操作系统	方式	操作指导
Windows 2012	自动配置启用IPv6	Windows 2012操作系统
Windows 2008	自动配置启用IPv6	Windows 2008操作系统
Linux	自动配置启用IPv6（推荐）	Linux操作系统（自动配置启用IPv6）
Linux	手动配置启用IPv6	Linux操作系统（手动配置启用IPv6）

开启云服务器的 IPv6 功能

开启子网的IPv6网段

📖 说明

云服务器所属子网的IPv6功能开启后会分配IPv6网段，开启后不允许关闭。

1. 登录管理控制台。
2. 选择“计算 > 弹性云服务器”。
3. 单击待开启IPv6功能的弹性云服务器，进入详情页面。
4. 单击“虚拟私有云”名称，进入弹性云服务器所属虚拟私有云列表。
5. 在虚拟私有云列表中，单击“子网个数”列对应的数字超链接。
进入子网列表页面。
6. 在子网列表中，单击待修改的子网名称超链接。
进入子网详情页面。
7. 在子网详情页，单击“开启IPv6”。
8. 单击“是”，完成子网IPv6网段的开启。

开启云服务器网卡的IPv6功能

1. 返回弹性云服务器详情页面。
2. 在“弹性网卡”页签，单击网卡折叠面板右上角的“开启IPv6”。

📖 说明

- 若不再使用弹性云服务器的IPv6功能，可以在当前页面单击“关闭IPv6”，关闭网卡的IPv6功能，关闭后该网卡的“IPv6地址”为空。
 - 关闭IPv6后，如果重新开启弹性云服务器的IPv6，在重启云服务器后，需要登录弹性云服务器手动清理IPv6缓存，并重新请求获取IPv6地址。
3. 单击“是”，完成开启网卡的IPv6功能。

Windows 2012 操作系统

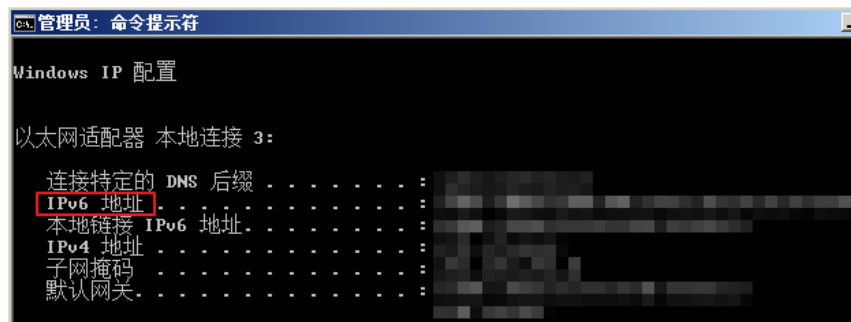
步骤1 检查是否启用IPv6。

打开cmd窗口，执行如下命令，查看当前云服务器是否启用IPv6。

ipconfig

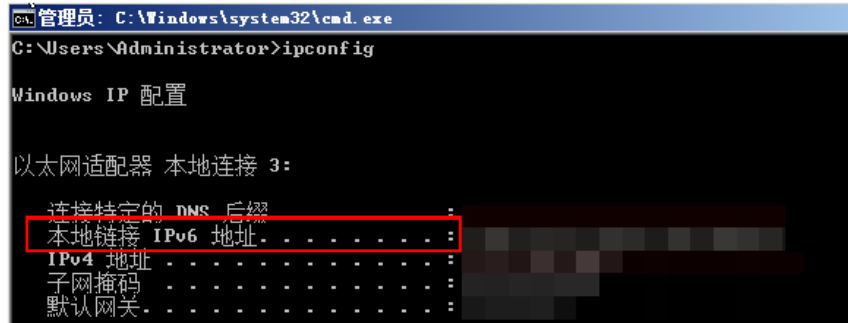
- 如果已启用IPv6，则会显示IPv6的地址。

图 10-10 显示 IPv6 的地址



- 如果显示只有本地链接IPv6地址，则表示无法动态获取到IPv6地址。请执行[步骤2](#)。

图 10-11 本地链接 IPv6 地址



- 如果未启用IPv6，则不会显示IPv6的地址。请执行[步骤3](#)。

图 10-12 未启用 IPv6



说明

Windows公共镜像默认已经配置了IPv6动态获取功能，即回显如[图10-10](#)所示，无需特殊配置。

步骤2 配置动态获取IPv6。

1. 单击“开始 > 控制面板”。
2. 单击“网络和共享中心”。
3. 单击以太网连接。

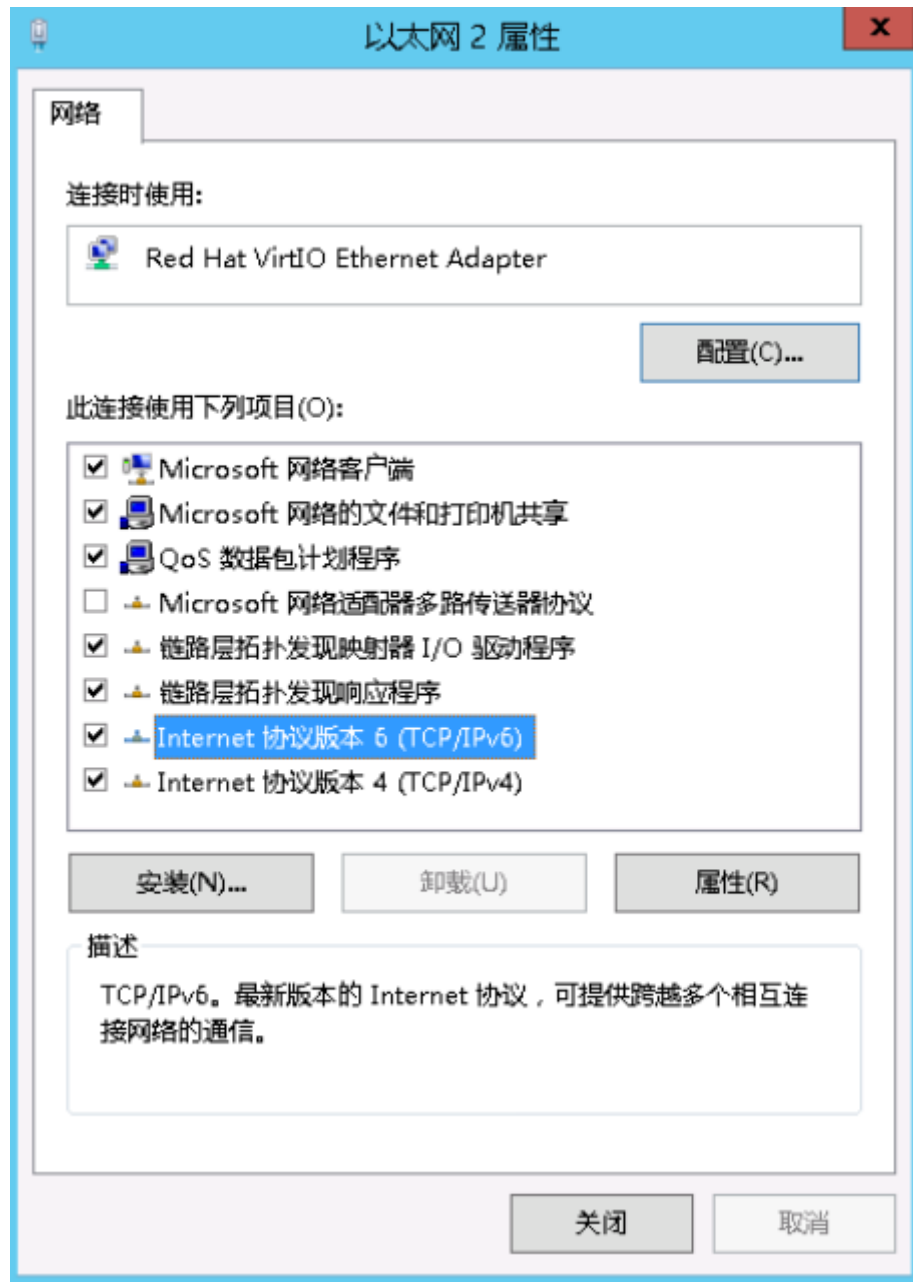
图 10-13 以太网连接



4. 在以太网状态的弹窗中单击左下角的“属性”。

5. 勾选“Internet协议版本 6 (TCP/IPv6)”，然后单击“确定”。

图 10-14 配置动态获取 IPv6

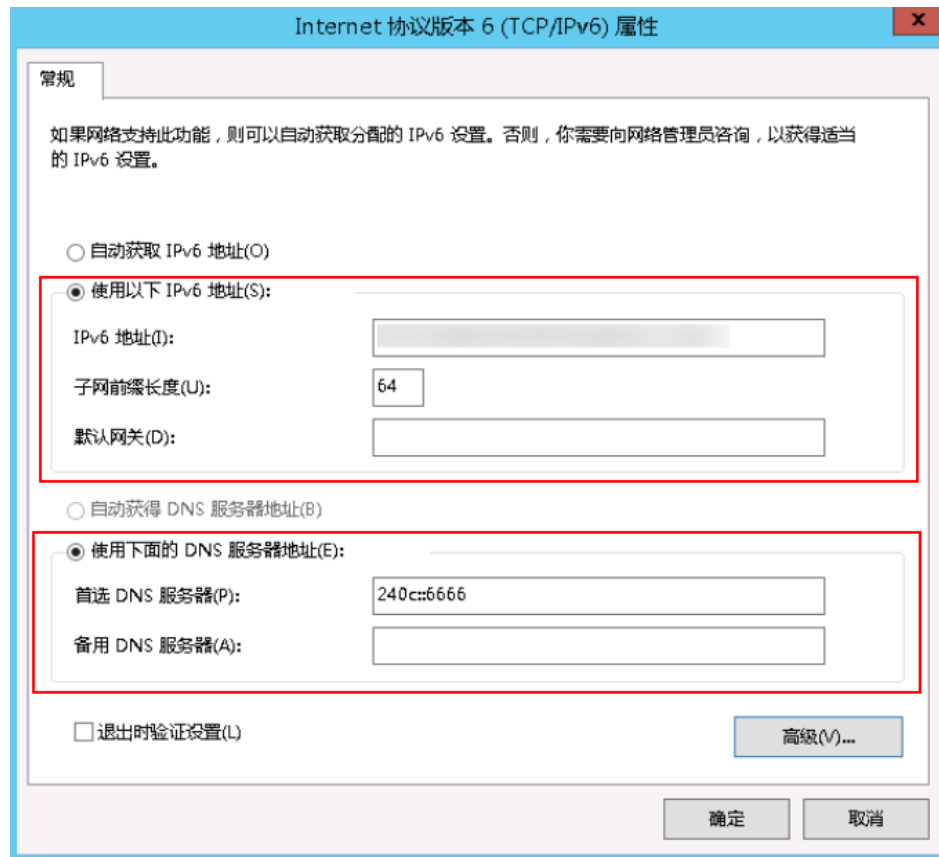


6. 执行**步骤1**检查是否已开启动态获取IPv6。

步骤3 启用和配置IPv6。

1. 在“Internet 协议版本 6 (TCP/IPv6) 属性”弹窗中，配置云服务器的IPv6地址和DNS服务器地址。
 - IPv6地址：创建云服务器时分配的IPv6地址，请从控制台云服务器的列表页面获取。
 - 子网前缀长度：64
 - DNS服务器：推荐使用240c::6666

图 10-15 在控制台获取 IPv6 地址



2. （可选配置）根据操作系统不同请分别执行以下命令。
Windows Server 2012操作系统云服务器请在PowerShell或者cmd中执行如下命令：

Set-NetIPv6Protocol -RandomizeIdentifiers disabled

3. 执行**步骤1**检查是否已开启动态获取IPv6。

----结束

Windows 2008 操作系统

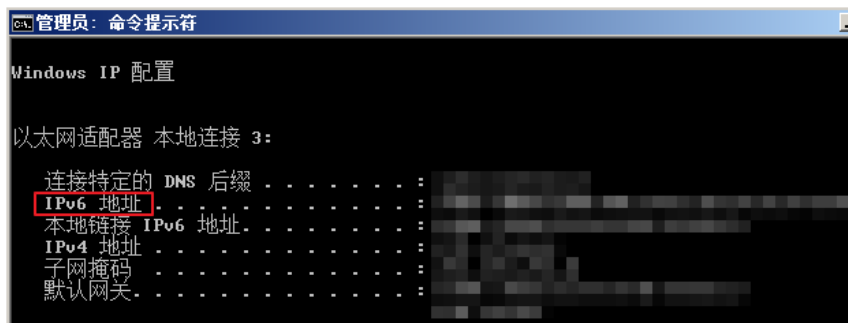
步骤1 检查是否启用IPv6。

打开cmd窗口，执行如下命令，查看当前云服务器是否启用IPv6。

ipconfig

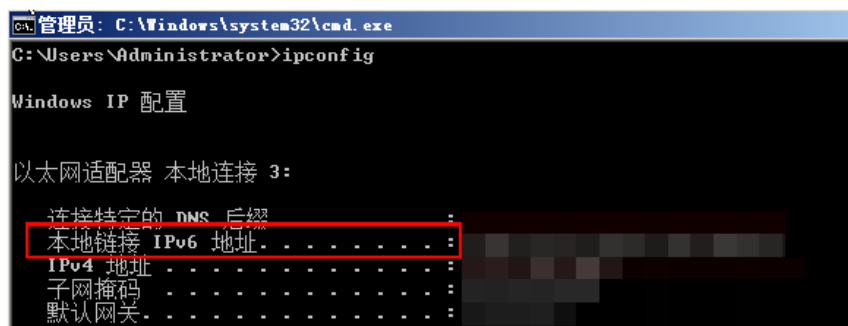
- 如果已启用IPv6，则会显示IPv6的地址。

图 10-16 显示 IPv6 的地址



- 如果显示只有本地链接IPv6地址，则表示无法动态获取到IPv6地址。请执行[步骤2](#)。

图 10-17 本地链接 IPv6 地址



- 如果未启用IPv6，则不会显示IPv6的地址。请执行[步骤3](#)。

图 10-18 未启用 IPv6



📖 说明

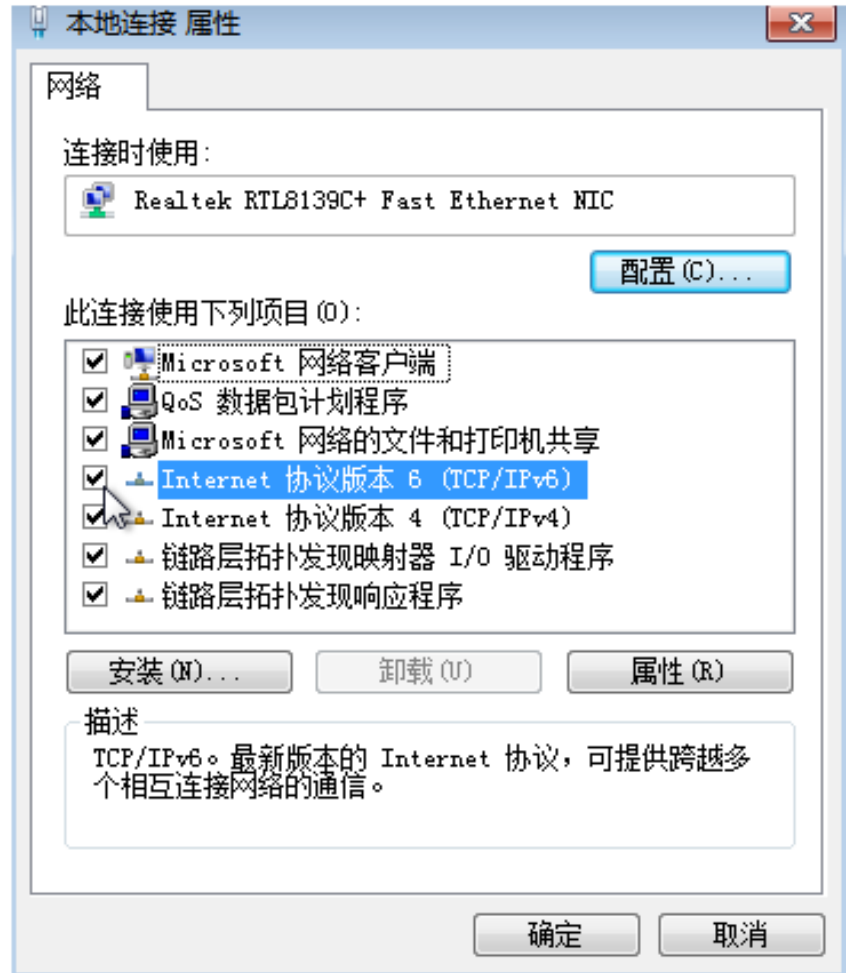
Windows公共镜像默认已经配置了IPv6动态获取功能，即回显如[图10-16](#)所示，无需特殊配置。

步骤2 配置动态获取IPv6。

1. 单击“开始 > 控制面板”。
2. 单击“网络和共享中心”。

3. 左键单击“更改适配器设置”。
4. 右键单击网络连接并选择“属性”。
5. 勾选“Internet协议版本 6 (TCP/IPv6)”，然后单击“确定”。

图 10-19 配置动态获取 IPv6

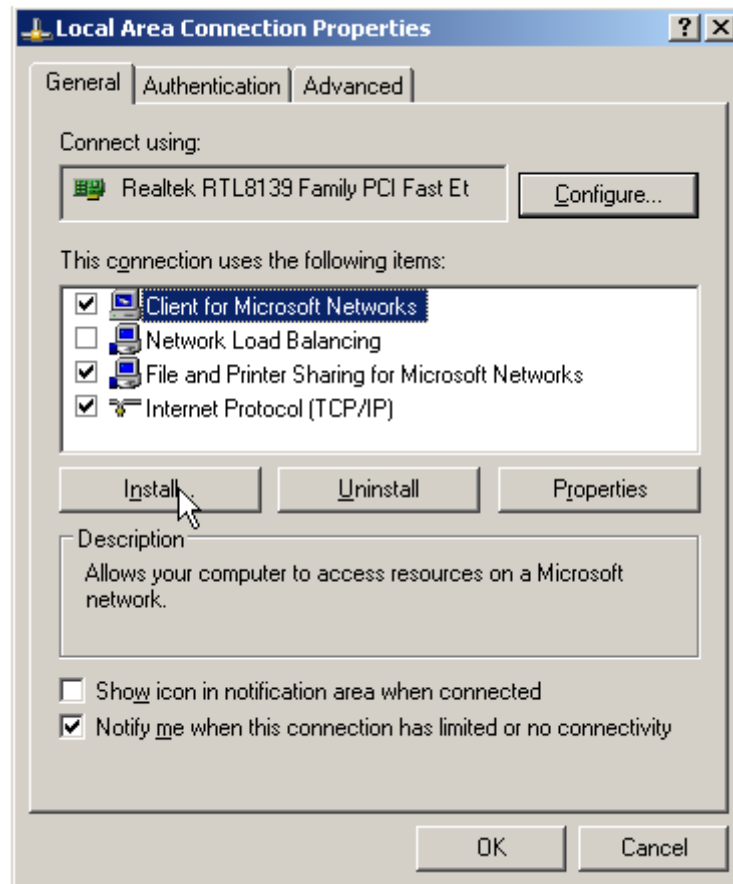


6. 执行**步骤1**检查是否已开启动态获取IPv6。

步骤3 启用和配置IPv6。

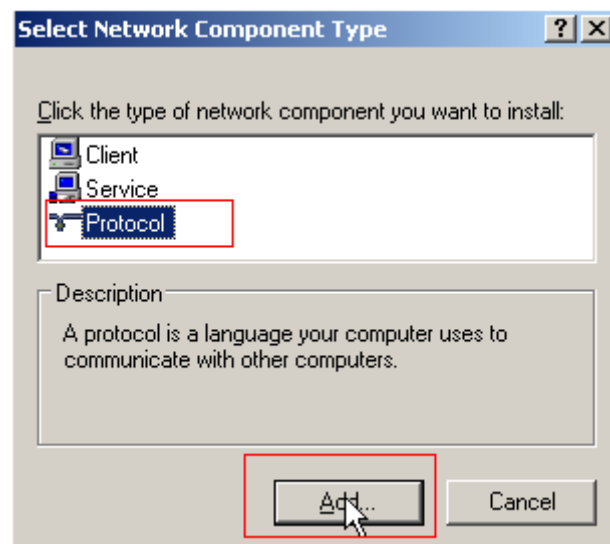
1. 选择“开始 > 控制面板 > 网络连接 > 本地连接”。
2. 选择“属性”，确认勾选以下选项后单击“安装”。

图 10-20 启用和配置 IPv6



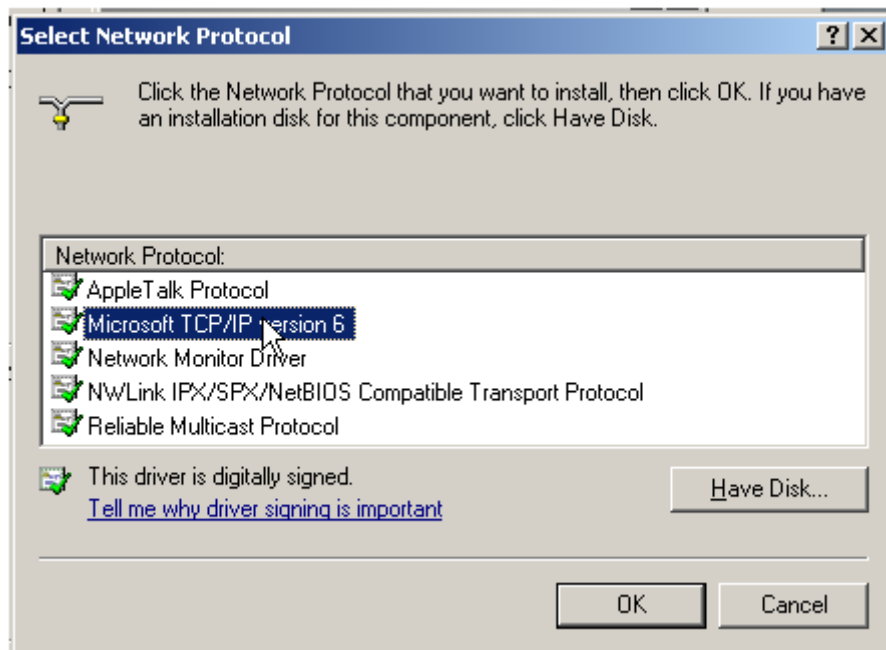
3. 选择“协议”，然后单击“添加”。

图 10-21 添加协议



4. 在网络协议列表中选择“Microsoft TCP/IP版本 6”，然后单击“确定”。

图 10-22 网络协议列表



5. （可选配置）根据操作系统不同请分别执行以下命令。

Windows Server 2008操作系统云服务器请在PowerShell或者cmd中执行如下命令：

netsh interface ipv6 set global randomizeidentifiers=disable

设置云服务器先禁用本地连接，再重启本地连接。

禁用本地连接：单击“开始 > 控制面板 > 网络和共享中心 > 更改适配器配置”，选择本地连接，单击右键选择“禁用”。

重启本地连接：单击“开始 > 控制面板 > 网络和共享中心 > 更改适配器配置”，选择本地连接，单击右键选择“启用”。

6. 执行[步骤1](#)检查是否已开启动态获取IPv6。

----结束

Linux 操作系统（自动配置启用 IPv6）

ipv6-setup-xxx工具能为开启IPv6协议栈的Linux操作系统自动配置动态获取IPv6地址。其中，xxx表示工具系列：rhel或debian。

您也可以参考[Linux操作系统（手动配置启用IPv6）](#)手动配置启用IPv6。

⚠ 注意

- ipv6-setup-xxx工具运行时会自动重启网络服务，导致网络短暂不可用。
- CentOS 6.x和Debian操作系统的云服务器内部配置IPv6自动获取功能之后，将该云服务器制作为私有镜像，使用该镜像在非IPv6网络环境中创建云服务器时，由于等待获取IPv6地址超时，导致云服务器启动较慢，您可以参考[设置云服务器获取IPv6地址超时时间](#)设置获取IPv6地址超时时间为30s，然后再重新制作私有镜像。

步骤1 执行如下命令，查看当前云服务器是否启用IPv6。

ip addr

- 如果没有开启IPv6协议栈，则只能看到IPv4地址，如下图所示，请参考[设置云服务器获取IPv6地址超时时间](#)先开启IPv6协议栈。

图 10-23 云服务器未开启 IPv6 协议栈

```
eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP qlen 1000
link/ether fa:16:3e: bnd ff:ff:ff:ff:ff:ff
inet bnd bnd scope global noprefixroute dynamic eth0
valid_lft 1193sec preferred_lft 1193sec
```

- 如果已开启IPv6协议栈，则可以看到LLA地址（fe80开头）。

图 10-24 云服务器已开启 IPv6 协议栈

```
eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
link/ether fa:16:3e:75:af:4c bnd ff:ff:ff:ff:ff:ff
inet bnd bnd scope global noprefixroute dynamic eth0
valid_lft 76391sec preferred_lft 76391sec
inet6 fe80::f816: /64 scope link
valid_lft forever preferred_lft forever
```

- 如果已开启IPv6协议栈并且已获取到IPv6地址，则会看到如下地址：

图 10-25 云服务器已开启 IPv6 协议栈并且已获取到 IPv6 地址

```
eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
link/ether fa:16:3e:75:af:4c bnd ff:ff:ff:ff:ff:ff
inet bnd bnd scope global noprefixroute dynamic eth0
valid_lft 86395sec preferred_lft 86395sec
inet6 2407:c080:802: /128 scope global dynamic
valid_lft 7496sec preferred_lft 7196sec
inet6 fe80::f816:3eff: /64 scope link noprefixroute
valid_lft forever preferred_lft forever
```

说明

Linux公共镜像均已开启IPv6协议栈，如[图10-24](#)所示；

步骤2 开启Linux云服务器IPv6协议栈。

1. 执行如下命令，确认内核是否支持IPv6协议栈。

sysctl -a | grep ipv6

- 如果有输出信息，表示内核支持IPv6协议栈。
- 如果没有任何输出，说明内核不支持IPv6协议栈，需要执行[2.b](#)加载IPv6模块。

2. 执行以下命令，加载IPv6模块。

modprobe ipv6

3. 修改“/etc/sysctl.conf”配置文件，增加如下配置：

net.ipv6.conf.all.disable_ipv6=0

4. 保存配置并退出，然后执行如下命令，加载配置。

sysctl -p

步骤3 自动配置启用IPv6。

1. 下载对应系统版本的工具ipv6-setup-rhel或ipv6-setup-debian，并上传至待操作的云服务器。

ipv6-setup-xxx工具会添加或者修改网卡设备的配置文件，添加IPv6动态获取的配置信息，然后重启网卡或者网络服务。

ipv6-setup-rhel和ipv6-setup-debian的工具下载地址请联系管理员获取。

2. 执行以下命令，添加执行权限。

```
chmod +x ipv6-setup-xxx
```

3. 执行以下命令，指定一个网卡设备，配置动态获取IPv6地址。

```
./ipv6-setup-xxx --dev [dev]
```

示例：

```
./ipv6-setup-xxx --dev eth0
```

📖 说明

- 如需对所有网卡配置动态获取IPv6地址，命令为./ipv6-setup-xxx，即不带参数。
- 如需查询工具的用法，请执行命令./ipv6-setup-xxx --help。

----结束

Linux 操作系统（手动配置启用 IPv6）

⚠️ 注意

CentOS 6.x和Debian操作系统的云服务器内部配置IPv6自动获取功能之后，将该云服务器制作作为私有镜像，使用该镜像在非IPv6网络环境中创建云服务器时，由于等待获取IPv6地址超时，导致云服务器启动较慢，您可以参考[设置云服务器获取IPv6地址超时时间](#)设置获取IPv6地址超时时间为30s，然后再重新制作私有镜像。

- 步骤1 执行如下命令，查看当前云服务器是否启用IPv6。

ip addr

- 如果没有开启IPv6协议栈，则只能看到IPv4地址，如下图所示，请参考[步骤2](#)先开启IPv6协议栈。

图 10-26 未开启 IPv6 协议栈

```
eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP qlen 1000
link/ether fa:16:3e: b8:00:00:00:00:00 brd ff:ff:ff:ff:ff:ff
inet 192.168.1.100 brd 192.168.1.255 scope global noprefixroute dynamic eth0
    valid_lft 1193sec preferred_lft 1193sec
```

- 如果已开启IPv6协议栈，则可以看到LLA地址（fe80开头）。

图 10-27 已开启 IPv6 协议栈

```
eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
link/ether fa:16:3e:75:af:4c brd ff:ff:ff:ff:ff:ff
inet 192.168.1.100 brd 192.168.1.255 scope global noprefixroute dynamic eth0
    valid_lft 76391sec preferred_lft 76391sec
inet6 fe80::f816:3e75:af4c:1/64 scope link
    valid_lft forever preferred_lft forever
```

- 如果已开启IPv6协议栈并且已获取到IPv6地址，则会看到如下地址：

图 10-28 已开启 IPv6 协议栈并且已获取到 IPv6 地址

```
eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
link/ether fa:16:3e:75:af:4c brd ff:ff:ff:ff:ff:ff
inet 192.168.1.100 brd 192.168.1.255 scope global noprefixroute dynamic eth0
    valid_lft 86395sec preferred_lft 86395sec
inet6 2407:c080:802:1::1/128 scope global dynamic
    valid_lft 7496sec preferred_lft 7196sec
inet6 fe80::f816:3e75:af4c:1/64 scope link noprefixroute
    valid_lft forever preferred_lft forever
```

📖 说明

Linux公共镜像均已开启IPv6协议栈，如图10-27所示；

步骤2 开启Linux云服务器IPv6协议栈。

1. 执行如下命令，确认内核是否支持IPv6协议栈。

```
sysctl -a | grep ipv6
```

- 如果有输出信息，表示内核支持IPv6协议栈。
- 如果没有任何输出，说明内核不支持IPv6协议栈，需要执行步骤2.2加载IPv6模块。

2. 执行以下命令，加载IPv6模块。

```
modprobe ipv6
```

3. 修改“/etc/sysctl.conf”配置文件，增加如下配置：

```
net.ipv6.conf.all.disable_ipv6=0
```

4. 保存配置并退出，然后执行如下命令，加载配置。

```
sysctl -p
```

步骤3 手动配置启用IPv6。操作系统不同，步骤有所差别。

- Ubuntu 18.04/20.04操作系统云服务器配置动态获取IPv6。

- a. 执行以下命令，进入“/etc/netplan/”。

```
cd /etc/netplan
```

- b. 执行以下命令，查询配置文件名。

```
ls
```

图 10-29 查询配置文件名



```
root@ecs-...:/etc/netplan# ls
01-netcfg.yaml 01-network-manager-all.yaml
```

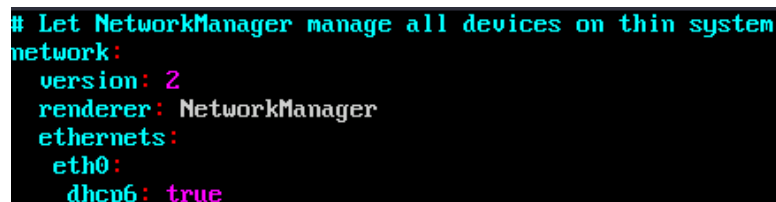
- c. 执行以下命令，编辑“01-network-manager-all.yaml”配置文件。

```
vi 01-network-manager-all.yaml
```

- d. 在“01-network-manager-all.yaml”下增加如下内容，注意yaml文件格式及缩进：

```
ethernets:
  eth0:
    dhcp6: true
```

图 10-30 修改结果



```
# Let NetworkManager manage all devices on this system
network:
  version: 2
  renderer: NetworkManager
  ethernets:
    eth0:
      dhcp6: true
```

修改完成后保存退出。

- e. 执行以下命令，使更改生效。

```
sudo netplan apply
```

- Ubuntu 22.04操作系统云服务器配置动态获取IPv6。

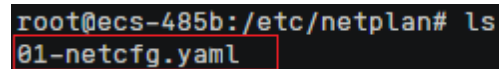
- a. 执行以下命令，进入“/etc/netplan/”。

```
cd /etc/netplan
```

- b. 执行以下命令，查询配置文件名。

```
ls
```

图 10-31 查询配置文件名



```
root@ecs-485b:/etc/netplan# ls
01-netcfg.yaml
```

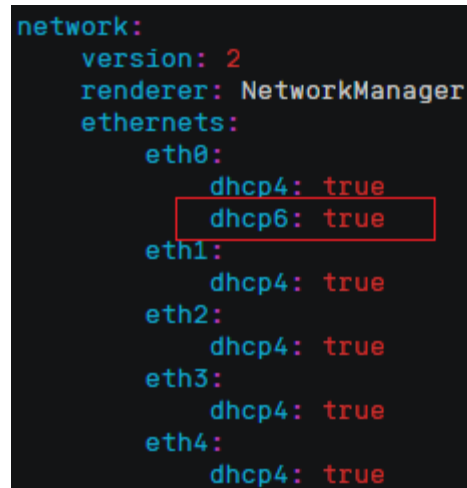
- c. 执行以下命令，编辑“01-netcfg.yaml”配置文件。

```
vi 01-netcfg.yaml
```

- d. 在“01-netcfg.yaml”中增加如下内容，注意yaml文件格式及缩进：

```
ethernets:
  eth0:
    dhcp6: true
```

图 10-32 修改结果



```
network:
  version: 2
  renderer: NetworkManager
  ethernets:
    eth0:
      dhcp4: true
      dhcp6: true
    eth1:
      dhcp4: true
    eth2:
      dhcp4: true
    eth3:
      dhcp4: true
    eth4:
      dhcp4: true
```

修改完成后保存退出。

- e. 执行以下命令，使更改生效。

```
sudo netplan apply
```

- f. 执行以下命令，编辑“/etc/NetworkManager/NetworkManager.conf”文件。

```
vi /etc/NetworkManager/NetworkManager.conf
```

- g. 在“NetworkManager.conf”中增加如下内容，注意文件格式及缩进：

```
[main]
plugins=ifupdown,keyfile
dhcp=dhclient

[ifupdown]
managed=true

[device]
wifi.scan-rand-mac-address=no
```

图 10-33 修改结果

```
[main]
plugins=ifupdown,keyfile
dhcp=dhclient

[ifupdown]
managed=true

[device]
wifi.scan-rand-mac-address=no
```

h. 执行以下命令，使配置生效。

```
systemctl restart NetworkManager
```

- Debian操作系统云服务器配置动态获取IPv6。

a. 编辑“/etc/network/interfaces”文件，使之包含以下内容：

```
auto lo
iface lo inet loopback
auto eth0
iface eth0 inet dhcp
iface eth0 inet6 dhcp
pre-up sleep 3
```

b. 如果有多个网卡，则在“/etc/network/interfaces”文件中，增加对应网卡的配置，以eth1为例，需要增加：

```
auto eth1
iface eth1 inet dhcp
iface eth1 inet6 dhcp
pre-up sleep 3
```

c. 执行如下命令重启网络服务。

```
service networking restart
```

说明

如果将网卡进行down/up操作之后无法获取IPv6地址，也可以通过此命令重启网络服务。

d. 执行步骤[步骤1](#)检查是否已开启动态IPv6。

- CentOS/EulerOS/Fedora操作系统云服务器配置动态获取IPv6。

a. 编辑主网卡配置文件“/etc/sysconfig/network-scripts/ifcfg-eth0”。

补充如下配置项：

```
IPV6INIT=yes
DHCPV6C=yes
```

b. 编辑“/etc/sysconfig/network”，按如下所示添加或修改以下行。

```
NETWORKING_IPV6=yes
```

c. CentOS 6系列从网卡需要编辑对应的配置文件，以eth1为例，编辑“/etc/sysconfig/network-scripts/ifcfg-eth1”。

补充如下配置项：

```
IPV6INIT=yes
DHCPV6C=yes
```

CentOS 6.3系统中默认ip6tables会过滤dhcpv6-client请求，所以CentOS 6.3除了需要编辑“ifcfg-eth*”文件外，还需要额外添加一条允许dhcpv6-client请求的ip6tables规则。操作如下：

i. 执行以下命令，添加ip6tables规则。

```
ip6tables -A INPUT -m state --state NEW -m udp -p udp --dport 546 -d fe80::/64 -j ACCEPT
```

- ii. 执行以下命令，保存ip6tables规则。
service ip6tables save

图 10-34 命令示例

```
root@ecs-cd02 log]# ip6tables -A INPUT -m state --state NEW -m udp -p udp --dport 546 -d fe80::/64 -j ACCEPT
nf_conntrack version 0.5.0 (7964 buckets, 31856 max)
root@ecs-cd02 log]# service ip6tables save
ip6tables: Saving firewall rules to /etc/sysconfig/ip6tablef OK ]
```

- d. (可选配置) CentOS 7/CentOS 8系列需要将扩展网卡的IPv6 LLA地址模式修改为EUI64。

- i. 执行如下命令查看网卡信息。
nmcli con

图 10-35 查看网卡信息

```
[root@ecs-166b ~]# nmcli con
NAME                UUID                                  TYPE      DEVICE
System eth0         5fb06bd0-0bb0-7ffb-45f1-d6edd65f3e03 ethernet  eth0
Wired connection 1  9c92fad9-6ecb-3e6c-eb4d-8a47c6f50c04 ethernet  eth1
Wired connection 1  3a73717e-65ab-93e8-b518-24f5af32dc0d ethernet  eth2
```

- ii. 将eth1的IPv6 LLA地址模式按以下命令修改为EUI64:
nmcli con modify "Wired connection 1" ipv6.addr-gen-mode eui64

说明

CentOS不同系列，网卡信息存在差异，命令中的“Wired connection 1”需要根据实际查询的网卡信息的“NAME”列进行替换。

- iii. 通过ifconfig命令将eth1进行down/up操作。

ifdown eth1
ifup eth1

- e. 重启网络服务。
 - i. CentOS 6系列执行以下命令，重启网络服务。
service network restart
 - ii. CentOS 7/EulerOS/Fedora系列执行以下命令，重启网络服务。
systemctl restart NetworkManager

- f. 执行步骤**步骤1**检查是否已开启动态IPv6。

- SUSE/openSUSE/CoreOS操作系统云服务器配置动态获取IPv6。
SUSE 11 SP4不支持IPv6自动获取。
SUSE 12 SP1、SUSE 12 SP2无需特殊配置。
openSUSE 13.2、openSUSE 42.2无需特殊配置。
CoreOS 10.10.5无需特殊配置。

----结束

设置云服务器获取 IPv6 地址超时时间

CentOS 6.x和Debian操作系统的云服务器内部配置IPv6自动获取功能之后，将该云服务器制作为私有镜像，使用该镜像在非IPv6网络环境中创建云服务器时，由于等待获取IPv6地址超时，导致云服务器启动较慢，您可以参考本节操作设置获取IPv6地址超时时间为30s，然后再重新制作私有镜像。

- CentOS 6.x:
 - a. 执行以下命令编辑“dhclient.conf”文件。
vi /etc/dhcp/dhclient.conf
 - b. 按“i”进入编辑模式，在文件中增加timeout属性。
timeout 30;
 - c. 输入:wq保存后退出。
- Debian 7.5:
 - a. 执行以下命令编辑“networking”文件。
vi /etc/init.d/networking
 - b. 按“i”进入编辑模式，增加延迟命令timeout，修改点如下图所示。

图 10-36 修改点 1

```

115 case "$1" in
116 start)
117     if init_is_upstart; then
118         exit 1
119     fi
120     process_options
121     check_ifstate
122
123     if [ "$CONFIGURE_INTERFACES" = no ]
124     then
125         log_action_msg "Not configuring network interfaces, see /etc/default/networking"
126         exit 0
127     fi
128     set -f
129     exclusions=$(process_exclusions)
130     log_action_begin_msg "Configuring network interfaces"
131     if /usr/bin/timeout 30 ifup -a $exclusions $verbose && ifup_hotplug $exclusions $verbose
132     then
133         log_action_end_msg $?
134     else
135         log_action_end_msg $?
136     fi
137     ;;
138
139 stop)
140     if init_is_upstart; then
141         exit 0
142     fi
143     check_network_file_systems
144     check_network_swap
145
146     log_action_begin_msg "Deconfiguring network interfaces"
147     if /usr/bin/timeout 30 ifdown -a --exclude=lo $verbose; then
148         log_action_end_msg $?

```

图 10-37 修改点 2

```

154 reload)
155     process_options
156
157     log_action_begin_msg "Reloading network interfaces configuration"
158     state=$(cat /run/network/ifstate)
159     if /usr/bin/timeout 30 ifdown -a --exclude=lo $verbose || true
160     if /usr/bin/timeout 30 ifup --exclude=lo $state $verbose ; then
161         log_action_end_msg $?
162     else
163         log_action_end_msg $?
164     fi
165     ;;
166
167 force-reload|restart)
168     if init_is_upstart; then
169         exit 1
170     fi
171     process_options
172
173     log_warning_msg "Running $0 $1 is deprecated because it may not re-enable some interfaces"
174     log_action_begin_msg "Reconfiguring network interfaces"
175     if /usr/bin/timeout 30 ifdown -a --exclude=lo $verbose || true
176     set -f
177     exclusions=$(process_exclusions)
178     if /usr/bin/timeout 30 ifup -a --exclude=lo $exclusions $verbose && ifup_hotplug $exclusions $verbose
179     then
180         log_action_end_msg $?
181     else
182         log_action_end_msg $?

```

- Debian 8.2.0/8.8.0
 - a. 执行以下命令编辑“network-pre.conf”文件。
vi /lib/systemd/system/networking.service.d/network-pre.conf

- b. 按“i”进入编辑模式，在文件中增加timeout属性。

```
[Service]
TimeoutStartSec=30
```
- Debian 9.0
 - a. 执行以下命令编辑“networking.service”文件。

```
vi /etc/systemd/system/network-online.target.wants/networking.service
```
 - b. 按“i”进入编辑模式，将TimeoutStartSec=5min改为TimeoutStartSec=30。

10.7.9 如何优化系统盘镜像使其支持实例快速发放？

操作场景

系统盘镜像支持快速发放特性，可有效缩短云服务器实例的下发时间。存量系统盘镜像可能不支持快速发放，建议您使用镜像复制功能进行优化。

假设A镜像不支持快速发放，复制为copy_A镜像后，就可以使用copy_A镜像快速发放云服务器实例了。

约束限制

整机镜像和ISO镜像暂不支持使用该方法优化。

查看镜像是否支持快速发放

1. 登录IMS控制台。
 - a. 登录管理控制台。
 - b. 选择“计算 > 镜像服务”。
进入镜像服务页面。
2. 单击“私有镜像”页签，进入私有镜像列表。
3. 单击待查看的镜像名称，进入镜像详情页面。
4. 查看“是否支持快速发放”的值，即可确定当前镜像是否支持快速发放。

优化方法

1. 在待优化的系统盘镜像所在行，单击“更多 > 复制”。
弹出“复制镜像”对话框。
2. 按照[复制镜像](#)的指导填写参数，无特殊要求。
3. 待镜像复制成功后，新的镜像就支持快速发放功能了，优化完成。

10.7.10 为什么 Windows 云服务器安装 Guest OS driver 会失败？

Guest OS driver安装失败，主要是由于以下几种原因：

- 如果您的镜像文件是从VMware虚拟机导出，可能是由于没有卸载VMware Tools或者VMware Tools没有卸载干净导致Guest OS driver安装失败。
- 请您务必选择正确的Windows版本的Guest OS driver下载，否则可能导致Guest OS driver安装失败。

- 由于安装Guest OS driver需要消耗的一定的空间，磁盘空间预留不足可能导致Guest OS driver安装失败。因此，请您预留约300MB的空间以确保Guest OS driver正确安装。

10.7.11 怎样安装原生的 XEN 和 KVM 驱动

操作场景

对于XEN虚拟化类型的弹性云服务器，在优化Linux私有镜像过程中，需要安装原生的XEN和KVM驱动。

本节介绍安装原生XEN和KVM驱动的具体方法。

注意

对于XEN实例：如果不安装XEN驱动，弹性云服务器的网络性能很差，并且安全组和防火墙也不会生效；

对于KVM实例：如果不安装KVM驱动，弹性云服务器的网卡可能无法检测到，无法与外部通信。因此，请您务必安装XEN和KVM驱动。

前提条件

- 请先确认您当前的云服务器为XEN构架。
- 对于使用Linux系统原生的XEN和KVM驱动的Linux云服务器，其内核版本必须高于2.6.24。
- 建议您禁用任何防病毒软件或入侵检测软件，安装原生的XEN和KVM驱动完成后，您可以再次启用这些软件。

操作步骤

请根据操作系统版本，修改不同的配置文件：

- CentOS/EulerOS系列操作系统

以CentOS 7.0为例，请修改“/etc/dracut.conf”文件，在add_drivers项中添加xen-pv以及virtio的驱动（xen-pv驱动：xen-blkfront、xen-netfront；virtio驱动：virtio_blk、virtio_scsi、virtio_net、virtio_pci、virtio_ring、virtio）。驱动名之间以空格隔开，保存并退出“/etc/dracut.conf”文件，执行**dracut -f**命令，重新生成initrd。

操作方法可参见[CentOS/EulerOS系列操作系统相关操作](#)。

- Ubuntu/Debian系列系统

请修改“/etc/initramfs-tools/modules”文件，添加xen-pv以及virtio的驱动（xen-pv驱动：xen-blkfront、xen-netfront；virtio驱动：virtio_blk、virtio_scsi、virtio_net、virtio_pci、virtio_ring、virtio）。驱动名之间以空格隔开，保存并退出“/etc/initramfs-tools/modules”文件，执行**update-initramfs -u**命令，重新生成initrd。

操作方法可参见[Ubuntu/Debian系列操作系统相关操作](#)。

- SUSE和openSUSE系列系统，根据操作系统版本不同，修改不同的配置文件。

- 当操作系统版本低于SUSE 12 SP1或低于openSUSE 13时，请修改“/etc/sysconfig/kernel”文件，在INITRD_MODULES=""添加xen-pv以及virtio的驱

- 动（xen-pv驱动：xen_vnif、xen_vbd、xen_platform_pci；virtio驱动：virtio_blk、virtio_scsi、virtio_net、virtio_pci、virtio_ring、virtio）。驱动名之间以空格隔开，执行**mkinitrd**命令，重新生成initrd。
- 当操作系统版本为SUSE 12 SP1时，修改“/etc/dracut.conf”文件，在add_drivers项中添加xen-pv以及virtio的驱动（xen-pv驱动：xen_vnif、xen_vbd、xen_platform_pci；virtio驱动：virtio_blk、virtio_scsi、virtio_net、virtio_pci、virtio_ring、virtio）。驱动名之间以空格隔开，执行命令**dracut -f**，重新生成initrd。
 - 当操作系统版本高于SUSE 12 SP1或高于openSUSE 13版本时，修改“/etc/dracut.conf”文件，在add_drivers项中添加xen-pv和virtio的驱动（xen-pv驱动：xen-blkfront、xen-netfront；virtio驱动：virtio_blk、virtio_scsi、virtio_net、virtio_pci、virtio_ring、virtio）。驱动名之间以空格隔开，保存并退出“/etc/dracut.conf”文件，执行**dracut -f**命令，重新生成initrd。

操作方法可参[SUSE/openSUSE系列操作系统相关操作](#)。

📖 说明

SUSE系列操作系统首先应确认OS是否已经安装了xen-kmp包（xen-pv的驱动包），执行以下命令：

```
rpm -qa |grep xen-kmp
```

回显类似如下：

```
xen-kmp-default-4.2.2_04_3.0.76_0.11-0.7.5
```

如果没有安装xen-kmp的包，请到ISO装机文件中获取并安装。

如果误将built-in形式的驱动添加到initrd或initramfs文件中，不会影响云服务器正常使用。

CentOS/EulerOS 系列操作系统相关操作

1. 执行以下命令，打开“/etc/dracut.conf”文件。

```
vi /etc/dracut.conf
```

2. 按“i”进入编辑模式，在“add_drivers”项中添加xen-pv和virtio的驱动（具体格式要根据操作系统本身的要求来决定）。

```
[root@CTU10000xxxx ~]# vi /etc/dracut.conf
# additional kernel modules to the default
add_drivers+="xen-blkfront xen-netfront virtio_blk virtio_scsi virtio_net virtio_pci virtio_ring
virtio"
.....
```

3. 按“Esc”后，输入:wq，按“Enter”。保存设置并退出“/etc/dracut.conf”文件。
4. 执行以下命令，重新生成initrd。

```
dracut -f /boot/initramfs-2.6.32-573.8.1.el6.x86_64.img
```

如果引导的虚拟文件系统不是默认的initramfs，则命令为：**dracut -f 实际使用的initramfs文件名或者initrd文件名**。“实际使用的initramfs文件名或者initrd文件名”可在grub.cfg配置（“/boot/grub/grub.cfg”或“/boot/grub2/grub.cfg”或“/boot/grub/grub.conf”，具体路径根据OS不同会有所区别）中获取。

5. 如果引导的虚拟文件系统是initramfs，执行以下命令，检查是否已经成功装载了原生的XEN和KVM驱动相应模块。

```
lsinitrd /boot/initramfs-`uname -r`.img | grep xen
```

```
lsinitrd /boot/initramfs-`uname -r`.img | grep virtio
```

如果引导的虚拟文件系统是initrd，执行如下命令，检查是否已经成功装载了原生的XEN和KVM驱动相应模块。

```
lsinitrd /boot/initrd-`uname -r` | grep xen
```

```
lsinitrd /boot/initrd-`uname -r` | grep virtio
```

以引导的虚拟文件系统是initramfs为例，回显信息如下所示：

```
[root@CTU10000xxxxx home]# lsinitrd /boot/initramfs-`uname -r`.img | grep xen
-rwxr--r-- 1 root root 54888 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/drivers/
block/xen-blkfront.ko
-rwxr--r-- 1 root root 45664 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/
drivers/net/xen-netfront.ko

[root@CTU10000xxxxx home]# lsinitrd /boot/initramfs-`uname -r`.img | grep virtio
-rwxr--r-- 1 root root 23448 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/drivers/
block/virtio_blk.ko
-rwxr--r-- 1 root root 50704 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/
drivers/net/virtio_net.ko
-rwxr--r-- 1 root root 28424 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/drivers/
scsi/virtio_scsi.ko
drwxr-xr-x 2 root root 0 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/drivers/
virtio
-rwxr--r-- 1 root root 14544 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/drivers/
virtio/virtio.ko
-rwxr--r-- 1 root root 21040 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/drivers/
virtio/virtio_pci.ko
-rwxr--r-- 1 root root 18016 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/drivers/
virtio/virtio_ring.ko
```

📖 说明

如果误将built-in形式存在内核中的驱动添加到initrd或initramfs文件中，不会影响云服务器器正常使用，这里全写进去只是为了修改的方便，但是使用lsinitrd命令无法检查到。可使用如下方法确定这些驱动是否以built-in形式存在内核中，例如：

```
cat /boot/config-`uname -r` | grep CONFIG_VIRTIO | grep y
```

```
cat /boot/config-`uname -r` | grep CONFIG_XEN | grep y
```

Ubuntu/Debian 系列操作系统相关操作

1. 执行以下命令，打开“modules”文件。

```
vi /etc/initramfs-tools/modules
```

2. 按“i”进入编辑模式，修改“/etc/initramfs-tools/modules”文件，添加xen-pv以及virtio的驱动（具体格式要根据操作系统本身的要求来决定）。

```
[root@CTU10000xxxxx ~]#vi /etc/initramfs-tools/modules
.....
# Examples:
#
# raid1
# sd_mOd
xen-blkfront
xen-netfront
virtio_blk
virtio_scsi
virtio_net
virtio_pci
virtio_ring
virtio
```

3. 按“Esc”后，输入:wq，按“Enter”。保存设置并退出“/etc/initramfs-tools/modules”文件。
4. 执行以下命令，重新生成initrd。

```
update-initramfs -u
```

5. 执行以下命令，检查是否已经成功装载了原生的XEN和KVM驱动相应模块。

```
lsinitramfs /boot/initrd.img-`uname -r` |grep xen
```

lsinitramfs /boot/initrd.img-`uname -r` |grep virtio

```
[root@ CTU10000xxxxx home]# lsinitramfs /boot/initrd.img-`uname -r` |grep xen
lib/modules/3.5.0-23-generic/kernel/drivers/net/ethernet/qlogic/netxen
lib/modules/3.5.0-23-generic/kernel/drivers/net/ethernet/qlogic/netxen/netxen_nic.ko
lib/modules/3.5.0-23-generic/kernel/drivers/net/xen-netback
lib/modules/3.5.0-23-generic/kernel/drivers/net/xen-netback/xen-netback.ko
lib/modules/3.5.0-23-generic/kernel/drivers/block/xen-blkback
lib/modules/3.5.0-23-generic/kernel/drivers/block/xen-blkback/xen-blkback.ko

[root@ CTU10000xxxxx home]# lsinitramfs /boot/initrd.img-`uname -r` |grep virtio
lib/modules/3.5.0-23-generic/kernel/drivers/scsi/virtio_scsi.ko
```

📖 说明

如果误将built-in形式存在内核中的驱动添加到initrd或initramfs文件中，不会影响云服务器器正常使用，这里全写进去只是为了修改方便，但是使用lsinitrd命令无法检查到。可使用如下方法确定这些驱动是否以built-in形式存在内核中，例如：

```
[root@ CTU10000xxxxx home]# cat /boot/config-`uname -r` | grep CONFIG_VIRTIO | grep y
CONFIG_VIRTIO_BLK=y
CONFIG_VIRTIO_NET=y
CONFIG_VIRTIO=y
CONFIG_VIRTIO_RING=y
CONFIG_VIRTIO_PCI=y
CONFIG_VIRTIO_MMIO_CMDLINE_DEVICES=y
[root@ CTU10000xxxxx home]# cat /boot/config-`uname -r` | grep CONFIG_XEN | grep y
CONFIG_XEN_BLKDEV_FRONTEND=y
CONFIG_XEN_NETDEV_FRONTEND=y
```

SUSE/openSUSE 系列操作系统相关操作

当操作系统版本低于SUSE 12 SP1或低于openSUSE 13时，请修改“/etc/sysconfig/kernel”文件，操作步骤请参考[第一种情况](#)。

当操作系统版本为SUSE 12 SP1时，修改“/etc/dracut.conf”文件，添加xen-pv以及virtio的驱动，操作步骤请参考[第二种情况](#)。

当操作系统版本高于SUSE 12 SP1或高于openSUSE 13版本时，修改“/etc/dracut.conf”文件，在add_drivers项中添加xen-pv和virtio的驱动，操作步骤请参考[第三种情况](#)。

- 当操作系统版本低于SUSE 12 SP1或低于openSUSE 13时，操作步骤如下。

📖 说明

SUSE系列操作系统首先应确认OS是否已经安装了xen-kmp包（xen-pv的驱动包），执行如下命令：

```
rpm -qa |grep xen-kmp
```

回显类似如下：

```
xen-kmp-default-4.2.2_04_3.0.76_0.11-0.7.5
```

如果没有安装xen-kmp的包，请到安装ISO中获取并安装。

- a. 执行如下命令，修改“/etc/sysconfig/kernel”文件。

```
vi /etc/sysconfig/kernel
```

- b. 在INITRD_MODULES=""中添加xen-pv以及virtio的驱动（具体格式要根据OS本身的要求来决定）。

```
SIA10000xxxxx:~ # vi /etc/sysconfig/kernel
# (like drivers for scsi-controllers, for lvm or reiserfs)
#
INITRD_MODULES="ata_piix ata_generic xen_vnif xen_vbd xen_platform_pci virtio_blk
virtio_scsi virtio_net virtio_pci virtio_ring virtio"
```

- c. 执行mkinitrd命令，重新生成initrd。

📖 说明

如果引导的虚拟文件系统不是默认的initramfs或者initrd，则命令为：**dracut -f** 实际使用的initramfs或者initrd文件名。“实际使用的initramfs或者initrd文件名”可在menu.lst或者grub.cfg配置（“/boot/grub/menu.lst”或“/boot/grub/grub.cfg”或“/boot/grub2/grub.cfg”）中获取。

以SUSE 11 SP4为例，如下所示：

```
default 0
timeout 10
gfxmenu (hd0,0)/boot/message
title sles11sp4_001_[_VMX_]
root (hd0,0)
kernel /boot/linux.vmx vga=0x314 splash=silent console=ttyS0,115200n8 console=tty0
net.ifnames=0 NON_PERSISTENT_DEVICE_NAMES=1 showopts
initrd /boot/initrd.vmx
title Failsafe_sles11sp4_001_[_VMX_]
root (hd0,0)
kernel /boot/linux.vmx vga=0x314 splash=silent ide=nodma apm=off noresume edd=off
powersaved=off nohz=off highres=off processor.max+cstate=1 nomodeset x11failsafe
console=ttyS0,115200n8 console=tty0 net.ifnames=0 NON_PERSISTENT_DEVICE_NAMES=1
showopts
initrd /boot/initrd.vmx
```

其中，initrd所在行的/boot/initrd.vmx为实际使用的initrd文件，执行的时候请按照**dracut -f /boot/initrd.vmx**执行。如果initrd所在行的initrd文件不包含/boot目录，如/initramfs-xxx，请在执行dracut命令时增加boot目录，例如：**dracut -f /boot/initramfs-xxx**。

- d. 执行如下命令，检查是否已经成功装载了XEN的PVOPS或者KVM的virtio相应模块。

```
lsinitrd /boot/initrd-`uname -r` | grep xen
```

```
lsinitrd /boot/initrd-`uname -r` | grep virtio
```

```
SIA10000xxxx:~ # lsinitrd /boot/initrd-`uname -r` | grep xen
-rwxr--r-- 1 root root 42400 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/block/
xen-blkfront.ko
-rwxr--r-- 1 root root 44200 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/net/
xen-netfront.ko

SIA10000xxxx:~ # lsinitrd /boot/initrd-`uname -r` | grep virtio
-rwxr--r-- 1 root root 19248 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/scsi/
virtio_scsi.ko
-rwxr--r-- 1 root root 23856 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/block/
virtio_blk.ko
drwxr-xr-x 2 root root 0 Jul 12 14:53 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio
-rwxr--r-- 1 root root 15848 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio/
virtio_ring.ko
-rwxr--r-- 1 root root 20008 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio/
virtio_pci.ko
-rwxr--r-- 1 root root 12272 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio/
virtio.ko
-rwxr--r-- 1 root root 38208 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/net/
virtio_net.ko
```

- e. 重启云服务器。

- f. 重启完毕后，修改“/boot/grub/menu.lst”文件，增加xen_platform_pci.dev_unplug=all和修改root的配置。

修改前如下所示：

```
###Don't change this comment -YaST2 identifier: Original name: linux###
title SUSE Linux Enterprise Server 11SP4 - 3.0.76-0.11 (default)
root (hd0,0)
kernel /boot/vmlinuz-3.0.76-0.11-default root=UUID=4eb40294-4c6f-4384-bbb6-
```



```
b8795bbb1130 splash=silentcrashkernel=256M-:128M showopts vga=0x314
initrd /boot/initrd-3.0.76-0.11-default
```

修改后如下所示：

```
###Don't change this comment -YaST2 identifier: Original name: linux###
title SUSE Linux Enterprise Server 11SP4 - 3.0.76-0.11 (default)
root (hd0,0)
kernel /boot/vmlinuz-3.0.76-0.11-default root=UUID=4eb40294-4c6f-4384-bbb6-
b8795bbb1130 splash=silentcrashkernel=256M-:128M showopts vga=0x314
xen_platform_pci.dev_unplug=all
initrd /boot/initrd-3.0.76-0.11-default
```

📖 说明

- 确保磁盘root分区为UUID的表示形式。
- xen_platform_pci.dev_unplug=all该参数的添加是为了屏蔽qemu设备。
- SUSE 11 SP1 64bit ~ SUSE 11 SP4 64bit系统需要在“menu.lst”文件添加 xen_platform_pci.dev_unplug=all，SUSE 12以后版本默认启用此功能，无需配置。

g. 执行如下命令确认initrd中是否存在XEN驱动。

```
lsinitrd /boot/initrd-`uname -r` | grep xen
```

```
lsinitrd /boot/initrd-`uname -r` | grep virtio
```

```
SIA10000xxxx:~ # lsinitrd /boot/initrd-`uname -r` | grep xen
-rwxr--r-- 1 root root 42400 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/block/
xen-blkfront.ko
-rwxr--r-- 1 root root 44200 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/net/
xen-netfront.ko

SIA10000xxxx:~ # lsinitrd /boot/initrd-`uname -r` | grep virtio
-rwxr--r-- 1 root root 19248 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/scsi/
virtio_scsi.ko
-rwxr--r-- 1 root root 23856 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/block/
virtio_blk.ko
drwxr-xr-x 2 root root 0 Jul 12 14:53 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio
-rwxr--r-- 1 root root 15848 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio/
virtio_ring.ko
-rwxr--r-- 1 root root 20008 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio/
virtio_pci.ko
-rwxr--r-- 1 root root 12272 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio/
virtio.ko
-rwxr--r-- 1 root root 38208 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/net/
virtio_net.ko
```

📖 说明

如果误将built-in形式存在内核中的驱动添加到initrd或initramfs文件中，不会影响云服务器正常使用，这里全写进去只是为了修改方便，但是使用lsinitrd命令无法检查到。可使用如下方法确定这些驱动是否以built-in形式存在内核中，例如：

```
cat /boot/config-`uname -r` | grep CONFIG_VIRTIO | grep y
```

```
cat /boot/config-`uname -r` | grep CONFIG_XEN | grep y
```

- 当操作系统版本为SUSE 12 SP1时，操作步骤如下。

a. 执行以下命令，打开“/etc/dracut.conf”文件。

```
vi /etc/dracut.conf
```

b. 按“i”进入编辑模式，在“add-drivers”项中添加xen-pv和virtio的驱动（具体格式要根据操作系统本身的要求来决定）。

```
[root@CTU10000xxxx ~]# vi /etc/dracut.conf
# additional kernel modules to the default
add_drivers+="ata_piix ata_generic xen_vnif xen_vbd xen_platform_pci virtio_blk virtio_scsi
virtio_net virtio_pci virtio_ring virtio"
```

- c. 按“Esc”后，输入:**wq**，按“Enter”。保存设置并退出“/etc/dracut.conf”文件。
- d. 执行以下命令，重新生成initrd。

dracut -f /boot/initramfs-文件名

如果引导的虚拟文件系统不是默认的initramfs，则命令为：**dracut -f 实际使用的initramfs文件名或者initrd文件名**。“实际使用的initramfs文件名或者initrd文件名”可在grub.cfg配置（“/boot/grub/grub.cfg”或“/boot/grub2/grub.cfg”或“/boot/grub/grub.conf”，具体路径根据OS不同会有所不同）中获取。

- e. 如果引导的虚拟文件系统是initramfs，执行以下命令，检查是否已经成功装载了原生的XEN和KVM驱动相应模块。

lsinitrd /boot/initramfs-`uname -r`.img | grep xen

lsinitrd /boot/initramfs-`uname -r`.img | grep virtio

如果引导的虚拟文件系统是initrd，执行如下命令，检查是否已经成功装载了原生的XEN和KVM驱动相应模块。

lsinitrd /boot/initrd-`uname -r` | grep xen

lsinitrd /boot/initrd-`uname -r` | grep virtio

- 当操作系统版本高于SUSE 12 SP1或高于openSUSE 13版本时，操作步骤如下。以SUSE Linux Enterprise Server 12 SP2 (x86_64)为例，如下所示：

- a. 执行以下命令，打开“/etc/dracut.conf”文件。

vi /etc/dracut.conf

- b. 按“i”进入编辑模式，在“add_drivers”项中添加xen-pv和virtio的驱动（具体格式要根据操作系统本身的要求来决定）。

```
[root@CTU10000xxxx ~]# vi /etc/dracut.conf
# additional kernel modules to the default
add_drivers+="ata_piix ata_generic xen-blkfront xen-netfront virtio_blk virtio_scsi
virtio_net virtio_pci virtio_ring virtio"
```

- c. 按“Esc”后，输入:**wq**，按“Enter”。保存设置并退出“/etc/dracut.conf”文件。
- d. 执行以下命令，重新生成initrd。

dracut -f /boot/initramfs-文件名

如果引导的虚拟文件系统不是默认的initramfs，则命令为：**dracut -f 实际使用的initramfs文件名或者initrd文件名**。“实际使用的initramfs文件名或者initrd文件名”可在grub.cfg配置（“/boot/grub/grub.cfg”或“/boot/grub2/grub.cfg”或“/boot/grub/grub.conf”，具体路径根据OS不同会有所区别）中获取。

- e. 如果引导的虚拟文件系统是initramfs，执行以下命令，检查是否已经成功装载了原生的XEN和KVM驱动相应模块。

lsinitrd /boot/initramfs-`uname -r`.img | grep xen

lsinitrd /boot/initramfs-`uname -r`.img | grep virtio

如果引导的虚拟文件系统是initrd，执行如下命令，检查是否已经成功装载了原生的XEN和KVM驱动相应模块。

lsinitrd /boot/initrd-`uname -r` | grep xen

lsinitrd /boot/initrd-`uname -r` | grep virtio

以引导的虚拟文件系统是initrd为例，回显信息如下所示：

```
sluo-ecs-30dc:~ # lsinitrd /boot/initrd-`uname -r` | grep xen
-rw-r--r-- 1 root root 69575 Oct 26 2016 lib/modules/4.4.21-69-default/kernel/drivers/block/xen-
```



```

blkfront.ko
-rw-r--r-- 1 root root 53415 Oct 26 2016 lib/modules/4.4.21-69-default/kernel/drivers/net/xen-
netfront.ko
drwxr-xr-x 2 root root 0 Sep 28 10:21 lib/modules/4.4.21-69-default/updates/pvdriver/xen-hcall
-rwxr-xr-x 1 root root 8320 Sep 28 10:21 lib/modules/4.4.21-69-default/updates/pvdriver/xen-
hcall/xen-hcall.ko

sluo-ecs-30dc:~ # lsinitrd /boot/initrd-`uname -r` | grep virtio
-rw-r--r-- 1 root root 29335 Oct 26 2016 lib/modules/4.4.21-69-default/kernel/drivers/block/
virtio_blk.ko
-rw-r--r-- 1 root root 57007 Oct 26 2016 lib/modules/4.4.21-69-default/kernel/drivers/net/
virtio_net.ko
-rw-r--r-- 1 root root 32415 Oct 26 2016 lib/modules/4.4.21-69-default/kernel/drivers/scsi/
virtio_scsi.ko
drwxr-xr-x 2 root root 0 Sep 28 10:21 lib/modules/4.4.21-69-default/kernel/drivers/virtio
-rw-r--r-- 1 root root 19623 Oct 26 2016 lib/modules/4.4.21-69-default/kernel/drivers/virtio/
virtio.ko
-rw-r--r-- 1 root root 38943 Oct 26 2016 lib/modules/4.4.21-69-default/kernel/drivers/virtio/
virtio_pci.ko
-rw-r--r-- 1 root root 24431 Oct 26 2016 lib/modules/4.4.21-69-default/kernel/drivers/virtio/
virtio_ring.ko
    
```

📖 说明

如果误将built-in形式存在内核中的驱动添加到initrd或initramfs文件中，不会影响云服务器正常使用，这里全写进去只是为了修改方便，但是使用lsinitrd命令无法检查到。可使用如下方法确定这些驱动是否以built-in形式存在内核中，例如：

```

cat /boot/config-`uname -r` | grep CONFIG_VIRTIO | grep y
cat /boot/config-`uname -r` | grep CONFIG_XEN | grep y
    
```

10.8 镜像复制类

什么场景下需要复制镜像？

- 区域内复制
使镜像具备一些高级特性（如快速发放）的场景。详情请参见[复制镜像](#)。

复制一个镜像需要多久？

复制镜像的时间取决于网络传输速度和任务队列的排队数量。

复制镜像怎么收费？

- 区域内复制
系统盘镜像、数据盘镜像：复制后的镜像存储在OBS桶中，免费使用。

📖 说明

整机镜像不支持区域内复制。

10.9 镜像删除类

云服务器退订或删除后，使用该云服务器创建的私有镜像是否自动删除？

使用云服务器创建的私有镜像存储在OBS中，退订或删除云服务器不会对私有镜像造成影响，您仍可以继续使用。

我的镜像配额不够了，想删除其中一个镜像，但是这个镜像曾共享给了其他人，能删除吗？

- 基于项目ID共享的镜像可以删除，无需镜像的接受方进行相关操作，您执行删除后，镜像接受方的共享镜像也同步删除。因此，为避免影响对方正常业务，请在删除前通知对方做好备份。

如何删除共享镜像，删除后对已创建好的云服务器或云硬盘有无影响？

可以通过“拒绝”操作删除共享镜像。删除后对已创建好的云服务器或云硬盘没有影响。

10.10 Cloud-Init 操作类

10.10.1 安装 Cloud-Init FAQ

为了保证使用私有镜像创建的新云服务器可以自定义配置（例如修改云服务器密码），请在创建私有镜像前安装Cloud-Init工具。

安装Cloud-Init相关操作请参考：[安装Cloud-Init工具](#)。

配置Cloud-Init相关操作请参考：[配置Cloud-Init工具](#)。

本节列举了安装Cloud-Init的常见问题及解决方法。

Ubuntu 16.04/CentOS 7: Cloud-Init 设置自启动失败

- 问题描述：
安装Cloud-Init后，执行以下命令设置Cloud-Init为开机自启动时，提示失败：
systemctl enable cloud-init-local.service cloud-init.service cloud-config.service cloud-final.service
回显信息如下所示：

图 10-38 设置 Cloud-Init 自启动失败

```
root@ecs-wjq-ubuntu14:~# systemctl enable cloud-init-local.service cloud-init.service cloud-config.service cloud-final.service
Failed to execute operation: Unit file is masked
root@ecs-wjq-ubuntu14:~#
```

- 解决方法：
 - a. 执行以下命令回退配置
systemctl unmask cloud-init-local.service cloud-init.service cloud-config.service cloud-final.service
 - b. 执行以下命令，重新设置Cloud-Init为开机自启动
systemctl enable cloud-init-local.service cloud-init.service cloud-config.service cloud-final.service
 - c. 执行以下命令，检查Cloud-Init服务状态
systemctl status cloud-init-local.service cloud-init.service cloud-config.service cloud-final.service
回显信息如下所示，提示failed且所有服务的状态都是inactive。

图 10-39 检查 Cloud-Init 服务状态

```
root@ecs-wjq-ubuntu14:~# systemctl status cloud-init-local.service
● cloud-init-local.service - Initial cloud-init job (pre-networking)
   Loaded: loaded (/lib/systemd/system/cloud-init-local.service; enabled; vendor
   Active: failed (Result: exit-code) since Fri 2018-08-17 07:12:20 UTC; 1min 25
   Process: 4418 ExecStart=/usr/bin/cloud-init init --local (code=exited, status=
   Main PID: 4418 (code=exited, status=203/EXEC)

Aug 17 07:12:20 ecs-wjq-ubuntu14 systemd[1]: Starting Initial cloud-init job (pr
Aug 17 07:12:20 ecs-wjq-ubuntu14 systemd[1]: cloud-init-local.service: Main proc
Aug 17 07:12:20 ecs-wjq-ubuntu14 systemd[1]: Failed to start initial cloud-init
Aug 17 07:12:20 ecs-wjq-ubuntu14 systemd[1]: cloud-init-local.service: Unit ente
Aug 17 07:12:20 ecs-wjq-ubuntu14 systemd[1]: cloud-init-local.service: Failed wi
lines 1-11/11 (END)
```

图 10-40 检查 Cloud-Init 服务状态

```
● cloud-init-local.service - Initial cloud-init job (pre-networking)
   Loaded: loaded (/lib/systemd/system/cloud-init-local.service; enabled; vendor
   Active: failed (Result: exit-code) since Fri 2018-08-17 07:12:20 UTC; 59s ago
   Process: 4418 ExecStart=/usr/bin/cloud-init init --local (code=exited, status=
   Main PID: 4418 (code=exited, status=203/EXEC)

Aug 17 07:12:20 ecs-wjq-ubuntu14 systemd[1]: Starting Initial cloud-init job (pr
Aug 17 07:12:20 ecs-wjq-ubuntu14 systemd[1]: cloud-init-local.service: Main proc
Aug 17 07:12:20 ecs-wjq-ubuntu14 systemd[1]: Failed to start Initial cloud-init
Aug 17 07:12:20 ecs-wjq-ubuntu14 systemd[1]: cloud-init-local.service: Unit ente
Aug 17 07:12:20 ecs-wjq-ubuntu14 systemd[1]: cloud-init-local.service: Failed wi
th result 'exit-code'.
```

原因是系统查询Cloud-Init的地址指向/usr/bin/，但是实际安装路径指定的是/usr/local/bin，因此出现了上述错误。

- d. 执行以下命令，将cloud-init复制至usr/bin目录下。

cp /usr/local/cloud-init /usr/bin/

- e. 执行以下命令，重启Cloud-Init。

systemctl restart cloud-init-local.service cloud-init.service cloud-config.service cloud-final.service

图 10-41 重启 Cloud-Init

```
root@ecs-wjq-ubuntu14:~# systemctl start cloud-init-local.service; systemctl sta
tus cloud-init-local.service
● cloud-init-local.service - Initial cloud-init job (pre-networking)
   Loaded: loaded (/lib/systemd/system/cloud-init-local.service; enabled; vendor
   Active: active (exited) since Fri 2018-08-17 07:18:01 UTC; 4ms ago
   Process: 4491 ExecStart=/usr/bin/cloud-init init --local (code=exited, status=
   Main PID: 4491 (code=exited, status=0/SUCCESS)

Aug 17 07:18:01 ecs-wjq-ubuntu14 cloud-init[4491]: [CLOUDINIT] util.py[DEBUG]: R
Aug 17 07:18:01 ecs-wjq-ubuntu14 cloud-init[4491]: [CLOUDINIT] util.py[DEBUG]: R
Aug 17 07:18:01 ecs-wjq-ubuntu14 cloud-init[4491]: [CLOUDINIT] __init__.py[DEBU
Aug 17 07:18:01 ecs-wjq-ubuntu14 cloud-init[4491]: [CLOUDINIT] util.py[DEBUG]: R
Aug 17 07:18:01 ecs-wjq-ubuntu14 cloud-init[4491]: [CLOUDINIT] util.py[DEBUG]: R
Aug 17 07:18:01 ecs-wjq-ubuntu14 cloud-init[4491]: [CLOUDINIT] util.py[DEBUG]: F
Aug 17 07:18:01 ecs-wjq-ubuntu14 cloud-init[4491]: [CLOUDINIT] cloud-init[DEBU
Aug 17 07:18:01 ecs-wjq-ubuntu14 cloud-init[4491]: [CLOUDINIT] util.py[DEBUG]: R
Aug 17 07:18:01 ecs-wjq-ubuntu14 cloud-init[4491]: [CLOUDINIT] util.py[DEBUG]: R
Aug 17 07:18:01 ecs-wjq-ubuntu14 cloud-init[4491]: [CLOUDINIT] util.py[DEBUG]: c
lines 1-16/16 (END)
```

- f. 执行以下命令，检查Cloud-Init服务状态。

systemctl status cloud-init-local.service cloud-init.service cloud-config.service cloud-final.service

Ubuntu 14.04: 未安装 chkconfig 和 systemctl

- 问题描述：
未安装chkconfig

- 解决方法：
执行以下命令安装chkconfig：
apt-get update
apt-get install sysv-rc-conf
cp /usr/sbin/sysv-rc-conf /usr/sbin/chkconfig
安装完成后，执行以下命令查看版本号。
cloud-init -v
如果回显信息如下所示：

```
-bash:/usr/bin/cloud-init: not found this command
```


解决方法：执行以下命令将cloud-init复制至usr/bin目录下。
cp /usr/local/bin/cloud-init /usr/bin/

Debian 9.5: 查看版本号失败和 Cloud-Init 设置开启自启动失败

1. Cloud-Init安装完成后，执行以下命令查看版本号。
cloud-init -v
回显信息如下所示：

```
-bash:/usr/bin/cloud-init: not found this command
```


解决方法：执行命令**cp /usr/local/bin/cloud-init /usr/bin/**，将cloud-init复制至usr/bin目录下。
2. 执行命令**cloud-init init --local**
回显信息如下所示：

图 10-42 设置 Cloud-Init 自启动回显信息

```
root@ecs-debian-9:/tmp/CLOUD-INIT# cloud-init# cloud-init init --local
/usr/local/lib/python2.7/dist-packages/Cheetah-2.4.4-py2.7.egg/Cheetah/Compiler.py:1509: UserWarning:
You don't have the C version of NameMapper installed! I'm disabling Cheetah's useStackFrames option as it is painfully slow with
the Python version of NameMapper. You should get a copy of Cheetah with the compiled C version of NameMapper.
"\nYou don't have the C version of NameMapper installed!"
Cloud-Init v. 0.7.6 running 'init-local' at Mon, 20 Aug 2018 02:31:45 +0000. Up 704.40 seconds.
root@ecs-debian-9:/tmp/CLOUD-INIT# cloud-init#
```

原因分析：未安装gcc导致无法编译。

解决方法：

执行以下命令安装gcc后重新安装Cloud-Init。

yum -y install gcc

3. 执行以下命令设置Cloud-Init为开机自启动时提示失败：
**systemctl enable cloud-init-local.service cloud-init.service cloud-
config.service cloud-final.service**
回显信息提示如下所示：

图 10-43 设置 Cloud-Init 自启动提示失败

```
Failed to enable unit: Unit file /etc/systemd/system/cloud-init-local.service is masked.
```

解决方法：

- a. 执行以下命令回退配置

**systemctl unmask cloud-init-local.service cloud-init.service cloud-
config.service cloud-final.service**

- b. 执行以下命令，重新设置Cloud-Init为开机自启动
systemctl enable cloud-init-local.service cloud-init.service cloud-config.service cloud-final.service
 - c. 执行以下命令，重启Cloud-Init
systemctl restart cloud-init-local.service cloud-init.service cloud-config.service cloud-final.service
- 然后执行命令**systemctl status**查看服务状态。回显信息如下所示：

图 10-44 查看服务状态

```

cloud-init-local.service - Initial cloud-init job (pre-networking)
Loaded: loaded (/lib/systemd/system/cloud-init-local.service; enabled; vendor preset: enabled)
Active: active (exited) since Mon 2018-08-20 02:48:37 UTC; 6s ago
Process: 1082 ExecStart=/usr/bin/cloud-init init --local (code=exited, status=0/SUCCESS)
Main PID: 1082 (code=exited, status=0/SUCCESS)
Tasks: 0 (limit: 4915)
CGroup: /system.slice/cloud-init-local.service

Aug 20 02:48:37 ecs-debian-9 cloud-init[1082]: [CLOUDINIT] util.py[DEBUG]: Running command ['blkid', '-tLABEL=config-2', '--device
Aug 20 02:48:37 ecs-debian-9 cloud-init[1082]: [CLOUDINIT] _init.py[DEBUG]: Seeing if we can get any data from class 'cloudi
Aug 20 02:48:37 ecs-debian-9 cloud-init[1082]: [CLOUDINIT] util.py[DEBUG]: Reading from /proc/mounts (quiet=False)
Aug 20 02:48:37 ecs-debian-9 cloud-init[1082]: [CLOUDINIT] util.py[DEBUG]: Read 1347 bytes from /proc/mounts
Aug 20 02:48:37 ecs-debian-9 cloud-init[1082]: [CLOUDINIT] util.py[DEBUG]: Fetched 'deuopts': {'mountpoint': '/dev/pts', 'opts':
Aug 20 02:48:37 ecs-debian-9 cloud-init[1082]: [CLOUDINIT] cloud-init[DEBUG]: No local datasource found
Aug 20 02:48:37 ecs-debian-9 cloud-init[1082]: [CLOUDINIT] util.py[DEBUG]: Reading from /proc/uptime (quiet=False)
Aug 20 02:48:37 ecs-debian-9 cloud-init[1082]: [CLOUDINIT] util.py[DEBUG]: Read 14 bytes from /proc/uptime
Aug 20 02:48:37 ecs-debian-9 cloud-init[1082]: [CLOUDINIT] util.py[DEBUG]: cloud-init mode 'init' took 0.104 seconds (0.10)
Aug 20 02:48:37 ecs-debian-9 systemd[1]: Started Initial cloud-init job (pre-networking).

cloud-init.service - Initial cloud-init job (metadata service crawler)
Loaded: loaded (/lib/systemd/system/cloud-init.service; enabled; vendor preset: enabled)
Active: active (exited) since Mon 2018-08-20 02:48:40 UTC; 3s ago
Process: 1096 ExecStart=/usr/bin/cloud-init init (code=exited, status=0/SUCCESS)
Main PID: 1096 (code=exited, status=0/SUCCESS)
Tasks: 0 (limit: 4915)
CGroup: /system.slice/cloud-init.service

Aug 20 02:48:40 ecs-debian-9 cloud-init[1096]: [CLOUDINIT] helpers.py[DEBUG]: config-ca-certs already ran (freq=once-per-instance
Aug 20 02:48:40 ecs-debian-9 cloud-init[1096]: [CLOUDINIT] stages.py[DEBUG]: Running module rsyslog (module 'cloudinit.config.c
Aug 20 02:48:40 ecs-debian-9 cloud-init[1096]: [CLOUDINIT] helpers.py[DEBUG]: config-rsyslog already ran (freq=once-per-instance
Aug 20 02:48:40 ecs-debian-9 cloud-init[1096]: [CLOUDINIT] stages.py[DEBUG]: Running module users-groups (module 'cloudinit.com
Aug 20 02:48:40 ecs-debian-9 cloud-init[1096]: [CLOUDINIT] helpers.py[DEBUG]: config-users-groups already ran (freq=once-per-ins
Aug 20 02:48:40 ecs-debian-9 cloud-init[1096]: [CLOUDINIT] cloud-init[DEBUG]: Ran 13 modules with 0 failures
Aug 20 02:48:40 ecs-debian-9 cloud-init[1096]: [CLOUDINIT] util.py[DEBUG]: Reading from /proc/uptime (quiet=False)
Aug 20 02:48:40 ecs-debian-9 cloud-init[1096]: [CLOUDINIT] util.py[DEBUG]: Read 14 bytes from /proc/uptime
Aug 20 02:48:40 ecs-debian-9 cloud-init[1096]: [CLOUDINIT] util.py[DEBUG]: cloud-init mode 'init' took 2.657 seconds (2.66)
Aug 20 02:48:40 ecs-debian-9 systemd[1]: Started Initial cloud-init job (metadata service crawler).

cloud-config.service - Apply the settings specified in cloud-config
Loaded: loaded (/lib/systemd/system/cloud-config.service; enabled; vendor preset: enabled)
Active: active (exited) since Mon 2018-08-20 02:48:41 UTC; 2s ago
Process: 1140 ExecStart=/usr/bin/cloud-init modules --mode=config (code=exited, status=0/SUCCESS)
Main PID: 1140 (code=exited, status=0/SUCCESS)
Tasks: 0 (limit: 4915)
CGroup: /system.slice/cloud-config.service
    
```

CentOS 7/Fedora 28 操作系统：安装 Cloud-Init 时提示未安装所需的 C 编译器

- 问题描述

Cloud-Init安装完成后，执行以下命令：

cloud-init init --local

回显信息出现以下警告：

```

/usr/lib/python2.5/site-packages/Cheetah/Compiler.py:1532: UserWarning:
You don't have the C version of NameMapper installed! I'm disabling Cheetah's useStackFrames
option as it is painfully slow with the Python version of NameMapper. You should get a copy of
Cheetah with the compiled C version of NameMapper.
"\nYou don't have the C version of NameMapper installed!
    
```

- 原因分析

出现该警告是因为在安装Cloud-Init时，需要编译C版本的NameMapper，但是该系统没有预装gcc，因此无法编译，这才导致C版本的NameMapper缺失。

- 处理方法

执行以下命令安装gcc：

yum -y install gcc

然后再重新安装Cloud-Init。

CentOS 7/Fedora 操作系统：Cloud-Init 安装成功，使用镜像创建的云服务器无法使用新密码登录

- 问题描述

Cloud-Init安装成功后，生成镜像并申请云服务器，启动新的云服务器，发现新密码无效。使用旧密码登录进去后发现，网卡没有启动：

图 10-45 网卡未启动

```
root@ecs-fedora28-wjq-test ~]# ifconfig
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

- 解决方法

登录创建镜像前的云服务器，打开dhcp配置文件“/etc/sysconfig/network-scripts/ifcfg-ethX”，注释HWADDR这一项。

10.10.2 云服务器安装 Cloud-Init 可以做什么？

Cloud-Init 介绍

Cloud-Init是开源的云初始化工具，使用安装了Cloud-Init的镜像创建云服务器时可以通过“用户数据注入”功能，注入初始化自定义信息（例如为云服务器设置登录密码）；还可以通过查询、使用元数据，对正在运行的云服务器进行配置和管理。不安装Cloud-Init工具，将无法对云服务器进行自定义配置，只能使用镜像原有密码登录云服务器。

安装方法

建议您在创建私有镜像前先安装Cloud-Init/Cloudbase-Init，安装方法如下：

- 如果是Windows操作系统，请下载并安装Cloudbase-Init。
安装Cloudbase-Init相关操作请参考[安装并配置Cloudbase-Init工具](#)。
- 如果是Linux操作系统，请下载并安装Cloud-Init。
安装Cloud-Init相关操作请参考[安装Cloud-Init工具](#)。
配置Cloud-Init相关操作请参考[配置Cloud-Init工具](#)。

10.10.3 安装 NetworkManager 后，使用 Cloud-Init 注入密钥或密码失败怎么办？

问题描述

安装NetworkManager后使用Cloud-Init注入密钥或密码失败的主要是由于Cloud-Init版本与NetworkManager不兼容导致。目前Debian 9.0及以上版本中NetworkManager与Cloud-Init 0.7.9版本存在兼容性问题。

处理方法

请卸载当前版本Cloud-Init，安装0.7.6及以下版本的Cloud-Init。

安装方法请参见[安装Cloud-Init工具](#)。

10.10.4 SUSE 11 SP4 如何安装 growpart?

操作场景

SUSE/openSUSE系列growpart工具是独立的工具包，不是以“cloud-*”开头。请参考以下步骤安装growpart工具。

操作步骤

1. 执行以下命令查询cloud-init和growpart是否已安装。

```
rpm -qa | grep cloud-init
```

回显如下：

```
cloud-init-0.7.8-39.2
```

```
rpm -qa | grep growpart
```

回显如下：

```
growpart-0.29-8.1
```

2. 执行以下命令卸载已安装的cloud-init和growpart。

```
zypper remove cloud-init growpart
```

3. 执行以下命令清理残留文件。

```
rm -fr /etc/cloud/*
```

```
rm -fr /var/lib/cloud/*
```

4. 执行以下命令安装growpart。

```
zypper install http://download.opensuse.org/repositories/home:/garloff:/OTC:/cloudinit/SLE_11_SP4/noarch/growpart-0.27-1.1.noarch.rpm
```

5. 执行以下命令安装python-oauth。

```
zypper install http://download.opensuse.org/repositories/home:/garloff:/OTC:/cloudinit/SLE_11_SP4/x86_64/python-oauth-1.0.1-35.1.x86_64.rpm
```

6. 执行以下命令安装cloud-init。

```
zypper install http://download.opensuse.org/repositories/home:/garloff:/OTC:/cloudinit/SLE_11_SP4/x86_64/cloud-init-0.7.6-27.23.1.x86_64.rpm
```

7. 执行以下命令查看growpart、python-oauth和cloud-init是否安装成功。

```
rpm -qa | grep growpart
```

回显如下：

```
growpart-0.27-1.1
```

```
rpm -qa | grep python-oauth
```

回显如下：

```
python-oauthlib-0.6.0-1.5
```

```
python-oauth-1.0.1-35.1
```

```
rpm -qa | grep cloud-init
```

回显如下：

```
cloud-init-0.7.6-27.19.1
```

8. 执行以下命令检查配置。

```
chkconfig cloud-init-local on;chkconfig cloud-init on;chkconfig cloud-  
config on;chkconfig cloud-final on
```

10.11 云服务器创建类

10.11.1 云服务器购买成功后可以换镜像吗？

可以。

如果由于镜像选择错误，业务需求变化，或者其他原因需要更换镜像，可以使用“切换操作系统”功能进行更换。

支持不同镜像类型（包括公共镜像、私有镜像、共享镜像）与不同操作系统之间互相切换。您可以将现有的操作系统切换为不同镜像类型的操作系统。如果您想将操作系统从英文更改为中文，也可以通过“切换操作系统”来实现。详细操作请参考《弹性云服务器用户指南》“切换操作系统”章节。

10.11.2 使用私有镜像创建的云服务器，是否可以与生成镜像的云服务器硬件规格不同？

使用私有镜像创建的云服务器，其系统盘大小可以指定，必须大于等于镜像的系统盘大小，且小于1024GB。CPU、内存、带宽、数据盘可以根据需要进行修改。

使用整机私有镜像创建云服务器时，如果整机镜像中包含了一块或多块数据盘，系统会自动设置好数据盘参数。您可以增加系统盘和数据盘容量，但不能缩小。

使用私有镜像创建云服务器具体操作请参见[通过镜像创建云服务器](#)。

10.11.3 使用镜像创建云服务器，可以指定系统盘大小吗？

使用整机私有镜像创建云服务器时，如果整机镜像中包含了一块或多块数据盘，系统会自动设置好数据盘参数。您可以增加系统盘和数据盘容量，但不能缩小。

使用私有镜像创建云服务器具体操作请参见[通过镜像创建云服务器](#)。

使用镜像创建云服务器，其系统盘大小可以指定，必须不小于镜像的系统盘大小并小于30768GB。

说明

请确认您的操作系统要能够支持创建云服务器时所设定的系统盘大小。

10.11.4 使用外部导入的私有镜像所创建的云服务器在启动过程中提示找不到分区，如何处理？

问题描述

将外部镜像文件制作为私有镜像后，使用该私有镜像所创建的云服务器在启动过程中提示找不到分区，通常是由于跨平台后，磁盘分区的ID发生变化，导致镜像无法按照原来磁盘分区ID查询对应的分区名，此时，需要将镜像中的磁盘分区修改为“UUID=**磁盘分区的UUID**”。

处理方法

以openSUSE 13.2操作系统的镜像为例，修改分区名的操作步骤如下：

1. 执行以下命令，确认当前磁盘分区对应的ID。

```
ls -l /dev/disk/by-id/
```

回显类似如下所示。

```
total 0
lrwxrwxrwx 1 root root 10 Jul 22 01:35 ata-QEMU_HARDDISK_QM00001 -> ../../xvda
lrwxrwxrwx 1 root root 11 Jul 22 01:35 ata-QEMU_HARDDISK_QM00001-part1 -> ../../xvda1
lrwxrwxrwx 1 root root 12 Jul 22 01:35 ata-QEMU_HARDDISK_QM00001-part10 -> ../../xvda10
lrwxrwxrwx 1 root root 11 Jul 22 01:35 ata-QEMU_HARDDISK_QM00001-part2 -> ../../xvda2
lrwxrwxrwx 1 root root 11 Jul 22 01:35 ata-QEMU_HARDDISK_QM00001-part5 -> ../../xvda5
lrwxrwxrwx 1 root root 11 Jul 22 01:35 ata-QEMU_HARDDISK_QM00001-part6 -> ../../xvda6
lrwxrwxrwx 1 root root 11 Jul 22 01:35 ata-QEMU_HARDDISK_QM00001-part7 -> ../../xvda7
lrwxrwxrwx 1 root root 11 Jul 22 01:35 ata-QEMU_HARDDISK_QM00001-part8 -> ../../xvda8
lrwxrwxrwx 1 root root 11 Jul 22 01:35 ata-QEMU_HARDDISK_QM00001-part9 -> ../../xvda9
lrwxrwxrwx 1 root root 10 Jul 22 01:35 ata-QEMU_HARDDISK_QM00005 -> ../../xvde
lrwxrwxrwx 1 root root 10 Jul 22 01:35 scsi-SATA_QEMU_HARDDISK_QM00001 -> ../../xvda
lrwxrwxrwx 1 root root 11 Jul 22 01:35 scsi-SATA_QEMU_HARDDISK_QM00001-part1 -> ../../xvda1
lrwxrwxrwx 1 root root 12 Jul 22 01:35 scsi-SATA_QEMU_HARDDISK_QM00001-part10 -> ../../xvda10
lrwxrwxrwx 1 root root 11 Jul 22 01:35 scsi-SATA_QEMU_HARDDISK_QM00001-part2 -> ../../xvda2
lrwxrwxrwx 1 root root 11 Jul 22 01:35 scsi-SATA_QEMU_HARDDISK_QM00001-part5 -> ../../xvda5
lrwxrwxrwx 1 root root 11 Jul 22 01:35 scsi-SATA_QEMU_HARDDISK_QM00001-part6 -> ../../xvda6
lrwxrwxrwx 1 root root 11 Jul 22 01:35 scsi-SATA_QEMU_HARDDISK_QM00001-part7 -> ../../xvda7
lrwxrwxrwx 1 root root 11 Jul 22 01:35 scsi-SATA_QEMU_HARDDISK_QM00001-part8 -> ../../xvda8
lrwxrwxrwx 1 root root 11 Jul 22 01:35 scsi-SATA_QEMU_HARDDISK_QM00001-part9 -> ../../xvda9
lrwxrwxrwx 1 root root 10 Jul 22 01:35 scsi-SATA_QEMU_HARDDISK_QM00005 -> ../../xvde
```

其中，“ata-QEMU_HARDDISK_xxx”和“scsi-SATA_QEMU_HARDDISK_xxx”表示该云服务器的磁盘为qemu模拟设备。磁盘分区ID为“->”左侧内容，对应的分区名为“->”右侧内容。

2. 执行以下命令，确认当前磁盘分区对应的UUID。

```
ls -l /dev/disk/by-uuid/
```

回显类似如下所示。

```
total 0
lrwxrwxrwx 1 root root 11 Jul 22 01:35 45ecd7a0-29da-4402-a017-4564a62308b8 -> ../../xvda5
lrwxrwxrwx 1 root root 11 Jul 22 01:35 55386c6a-9e32-41d4-af7a-e79596221f51 -> ../../xvda9
lrwxrwxrwx 1 root root 11 Jul 22 01:35 55f36660-9bac-478c-a701-7ecc5347f789 -> ../../xvda8
lrwxrwxrwx 1 root root 11 Jul 22 01:35 780f36bc-0ada-4c98-9a8d-44570d65333d -> ../../xvda1
lrwxrwxrwx 1 root root 11 Jul 22 01:35 b3b7c47f-6a91-45ef-80d6-275b1cc16e19 -> ../../xvda6
lrwxrwxrwx 1 root root 11 Jul 22 01:35 ea63b55d-3b6e-4dcd-8986-956b72bac3e9 -> ../../xvda7
lrwxrwxrwx 1 root root 12 Jul 22 01:35 eb3cc645-925e-4bc5-bedf-c2a6f3b65809 -> ../../xvda10
```

其中，每条磁盘信息中，“->”左侧为磁盘的UUID，“->”右侧为对应的磁盘分区名。结合1结果，可以得到磁盘分区名，磁盘分区ID和磁盘分区UUID的对应关系。

3. 执行以下命令，查看“/etc/fstab”文件中分区名称。

```
vi /etc/fstab
```

回显类似如下所示。

```
/dev/disk/by-id/scsi-SATA_QEMU_HARDDISK_QM00001-part5 / ext3 defaults,errors=panic 1 1
/dev/disk/by-id/scsi-SATA_QEMU_HARDDISK_QM00001-part1 /boot ext3 defaults,errors=panic 1 2
/dev/disk/by-id/scsi-SATA_QEMU_HARDDISK_QM00001-part6 /home ext3 nosuid,errors=panic 1 2
/dev/disk/by-id/scsi-SATA_QEMU_HARDDISK_QM00001-part10 /opt ext3 defaults,errors=panic 1 2
/dev/disk/by-id/scsi-SATA_QEMU_HARDDISK_QM00001-part7 /tmp ext3 nodev,nosuid,errors=panic 1 2
/dev/disk/by-id/scsi-SATA_QEMU_HARDDISK_QM00001-part9 /usr ext3 defaults,errors=panic 1 2
/dev/disk/by-id/scsi-SATA_QEMU_HARDDISK_QM00001-part8 /var ext3 nodev,nosuid,errors=panic 1 2
sysfs /sys sysfs noauto 0 0
proc /proc proc defaults 0 0
usbfs /proc/bus/usb usbfs noauto 0 0
devpts /dev/pts devpts mode=0620,gid=5 0 0
```

```
/dev/cdrom /media/ udf,iso9660 noexec,noauto,nouser,nodev,nosuid 1 2
tmpfs /dev/shm tmpfs noexec,nodev,nosuid 0 0
```

回显中第一列为磁盘分区ID。

- 按“i”进入编辑模式，按照1和2的查询结果，将3“/etc/fstab”文件中“/dev/disk/xxx”所在行磁盘分区ID修改为“UUID=**磁盘分区的UUID**”的形式。

修改结果如下所示。

```
UUID=45ecd7a0-29da-4402-a017-4564a62308b8 / ext3 defaults,errors=panic 1 1
UUID=780f36bc-0ada-4c98-9a8d-44570d65333d /boot ext3 defaults,errors=panic 1 2
UUID=b3b7c47f-6a91-45ef-80d6-275b1cc16e19 /home ext3 nosuid,errors=panic 1 2
UUID=eb3cc645-925e-4bc5-bedf-c2a6f3b65809 /opt ext3 defaults,errors=panic 1 2
UUID=ea63b55d-3b6e-4dcd-8986-956b72bac3e9 /tmp ext3 nodev,nosuid,errors=panic 1 2
UUID=55386c6a-9e32-41d4-af7a-e79596221f51 /usr ext3 defaults,errors=panic 1 2
UUID=55f36660-9bac-478c-a701-7ecc5347f789 /var ext3 nodev,nosuid,errors=panic 1 2
sysfs /sys sysfs noauto 0 0
proc /proc proc defaults 0 0
usbfs /proc/bus/usb usbfs noauto 0 0
devpts /dev/pts devpts mode=0620,gid=5 0 0
/dev/cdrom /media/ udf,iso9660 noexec,noauto,nouser,nodev,nosuid 1 2
tmpfs /dev/shm tmpfs noexec,nodev,nosuid 0 0
```

📖 说明

请确保对应的UUID修改正确，否则会造成云服务器无法正常启动。

- 按“Esc”后，输入:wq，按“Enter”。保存设置并退出vi编辑器。
- 检查引导系统的配置文件中的分区名。
不同操作系统对应的系统引导配置文件，通常为如下几种。请确认当前操作系统使用系统引导配置文件的文件名。

- grub引导配置文件
 - /boot/grub/grub.conf
 - /boot/grub/menu.lst
 - /boot/grub/grub.cfg
 - /boot/grub2/grub.cfg
- syslinux引导配置文件
 - /extlinux.conf
 - /boot/syslinux/extlinux.conf
 - /boot/extlinux/extlinux.conf
 - /boot/syslinux/syslinux.cfg
 - /syslinux/syslinux.cfg
 - /syslinux.cfg

本例中引导文件为“/boot/grub/menu.lst”，请执行以下命令，查看引导文件。

vi /boot/grub/menu.lst

```
default 0
timeout 3
title xxx Server OS - xxxxxx
kernel /boot/vmlinuz-3.0.101-0.47.52-default root=/dev/disk/by-id/scsi-
SATA_QEMU_HARDDISK_QM00001-part5 resume= memmap=0x2000000$0x3E000000
nmi_watchdog=2 crashkernel=512M-:256M console=ttyS0,115200 console=tty0 xen_emul_unplug=all
initrd /boot/initrd-3.0.101-0.47.52-default
```

- 按“i”进入编辑模式，修改引导系统的配置文件中的分区名。
按照1和2的查询结果，将6“/boot/grub/menu.lst”文件中的磁盘分区名修改为“UUID=**磁盘分区的UUID**”的形式。

```
default 0
timeout 3
title xxx Server OS - xxxxxx
kernel /boot/vmlinuz-3.0.101-0.47.52-default root=UUID=45ecd7a0-29da-4402-a017-4564a62308b8
resume= memmap=0x2000000$0x3E000000 nmi_watchdog=2 crashkernel=512M-:256M
console=ttyS0,115200 console=tty0 xen_emul_unplug=all
initrd /boot/initrd-3.0.101-0.47.52-default
```

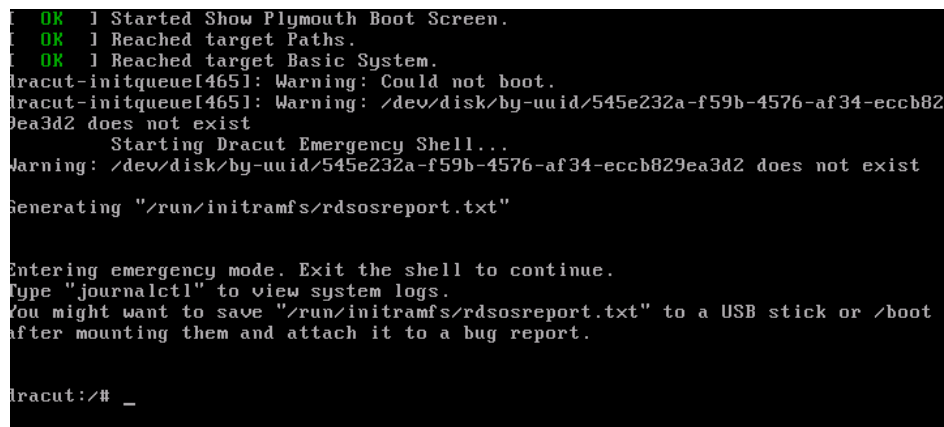
- 按“Esc”后，输入:wq，按“Enter”。保存设置并退出vi编辑器。

10.11.5 注册的镜像操作系统是 CentOS 类型，使用该镜像创建的云服务器找不到磁盘该如何处理？

问题描述

使用CentOS系列私有镜像创建的云服务器找不到磁盘，通常情况下，这是由于启动时没有加载xen-blkfront.ko模块导致的，您需要修改操作系统内核启动参数。登录云服务器后的启动界面如图10-46所示。

图 10-46 启动界面



```
OK | Started Show Plymouth Boot Screen.
OK | Reached target Paths.
OK | Reached target Basic System.
dracut-initqueue[465]: Warning: Could not boot.
dracut-initqueue[465]: Warning: /dev/disk/by-uuid/545e232a-f59b-4576-af34-eccb829ea3d2 does not exist
Starting Dracut Emergency Shell...
Warning: /dev/disk/by-uuid/545e232a-f59b-4576-af34-eccb829ea3d2 does not exist
Generating "/run/initramfs/rdsosreport.txt"

Entering emergency mode. Exit the shell to continue.
Type "journalctl" to view system logs.
You might want to save "/run/initramfs/rdsosreport.txt" to a USB stick or /boot
after mounting them and attach it to a bug report.

dracut:/# _
```

处理方法

修改操作系统内核启动参数操作步骤如下：

📖 说明

以下步骤需要在操作系统启动后执行，建议用户在原来创建镜像的源云服务器中进行修改。

- 执行以下命令，登录操作系统。
lsinitrd /boot/initramfs-`uname -r`.img |grep -i xen
 - 如果回显中包含“xen-blkfront.ko”的相关信息，请您与管理员沟通。
 - 如果没有回显信息，请执行2。
- 对下面原有的grub文件进行备份。
 - 对于CentOS 6系列，请执行以下命令进行备份。
cp /boot/grub/grub.conf /boot/grub/grub.conf.bak

- 对于CentOS 7系列，请执行以下命令进行备份。
cp /boot/grub2/grub.cfg /boot/grub2/grub.cfg.bak
- 3. 使用vi编辑器打开grub配置文件。以CentOS 7系列为例，执行以下命令。
vi /boot/grub2/grub.cfg
- 4. 在对应默认的启动内核后面增加参数“xen_emul_unplug=all”。

📖 说明

查找存在“root=UUID=”的配置行，在对应行的末尾都需要补充参数“xen_emul_unplug=all”。

```
menuentry 'CentOS Linux (3.10.0-229.el7.x86_64) 7 (Core) with debugging' --class centos --class gnu-  
linux --class gnu --class os --unrestricted $menuentry_id_option 'gnulinux-3.10.0-229.el7.x86_64-  
advanced-bf3cc825-7638-48d8-8222-cd2f412dd0de' {  
    load_video  
    set gfxpayload=keep  
    insmod gzio  
    insmod part_msdos  
    insmod ext2  
    set root='hd0,msdos1'  
    if [ x$feature_platform_search_hint = xy ]; then  
        search --no-floppy --fs-uuid --set=root --hint='hd0,msdos1' bf3cc825-7638-48d8-8222-  
cd2f412dd0de  
    else  
        search --no-floppy --fs-uuid --set=root bf3cc825-7638-48d8-8222-cd2f412dd0de  
    fi  
    linux16 /boot/vmlinuz-3.10.0-229.el7.x86_64 root=UUID=bf3cc825-7638-48d8-8222-  
cd2f412dd0de xen_emul_unplug=all ro crashkernel=auto rhgb quiet systemd.log_level=debug  
systemd.log_target=kmsg  
    initrd16 /boot/initramfs-3.10.0-229.el7.x86_64.img  
}
```

- 5. 按“ESC”，再输入:wq并单击回车，退出vi编辑器。
- 6. 将该云服务器制作成为镜像，重新上传并注册到云平台。

10.11.6 如何解决注册 Windows 镜像时已勾选“进行后台自动化设置”，但云服务器仍然无法正常启动？

问题描述

注册Windows镜像时已勾选“进行后台自动化设置”，但使用该镜像创建的云服务器仍然无法正常启动，可能是因为virtio驱动离线注入失败。

处理方法

Windows操作系统的云服务器virtio驱动离线注入有以下约束：

- 不支持为UEFI启动方式的镜像文件离线注入virtio驱动。
- GPO（Group Policy Object，组策略）建议关闭，某些策略可能会导致离线注入virtio驱动失败。
- 杀毒软件建议关闭，可能导致线注入virtio驱动失败。

如需安装virtio驱动，请参考[优化私有镜像（Windows）](#)。

10.11.7 使用 UEFI 启动方式的镜像创建云服务器，云服务器启动异常怎么办？

问题描述

使用UEFI启动方式的私有镜像，创建弹性云服务器。创建成功后，云服务器无法正常启动。

可能原因

镜像的操作系统是UEFI启动方式，但是镜像属性中未添加uefi属性。

处理方法

1. 删除启动异常的弹性云服务器。
2. 调用接口更新镜像属性，使“hw_firmware_type”取值为“uefi”。
接口URI：PATCH /v2/cloudimages/{image_id}
接口调用方法，请参见《镜像服务接口参考》的“更新镜像信息”章节。
3. 使用更新后的镜像，重新创建弹性云服务器。

10.12 驱动安装类

10.12.1 一定要在云服务器中安装 Guest OS driver 吗？

云服务器中安装Guest OS driver，可以增强云服务器使用时用户的体验。同时，可以保证云服务器的高可靠性以及稳定性。

- Windows操作系统必须安装PV driver和virtio。
- Linux操作系统必须使用xen-pv以及virtio原生驱动并装载到initrd。

10.12.2 Windows 操作系统为什么要安装并更新 virtio？

为什么要安装 virtio 驱动？

virtio是为弹性云服务器提供高性能磁盘和网卡的半虚拟化驱动（virtio driver）。

- 标准的Windows系统不会自带virtio driver。
- 平台提供的公共镜像默认已安装virtio。
- 私有镜像需用户自行安装virtio，安装指导请参见[安装virtio驱动](#)。

为什么要更新 virtio？

保持最新的驱动版本能有效避免触发社区发现的已知问题。

用户需要做哪些操作？

- 根据指导升级Windows私有镜像或正在运行的Windows实例中的驱动程序。

- 如有任何技术问题或疑问，请联系管理员获取帮助。

10.12.3 为什么 Windows 云服务器安装 Guest OS driver 会失败？

Guest OS driver安装失败，主要是由于以下几种原因：

- 如果您的镜像文件是从VMware虚拟机导出，可能是由于没有卸载VMware Tools或者VMware Tools没有卸载干净导致Guest OS driver安装失败。
- 请您务必选择正确的Windows版本的Guest OS driver下载，否则可能导致Guest OS driver安装失败。
- 由于安装Guest OS driver需要消耗的一定的空间，磁盘空间预留不足可能导致Guest OS driver安装失败。因此，请您预留约300MB的空间以确保Guest OS driver正确安装。

10.12.4 Windows 系统如何安装 virtio？

本章节适用于虚拟化类型为KVM架构的云服务器。

使用弹性云服务器或者外部镜像文件创建私有镜像时，必须确保操作系统中已安装virtio，使新发放的云服务器支持KVM虚拟化，同时也可以提升云服务器的网络性能。

具体操作步骤详见[安装virtio驱动](#)。

10.12.5 Linux 系统如何安装原生的 KVM 驱动？

在优化Linux私有镜像过程中，需要在云服务器上安装原生的KVM驱动。若云服务器已安装原生的KVM驱动，请忽略本章节。

具体操作步骤详见[安装原生的KVM驱动](#)。

10.12.6 怎样安装原生的 XEN 和 KVM 驱动

操作场景

对于XEN虚拟化类型的弹性云服务器，在优化Linux私有镜像过程中，需要安装原生的XEN和KVM驱动。

本节介绍安装原生XEN和KVM驱动的具体方法。

注意

对于XEN实例：如果不安装XEN驱动，弹性云服务器的网络性能很差，并且安全组和防火墙也不会生效；

对于KVM实例：如果不安装KVM驱动，弹性云服务器的网卡可能无法检测到，无法与外部通信。因此，请您务必安装XEN和KVM驱动。

前提条件

- 请先确认您当前的云服务器为XEN构架。
- 对于使用Linux系统原生的XEN和KVM驱动的Linux云服务器，其内核版本必须高于2.6.24。

- 建议您禁用任何防病毒软件或入侵检测软件，安装原生的XEN和KVM驱动完成后，您可以再次启用这些软件。

操作步骤

请根据操作系统版本，修改不同的配置文件：

- CentOS/EulerOS系列操作系统

以CentOS 7.0为例，请修改“/etc/dracut.conf”文件，在add_drivers项中添加xen-pv以及virtio的驱动（xen-pv驱动：xen-blkfront、xen-netfront；virtio驱动：virtio_blk、virtio_scsi、virtio_net、virtio_pci、virtio_ring、virtio）。驱动名之间以空格隔开，保存并退出“/etc/dracut.conf”文件，执行**dracut -f**命令，重新生成initrd。

操作方法可参见[CentOS/EulerOS系列操作系统相关操作](#)。

- Ubuntu/Debian系列系统

请修改“/etc/initramfs-tools/modules”文件，添加xen-pv以及virtio的驱动（xen-pv驱动：xen-blkfront、xen-netfront；virtio驱动：virtio_blk、virtio_scsi、virtio_net、virtio_pci、virtio_ring、virtio）。驱动名之间以空格隔开，保存并退出“/etc/initramfs-tools/modules”文件，执行**update-initramfs -u**命令，重新生成initrd。

操作方法可参见[Ubuntu/Debian系列操作系统相关操作](#)。

- SUSE和openSUSE系列系统，根据操作系统版本不同，修改不同的配置文件。
 - 当操作系统版本低于SUSE 12 SP1或低于openSUSE 13时，请修改“/etc/sysconfig/kernel”文件，在INITRD_MODULES=""添加xen-pv以及virtio的驱动（xen-pv驱动：xen_vnif、xen_vbd、xen_platform_pci；virtio驱动：virtio_blk、virtio_scsi、virtio_net、virtio_pci、virtio_ring、virtio）。驱动名之间以空格隔开，执行**mkinitrd**命令，重新生成initrd。
 - 当操作系统版本为SUSE 12 SP1时，修改“/etc/dracut.conf”文件，在add_drivers项中添加xen-pv以及virtio的驱动（xen-pv驱动：xen_vnif、xen_vbd、xen_platform_pci；virtio驱动：virtio_blk、virtio_scsi、virtio_net、virtio_pci、virtio_ring、virtio）。驱动名之间以空格隔开，执行命令**dracut -f**，重新生成initrd。
 - 当操作系统版本高于SUSE 12 SP1或高于openSUSE 13版本时，修改“/etc/dracut.conf”文件，在add_drivers项中添加xen-pv和virtio的驱动（xen-pv驱动：xen-blkfront、xen-netfront；virtio驱动：virtio_blk、virtio_scsi、virtio_net、virtio_pci、virtio_ring、virtio）。驱动名之间以空格隔开，保存并退出“/etc/dracut.conf”文件，执行**dracut -f**命令，重新生成initrd。

操作方法可参[SUSE/openSUSE系列操作系统相关操作](#)。

📖 说明

SUSE系列操作系统首先应确认OS是否已经安装了xen-kmp包（xen-pv的驱动包），执行以下命令：

```
rpm -qa |grep xen-kmp
```

回显类似如下：

```
xen-kmp-default-4.2.2_04_3.0.76_0.11-0.7.5
```

如果没有安装xen-kmp的包，请到ISO装机文件中获取并安装。

如果误将built-in形式的驱动添加到initrd或initramfs文件中，不会影响云服务器正常使用。

CentOS/EulerOS 系列操作系统相关操作

1. 执行以下命令，打开“/etc/dracut.conf”文件。

```
vi /etc/dracut.conf
```

2. 按“i”进入编辑模式，在“add_drivers”项中添加xen-pv和virtio的驱动（具体格式要根据操作系统本身的要求来决定）。

```
[root@CTU10000xxxx ~]# vi /etc/dracut.conf
# additional kernel modules to the default
add_drivers+="xen-blkfront xen-netfront virtio_blk virtio_scsi virtio_net virtio_pci virtio_ring virtio"
.....
```

3. 按“Esc”后，输入:wq，按“Enter”。保存设置并退出“/etc/dracut.conf”文件。

4. 执行以下命令，重新生成initrd。

```
dracut -f /boot/initramfs-2.6.32-573.8.1.el6.x86_64.img
```

如果引导的虚拟文件系统不是默认的initramfs，则命令为：**dracut -f 实际使用的initramfs文件名或者initrd文件名**。“实际使用的initramfs文件名或者initrd文件名”可在grub.cfg配置（“/boot/grub/grub.cfg”或“/boot/grub2/grub.cfg”或“/boot/grub/grub.conf”，具体路径根据OS不同会有所区别）中获取。

5. 如果引导的虚拟文件系统是initramfs，执行以下命令，检查是否已经成功装载了原生的XEN和KVM驱动相应模块。

```
lsinitrd /boot/initramfs-`uname -r`.img | grep xen
```

```
lsinitrd /boot/initramfs-`uname -r`.img | grep virtio
```

如果引导的虚拟文件系统是initrd，执行如下命令，检查是否已经成功装载了原生的XEN和KVM驱动相应模块。

```
lsinitrd /boot/initrd-`uname -r` | grep xen
```

```
lsinitrd /boot/initrd-`uname -r` | grep virtio
```

以引导的虚拟文件系统是initramfs为例，回显信息如下所示：

```
[root@CTU10000xxxx home]# lsinitrd /boot/initramfs-`uname -r`.img | grep xen
-rwxr--r-- 1 root root 54888 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/drivers/block/xen-blkfront.ko
-rwxr--r-- 1 root root 45664 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/drivers/net/xen-netfront.ko

[root@CTU10000xxxx home]# lsinitrd /boot/initramfs-`uname -r`.img | grep virtio
-rwxr--r-- 1 root root 23448 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/drivers/block/virtio_blk.ko
-rwxr--r-- 1 root root 50704 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/drivers/net/virtio_net.ko
-rwxr--r-- 1 root root 28424 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/drivers/scsi/virtio_scsi.ko
drwxr-xr-x 2 root root 0 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/drivers/virtio
-rwxr--r-- 1 root root 14544 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/drivers/virtio/virtio.ko
-rwxr--r-- 1 root root 21040 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/drivers/virtio/virtio_pci.ko
-rwxr--r-- 1 root root 18016 Jul 16 17:53 lib/modules/2.6.32-573.8.1.el6.x86_64/kernel/drivers/virtio/virtio_ring.ko
```


📖 说明

如果误将built-in形式存在内核中的驱动添加到initrd或initramfs文件中，不会影响云服务器正常使用，这里全写进去只是为了修改的方便，但是使用lsinitrd命令无法检查到。可使用如下方法确定这些驱动是否以built-in形式存在内核中，例如：

```
cat /boot/config-`uname -r` | grep CONFIG_VIRTIO | grep y
cat /boot/config-`uname -r` | grep CONFIG_XEN | grep y
```

Ubuntu/Debian 系列操作系统相关操作

1. 执行以下命令，打开“modules”文件。
vi /etc/initramfs-tools/modules
2. 按“i”进入编辑模式，修改“/etc/initramfs-tools/modules”文件，添加xen-pv以及virtio的驱动（具体格式要根据操作系统本身的要求来决定）。

```
[root@CTU10000xxxx ~]#vi /etc/initramfs-tools/modules
.....
# Examples:
#
# raid1
# sd_mOd
xen-blkfront
xen-netfront
virtio_blk
virtio_scsi
virtio_net
virtio_pci
virtio_ring
virtio
```
3. 按“Esc”后，输入:wq，按“Enter”。保存设置并退出“/etc/initramfs-tools/modules”文件。
4. 执行以下命令，重新生成initrd。
update-initramfs -u
5. 执行以下命令，检查是否已经成功装载了原生的XEN和KVM驱动相应模块。

```
lsinitramfs /boot/initrd.img-`uname -r` |grep xen
```

```
lsinitramfs /boot/initrd.img-`uname -r` |grep virtio
```

```
[root@ CTU10000xxxx home]# lsinitramfs /boot/initrd.img-`uname -r` |grep xen
lib/modules/3.5.0-23-generic/kernel/drivers/net/ethernet/qlogic/netxen
lib/modules/3.5.0-23-generic/kernel/drivers/net/ethernet/qlogic/netxen/netxen_nic.ko
lib/modules/3.5.0-23-generic/kernel/drivers/net/xen-netback
lib/modules/3.5.0-23-generic/kernel/drivers/net/xen-netback/xen-netback.ko
lib/modules/3.5.0-23-generic/kernel/drivers/block/xen-blkback
lib/modules/3.5.0-23-generic/kernel/drivers/block/xen-blkback/xen-blkback.ko
```

```
[root@ CTU10000xxxx home]# lsinitramfs /boot/initrd.img-`uname -r` |grep virtio
lib/modules/3.5.0-23-generic/kernel/drivers/scsi/virtio_scsi.ko
```

📖 说明

如果误将built-in形式存在内核中的驱动添加到initrd或initramfs文件中，不会影响云服务器器正常使用，这里全写进去只是为了修改方便，但是使用lsinitrd命令无法检查到。可使用如下方法确定这些驱动是否以built-in形式存在内核中，例如：

```
[root@ CTU10000xxxxx home]# cat /boot/config-`uname -r` | grep CONFIG_VIRTIO | grep y
CONFIG_VIRTIO_BLK=y
CONFIG_VIRTIO_NET=y
CONFIG_VIRTIO=y
CONFIG_VIRTIO_RING=y
CONFIG_VIRTIO_PCI=y
CONFIG_VIRTIO_MMIO_CMDLINE_DEVICES=y
[root@ CTU10000xxxxx home]# cat /boot/config-`uname -r` | grep CONFIG_XEN | grep y
CONFIG_XEN_BLKDEV_FRONTEND=y
CONFIG_XEN_NETDEV_FRONTEND=y
```

SUSE/openSUSE 系列操作系统相关操作

当操作系统版本低于SUSE 12 SP1或低于openSUSE 13时，请修改“/etc/sysconfig/kernel”文件，操作步骤请参考[第一种情况](#)。

当操作系统版本为SUSE 12 SP1时，修改“/etc/dracut.conf”文件，添加xen-pv以及virtio的驱动，操作步骤请参考[第二种情况](#)。

当操作系统版本高于SUSE 12 SP1或高于openSUSE 13版本时，修改“/etc/dracut.conf”文件，在add_drivers项中添加xen-pv和virtio的驱动，操作步骤请参考[第三种情况](#)。

- 当操作系统版本低于SUSE 12 SP1或低于openSUSE 13时，操作步骤如下。

📖 说明

SUSE系列操作系统首先应确认OS是否已经安装了xen-kmp包（xen-pv的驱动包），执行如下命令：

```
rpm -qa |grep xen-kmp
```

回显类似如下：

```
xen-kmp-default-4.2.2_04_3.0.76_0.11-0.7.5
```

如果没有安装xen-kmp的包，请到安装ISO中获取并安装。

- a. 执行如下命令，修改“/etc/sysconfig/kernel”文件。

```
vi /etc/sysconfig/kernel
```

- b. 在INITRD_MODULES=""中添加xen-pv以及virtio的驱动（具体格式要根据OS本身的要求来决定）。

```
SIA10000xxxxx:~ # vi /etc/sysconfig/kernel
# (like drivers for scsi-controllers, for lvm or reiserfs)
#
INITRD_MODULES="ata_piix ata_generic xen_vnif xen_vbd xen_platform_pci virtio_blk
virtio_scsi virtio_net virtio_pci virtio_ring virtio"
```

- c. 执行mkinitrd命令，重新生成initrd。

📖 说明

如果引导的虚拟文件系统不是默认的initramfs或者initrd，则命令为：**dracut -f** 实际使用的initramfs或者initrd文件名。“实际使用的initramfs或者initrd文件名”可在menu.lst或者grub.cfg配置（“/boot/grub/menu.lst”或“/boot/grub/grub.cfg”或“/boot/grub2/grub.cfg”）中获取。

以SUSE 11 SP4为例，如下所示：

```
default 0
timeout 10
```

```
gfxmenu (hd0,0)/boot/message
title sles11sp4_001_[_VMX_]
root (hd0,0)
kernel /boot/linux.vmx vga=0x314 splash=silent console=ttyS0,115200n8 console=tty0
net.ifnames=0 NON_PERSISTENT_DEVICE_NAMES=1 showopts
initrd /boot/initrd.vmx
title Failsafe_sles11sp4_001_[_VMX_]
root (hd0,0)
kernel /boot/linux.vmx vga=0x314 splash=silent ide=nodma apm=off noresume edd=off
powersaved=off nohz=off highres=off processor.max+cstate=1 nomodeset x11failsafe
console=ttyS0,115200n8 console=tty0 net.ifnames=0 NON_PERSISTENT_DEVICE_NAMES=1
showopts
initrd /boot/initrd.vmx
```

其中，initrd所在行的/boot/initrd.vmx为实际使用的initrd文件，执行的时候请按照**dracut -f /boot/initrd.vmx**执行。如果initrd所在行的initrd文件不包含/boot目录，如/initramfs-xxx，请在执行dracut命令时增加boot目录，例如：**dracut -f /boot/initramfs-xxx**。

- d. 执行如下命令，检查是否已经成功装载了XEN的PVOPS或者KVM的virtio相应模块。

```
lsinitrd /boot/initrd-`uname -r` | grep xen
```

```
lsinitrd /boot/initrd-`uname -r` | grep virtio
```

```
SIA10000xxxx:~ # lsinitrd /boot/initrd-`uname -r` | grep xen
-rwxr--r-- 1 root root 42400 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/block/
xen-blkfront.ko
-rwxr--r-- 1 root root 44200 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/net/
xen-netfront.ko
```

```
SIA10000xxxx:~ # lsinitrd /boot/initrd-`uname -r` | grep virtio
-rwxr--r-- 1 root root 19248 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/scsi/
virtio_scsi.ko
-rwxr--r-- 1 root root 23856 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/block/
virtio_blk.ko
drwxr-xr-x 2 root root 0 Jul 12 14:53 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio
-rwxr--r-- 1 root root 15848 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio/
virtio_ring.ko
-rwxr--r-- 1 root root 20008 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio/
virtio_pci.ko
-rwxr--r-- 1 root root 12272 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio/
virtio.ko
-rwxr--r-- 1 root root 38208 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/net/
virtio_net.ko
```

- e. 重启云服务器。
- f. 重启完毕后，修改“/boot/grub/menu.lst”文件，增加xen_platform_pci.dev_unplug=all和修改root的配置。

修改前如下所示：

```
###Don't change this comment -YaST2 identifier: Original name: linux###
title SUSE Linux Enterprise Server 11SP4 - 3.0.76-0.11 (default)
root (hd0,0)
kernel /boot/vmlinuz-3.0.76-0.11-default root=UUID=4eb40294-4c6f-4384-bbb6-
b8795bbb1130 splash=silentcrashkernel=256M-:128M showopts vga=0x314
initrd /boot/initrd-3.0.76-0.11-default
```

修改后如下所示：

```
###Don't change this comment -YaST2 identifier: Original name: linux###
title SUSE Linux Enterprise Server 11SP4 - 3.0.76-0.11 (default)
root (hd0,0)
kernel /boot/vmlinuz-3.0.76-0.11-default root=UUID=4eb40294-4c6f-4384-bbb6-
b8795bbb1130 splash=silentcrashkernel=256M-:128M showopts vga=0x314
xen_platform_pci.dev_unplug=all
initrd /boot/initrd-3.0.76-0.11-default
```

📖 说明

- 确保磁盘root分区为UUID的表示形式。
 - xen_platform_pci.dev_unplug=all该参数的添加是为了屏蔽qemu设备。
 - SUSE 11 SP1 64bit ~ SUSE 11 SP4 64bit系统需要在“menu.lst”文件添加 xen_platform_pci.dev_unplug=all，SUSE 12以后版本默认启用此功能，无需配置。
- g. 执行如下命令确认initrd中是否存在XEN驱动。

lsinitrd /boot/initrd-`uname -r` | grep xen

lsinitrd /boot/initrd-`uname -r` | grep virtio

```
SIA10000xxxx:~ # lsinitrd /boot/initrd-`uname -r` | grep xen
-rwxr--r-- 1 root root 42400 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/block/
xen-blkfront.ko
-rwxr--r-- 1 root root 44200 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/net/
xen-netfront.ko

SIA10000xxxx:~ # lsinitrd /boot/initrd-`uname -r` | grep virtio
-rwxr--r-- 1 root root 19248 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/scsi/
virtio_scsi.ko
-rwxr--r-- 1 root root 23856 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/block/
virtio_blk.ko
drwxr-xr-x 2 root root 0 Jul 12 14:53 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio
-rwxr--r-- 1 root root 15848 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio/
virtio_ring.ko
-rwxr--r-- 1 root root 20008 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio/
virtio_pci.ko
-rwxr--r-- 1 root root 12272 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/virtio/
virtio.ko
-rwxr--r-- 1 root root 38208 Jun 22 2012 lib/modules/2.6.32-279.el6.x86_64/kernel/drivers/net/
virtio_net.ko
```

📖 说明

如果误将built-in形式存在内核中的驱动添加到initrd或initramfs文件中，不会影响云服务器正常使用，这里全写进去只是为了修改方便，但是使用lsinitrd命令无法检查到。可使用如下方法确定这些驱动是否以built-in形式存在内核中，例如：

cat /boot/config-`uname -r` | grep CONFIG_VIRTIO | grep y

cat /boot/config-`uname -r` | grep CONFIG_XEN | grep y

- 当操作系统版本为SUSE 12 SP1时，操作步骤如下。
 - a. 执行以下命令，打开“/etc/dracut.conf”文件。

vi /etc/dracut.conf
 - b. 按“i”进入编辑模式，在“add-drivers”项中添加xen-pv和virtio的驱动（具体格式要根据操作系统本身的要求来决定）。

```
[root@CTU10000xxxx ~]# vi /etc/dracut.conf
# additional kernel modules to the default
add_drivers+="ata_piix ata_generic xen_vnif xen_vbd xen_platform_pci virtio_blk virtio_scsi
virtio_net virtio_pci virtio_ring virtio"
```
 - c. 按“Esc”后，输入:wq，按“Enter”。保存设置并退出“/etc/dracut.conf”文件。
 - d. 执行以下命令，重新生成initrd。

dracut -f /boot/initramfs-文件名

如果引导的虚拟文件系统不是默认的initramfs，则命令为：**dracut -f 实际使用的initramfs文件名或者initrd文件名**。“实际使用的initramfs文件名或者initrd文件名”可在grub.cfg配置（“/boot/grub/grub.cfg”或“/boot/grub2/grub.cfg”或“/boot/grub/grub.conf”，具体路径根据OS不同会有所不同）中获取。

- e. 如果引导的虚拟文件系统是initramfs，执行以下命令，检查是否已经成功装载了原生的XEN和KVM驱动相应模块。

```
lsinitrd /boot/initramfs-`uname -r`.img | grep xen
```

```
lsinitrd /boot/initramfs-`uname -r`.img | grep virtio
```

如果引导的虚拟文件系统是initrd，执行如下命令，检查是否已经成功装载了原生的XEN和KVM驱动相应模块。

```
lsinitrd /boot/initrd-`uname -r` | grep xen
```

```
lsinitrd /boot/initrd-`uname -r` | grep virtio
```

- 当操作系统版本高于SUSE 12 SP1或高于openSUSE 13版本时，操作步骤如下。以SUSE Linux Enterprise Server 12 SP2 (x86_64)为例，如下所示：

- a. 执行以下命令，打开“/etc/dracut.conf”文件。

```
vi /etc/dracut.conf
```

- b. 按“i”进入编辑模式，在“add_drivers”项中添加xen-pv和virtio的驱动（具体格式要根据操作系统本身的要求来决定）。

```
[root@CTU10000xxxx ~]# vi /etc/dracut.conf
# additional kernel modules to the default
add_drivers+="ata_piix ata_generic xen-blkfront xen-netfront virtio_blk virtio_scsi
virtio_net virtio_pci virtio_ring virtio"
```

- c. 按“Esc”后，输入:wq，按“Enter”。保存设置并退出“/etc/dracut.conf”文件。

- d. 执行以下命令，重新生成initrd。

```
dracut -f /boot/initramfs-文件名
```

如果引导的虚拟文件系统不是默认的initramfs，则命令为：**dracut -f 实际使用的initramfs文件名或者initrd文件名**。“实际使用的initramfs文件名或者initrd文件名”可在grub.cfg配置（“/boot/grub/grub.cfg”或“/boot/grub2/grub.cfg”或“/boot/grub/grub.conf”，具体路径根据OS不同会有所区别）中获取。

- e. 如果引导的虚拟文件系统是initramfs，执行以下命令，检查是否已经成功装载了原生的XEN和KVM驱动相应模块。

```
lsinitrd /boot/initramfs-`uname -r`.img | grep xen
```

```
lsinitrd /boot/initramfs-`uname -r`.img | grep virtio
```

如果引导的虚拟文件系统是initrd，执行如下命令，检查是否已经成功装载了原生的XEN和KVM驱动相应模块。

```
lsinitrd /boot/initrd-`uname -r` | grep xen
```

```
lsinitrd /boot/initrd-`uname -r` | grep virtio
```

以引导的虚拟文件系统是initrd为例，回显信息如下所示：

```
sluo-ecs-30dc:~ # lsinitrd /boot/initrd-`uname -r` | grep xen
-rw-r--r-- 1 root root 69575 Oct 26 2016 lib/modules/4.4.21-69-default/kernel/drivers/block/xen-blkfront.ko
-rw-r--r-- 1 root root 53415 Oct 26 2016 lib/modules/4.4.21-69-default/kernel/drivers/net/xen-netfront.ko
drwxr-xr-x 2 root root 0 Sep 28 10:21 lib/modules/4.4.21-69-default/updates/pvdriver/xen-hcall
-rwxr-xr-x 1 root root 8320 Sep 28 10:21 lib/modules/4.4.21-69-default/updates/pvdriver/xen-hcall/xen-hcall.ko

sluo-ecs-30dc:~ # lsinitrd /boot/initrd-`uname -r` | grep virtio
-rw-r--r-- 1 root root 29335 Oct 26 2016 lib/modules/4.4.21-69-default/kernel/drivers/block/virtio_blk.ko
-rw-r--r-- 1 root root 57007 Oct 26 2016 lib/modules/4.4.21-69-default/kernel/drivers/net/virtio_net.ko
-rw-r--r-- 1 root root 32415 Oct 26 2016 lib/modules/4.4.21-69-default/kernel/drivers/scsi/
```

```
virtio_scsi.ko
drwxr-xr-x 2 root root 0 Sep 28 10:21 lib/modules/4.4.21-69-default/kernel/drivers/virtio
-rw-r--r-- 1 root root 19623 Oct 26 2016 lib/modules/4.4.21-69-default/kernel/drivers/virtio/
virtio.ko
-rw-r--r-- 1 root root 38943 Oct 26 2016 lib/modules/4.4.21-69-default/kernel/drivers/virtio/
virtio_pci.ko
-rw-r--r-- 1 root root 24431 Oct 26 2016 lib/modules/4.4.21-69-default/kernel/drivers/virtio/
virtio_ring.ko
```

说明

如果误将built-in形式存在内核中的驱动添加到initrd或initramfs文件中，不会影响云服务器正常使用，这里全写进去只是为了修改方便，但是使用lsinitrd命令无法检查到。可使用如下方法确定这些驱动是否以built-in形式存在内核中，例如：

```
cat /boot/config-`uname -r` | grep CONFIG_VIRTIO | grep y
cat /boot/config-`uname -r` | grep CONFIG_XEN | grep y
```