

裸金属服务器

私有镜像制作指南

文档版本 01
发布日期 2025-11-07



版权所有 © 华为技术有限公司 2025。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公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为公司对本文档内容不做任何明示或暗示的声明或保证。

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

安全声明

漏洞处理流程

华为公司对产品漏洞管理的规定以“漏洞处理流程”为准，该流程的详细内容请参见如下网址：

<https://www.huawei.com/cn/psirt/vul-response-process>

如企业客户须获取漏洞信息，请参见如下网址：

<https://securitybulletin.huawei.com/enterprise/cn/security-advisory>

目录

1 制作镜像简介.....	1
2 制作环境准备.....	6
3 Linux 镜像制作流程.....	8
3.1 软件准备.....	8
3.2 工具准备.....	12
3.3 创建 linux 虚拟机.....	13
3.3.1 使用 virt-manager 工具创建虚拟机.....	13
3.3.1.1 创建虚拟机（Linux）.....	13
3.3.2 使用镜像服务的 ISO 镜像功能创建虚拟机.....	18
3.3.2.1 注册 ISO 文件（Linux）.....	18
3.3.2.2 使用 ISO 镜像创建 Linux 云服务器.....	20
3.4 安装虚拟机.....	20
3.4.1 安装虚拟机说明.....	20
3.4.2 操作系统分区建议.....	21
3.4.3 SUSE 11 SP4.....	23
3.4.4 Ubuntu 18.04/Ubuntu 16.04/Ubuntu 14.04/Debian 8.6.....	26
3.4.5 SUSE 12 系列.....	31
3.4.6 CentOS 7.4 for ARM.....	34
3.5 启动引导文件修改（仅 UEFI 启动场景涉及）.....	38
3.6 虚拟机环境配置.....	40
3.6.1 概述.....	40
3.6.2 安装基础组件包（可选）.....	40
3.6.3 虚拟机网络配置.....	41
3.6.4 设置 systemd 超时时间参数默认值.....	42
3.6.5 关闭防火墙.....	42
3.6.6 升级 wicked 组件（可选）.....	43
3.6.7 关闭虚拟机的网络管理工具（可选）.....	43
3.6.8 删除虚拟机网络管理工具插件（可选）.....	45
3.6.9 删除虚拟机的本地用户（可选）.....	45
3.6.10 修改 DHCP 配置项（可选）.....	45
3.6.11 设置 grub 配置文件超时参数（可选）.....	46
3.6.12 设置句柄链接数为 65535.....	46

3.6.13 上传需要的软件包到虚拟机.....	47
3.7 安装 Cloud-Init.....	49
3.7.1 安装 Cloud-Init 说明.....	49
3.7.2 SUSE/Red Hat/CentOS/Oracle Linux/Ubuntu/Debian 系列.....	49
3.7.3 EulerOS/OpenEuler.....	54
3.8 配置 Cloud-Init.....	55
3.8.1 cloud-init-0.7.5 配置.....	55
3.8.2 cloud-init-0.7.6 配置.....	57
3.8.3 cloud-init-0.7.9 及以上版本配置.....	58
3.9 查看 Cloud-Init 服务状态.....	61
3.9.1 SUSE 11 SP4.....	61
3.9.2 SUSE 12 SP1.....	62
3.9.3 SUSE 12 SP2/SUSE 12 SP3/SUSE 15/Oracle Linux 7 系列/Red Hat 7 系列/CentOS 7 系列/CentOS 8 系列.....	62
3.9.4 EulerOS/OpenEuler.....	63
3.9.5 Red Hat 6 系列/CentOS 6 系列/Oracle Linux 6 系列.....	64
3.9.6 Ubuntu 16.04/Ubuntu 18.04.....	65
3.9.7 Ubuntu 14.04.....	66
3.10 修改引导的硬件设备驱动.....	66
3.11 安装 bms-network-config 软件包.....	74
3.12 安装 network 服务.....	76
3.13 安装 SDI 卡驱动.....	76
3.14 安装 Hi1822 网卡驱动.....	77
3.14.1 标卡驱动（集中式裸金属服务器）.....	77
3.14.2 卸载卡驱动（SDI 2.2 分布式裸金属服务器）.....	78
3.14.3 卸载卡驱动（SDI 3.0 分布式裸金属服务器）.....	79
3.15 安装 IB 驱动（可选）.....	79
3.16 安装 FusionServer/TaiShanServer 服务器 iDriver 驱动.....	81
3.16.1 准备工作.....	81
3.16.2 安装服务器板载网卡驱动.....	82
3.16.3 安装 mpt3 驱动.....	82
3.16.4 安装 megaraid_sas 驱动.....	83
3.17 安装多路径软件（可选）.....	83
3.18 安装一键式重置密码插件.....	84
3.19 安全性配置.....	85
3.19.1 修改 SSH 服务配置项.....	85
3.19.2 修改网络脚本权限（可选）.....	86
3.19.3 修改/etc/motd 配置项（可选）.....	87
3.19.4 修改历史记录配置项.....	87
3.19.5 优化 udev 配置和 Selinux.....	87
3.19.6 卸载 denyhosts.....	88
3.19.7 主机名自动更新问题设置（可选）.....	88
3.19.8 安装常用运维工具（可选）.....	89

3.19.9 设置登录口令有效期（可选） 89

3.20 配置裸金属服务器远程登录..... 90

3.20.1 x86： Oracle Linux 7.3/Oracle Linux 7.4/Red Hat 7 系列/CentOS 7.2/CentOS 7.4/CentOS 7.5/CentOS 7.6..... 90

3.20.2 x86： Oracle Linux 6 系列/Red Hat 6 系列/CentOS 6 系列..... 92

3.20.3 x86： SUSE 11 SP4..... 92

3.20.4 ARM： CentOS 7..... 93

3.20.5 x86： SUSE 12 系列/SUSE 15 系列/CentOS 7.3/EulerOS/OpenEuler/Oracle Linux 7.2..... 93

3.20.6 ARM： EulerOS/OpenEuler..... 93

3.20.7 x86： Ubuntu 16.04/Ubuntu 18.04..... 93

3.20.8 ARM： Ubuntu 16.04、 Ubuntu 18.04..... 94

3.20.9 x86： Ubuntu 14.04/Debian..... 95

3.21 配置根分区自动扩盘..... 95

3.21.1 配置说明..... 95

3.21.2 CentOS 6/RedHat 6 系列..... 95

3.21.3 CentOS 7/RedHat 7/Oracle Linux 7 系列..... 96

3.21.4 CentOS/EulerOS ARM/OpenEuler ARM..... 96

3.21.5 EulerOS/OpenEuler..... 97

3.21.6 Debian..... 97

3.22 昇腾服务器驱动和固件安装升级..... 98

3.22.1 安装与维护..... 98

3.22.1.1 安装前必读..... 98

3.22.1.2 安装前准备..... 98

3.22.1.2.1 获取软件包..... 98

3.22.1.2.2 检验软件包完整性..... 99

3.22.1.2.3 系统要求和环境检查..... 100

3.22.1.2.4 创建运行用户..... 101

3.22.1.3 安装 nvme 驱动..... 103

3.22.1.4 安装驱动和固件..... 103

3.22.1.5 卸载驱动和固件..... 106

3.22.2 升级..... 106

3.22.2.1 升级前必读..... 106

3.22.2.2 升级前准备..... 107

3.22.2.3 升级组件..... 109

3.22.2.3.1 升级昇腾服务器芯片固件..... 110

3.22.2.3.2 升级昇腾服务器芯片驱动..... 110

3.23 清理文件..... 112

4 Windows 镜像制作流程..... 113

4.1 硬件和软件准备..... 113

4.2 安装 v5 服务器驱动（可选） 115

4.2.1 准备工作..... 115

4.2.2 安装 chipset 驱动..... 115

4.2.3 安装 v5 服务器板载网卡驱动（可选）	119
4.2.4 3408/3508 RAID 卡驱动安装（可选）	119
4.3 通过 Dism++为 ISO 文件集成 virtio 驱动.....	120
4.4 ISO 镜像合成.....	121
4.5 创建 windows 虚拟机.....	122
4.5.1 使用 virt-manager 工具创建虚拟机.....	122
4.5.1.1 创建并安装虚拟机（Windows）	122
4.5.2 使用镜像服务的 ISO 镜像功能创建虚拟机.....	126
4.5.2.1 注册 ISO 文件（Windows）	126
4.5.2.2 使用 ISO 镜像创建 Windows 云服务器.....	128
4.5.2.3 安装 Windows 操作系统和 virtio 驱动.....	129
4.6 虚拟机环境配置.....	138
4.7 安装 Cloudbase-Init.....	140
4.8 安装 bms-network-config 软件包（可选）	142
4.9 安装 SDI 卡驱动（可选）	143
4.10 安装一键式重置密码插件（可选）	146
4.11 Windows 时区校准设置.....	147
4.12 Windows 虚拟内存设置.....	148
4.13 Windows 自动更新配置（可选）	149
4.14 SID 配置.....	150
5 关闭虚拟机并获取镜像.....	152
6 镜像格式转换.....	153
7 FAQ.....	154
7.1 如何处理从 Linux 裸金属服务器的 bond0 的 vlan 子接口输出的报文，其源 mac 为 bond0 的 mac 问题？	154
7.2 如何处理 SUSE 12 SP1 操作系统自带的 wicked 模块，其概率性将 bond 端口模式配置不正确，进而导致 IP 链路不通的问题？	155
7.3 如何设置裸金属服务器 CPU 频率调节模式？	155
7.4 如何处理 Windows 裸金属服务器下发后 Cloudbase-init 服务异常退出的问题？	155
7.5 如何处理 cloud-init-local 概率性启动失败导致裸金属服务器不能正确注入数据的问题？	156
7.6 如何激活 Windows 裸金属服务器？	156
7.7 x86 EulerOS 2.3 UDP 报文性能调优.....	157
7.8 如何对软件进行完整性校验？	157
7.9 如何检查 device 是否正常运行.....	158
A 附录.....	159
A.1 配置 SSH 服务.....	159
A.2 配置 vsftpd 服务.....	159
A.3 配置 VNC 服务.....	159
A.4 裸金属服务器规格与驱动对应关系.....	161

1 制作镜像简介

如果您需要使用裸金属服务器私有镜像，可以通过外部镜像文件创建私有镜像。本文档提供了完整的私有镜像制作流程（包括创建虚拟机、为虚拟机安装操作系统、软件、驱动等），并枚举了多种类型的操作系统，指导您完成私有镜像的制作。当然，您还可以根据实际需要选择安装其他软件，定制您的私有镜像。

私有镜像文件制作完成后，需要在云平台上进行注册，具体操作请参见《裸金属服务器用户指南》的“[注册私有镜像](#)”章节。注册成功后，您才可以在申请裸金属服务器时选择到该私有镜像。

制作流程

完整的制作镜像的流程为：

图 1-1 制作流程

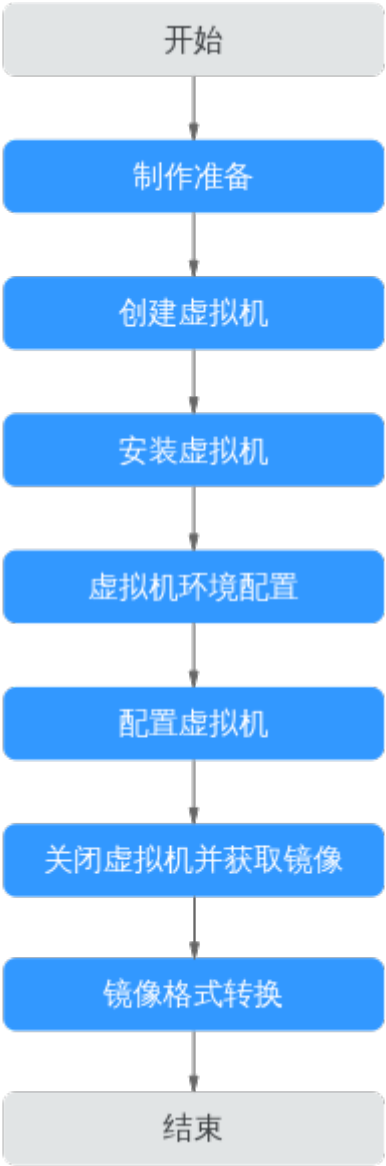


表 1-1 制作流程说明

步骤	说明
制作准备	在制作镜像前，需要准备： • 一台Linux系统的物理服务器或虚拟机作为镜像制作服务器（即宿主机）。 • 相关软件包，如操作系统ISO文件、SDI卡驱动程序软件包、bms-network-config网络配置程序软件包。 • 相关工具，如跨平台远程访问工具、文件传输工具。
创建虚拟机	使用virt-manager创建虚拟机。
安装虚拟机	根据操作系统类型安装对应类型的虚拟机。

步骤	说明
虚拟机环境配置	为虚拟机配置网络环境使其能够连接互联网，上传安装包到虚拟机。
配置虚拟机	Linux操作系统，需要完成如下配置： ● 安装并配置Cloud-Init ● 修改引导的硬件设备驱动 ● 安装bms-network-config软件包 ● （可选）安装SDI卡驱动 ● （可选）安装Hi1822网卡驱动 ● （可选）安装IB驱动 ● （可选）安装v5/TaiShan服务器驱动 ● （可选）安装多路径软件 ● （可选）安装一键式重置密码插件 ● 安全性配置 ● 配置裸金属服务器远程登录 ● 配置根分区自动扩盘 Windows操作系统，需要完成如下配置： ● 安装v5服务器驱动 ● 安装Cloudbase-Init ● 安装bms-network-config软件包 ● （可选）安装SDI卡驱动 ● （可选）安装一键式重置密码插件 ● Windows时区校准设置 ● Windows虚拟内存设置 ● （可选）Windows自动更新配置 ● 配置SID
关闭虚拟机并获取镜像	关闭虚拟机，获取镜像文件。如果生成的镜像文件过大，可以进行压缩。
镜像格式转换	当前版本仅支持zvhd2格式的镜像，所以在获取镜像文件后，需要将镜像格式转换为zvhd2。

操作系统列表

裸金属服务器镜像支持的操作系统如下：

表 1-2 x86 操作系统列表

操作系统类型	操作系统版本	内核版本
RedHat	Red Hat Linux Enterprise 6.5 64bit	2.6.32-431.el6.x86_64
	Red Hat Linux Enterprise 6.7 64bit	2.6.32-573.el6.x86_64
	Red Hat Linux Enterprise 6.8 64bit	2.6.32-642.el6.x86_64
	Red Hat Linux Enterprise 6.9 64bit	2.6.32-696.el6.x86_64
	Red Hat Linux Enterprise 7.2 64bit	3.10.0-327.el7.x86_64
	Red Hat Linux Enterprise 7.3 64bit	3.10.0-514.el7.x86_64
	Red Hat Linux Enterprise 7.4 64bit	3.10.0-693.el7.x86_64
	Red Hat Linux Enterprise 7.5 64bit	3.10.0-862.el7.x86_64
SUSE	SUSE Linux Enterprise Server 11 SP4 64bit	3.0.101-63-default
	SUSE Linux Enterprise Server 12 SP1 64bit	3.12.49-11-default
	SUSE Linux Enterprise Server 12 SP2 64bit	4.4.21-69-default
	SUSE Linux Enterprise Server 12 SP3 64bit	4.4.73-5-default
Oracle Linux	Oracle Linux Server release 6.8 64bit	4.1.12-37.4.1.el6uek.x86_64
	Oracle Linux Server release 6.9 64bit	4.1.12-61.1.28.el6uek.x86_64
	Oracle Linux Server release 7.2 64bit	3.10.0-327.el7.x86_64
	Oracle Linux Server release 7.3 64bit	3.10.0-327.el7.x86_64或 4.1.12-61.1.18.el7uek.x86_64
	Oracle Linux Server release 7.4 64bit	4.1.12-94.3.9.el7uek.x86_64
EulerOS	EulerOS 2.2 64bit	3.10.0-327.44.58.35.x86_64
	EulerOS 2.3 64bit	3.10.0-514.44.5.10.h142.x86_64

操作系统类型	操作系统版本	内核版本
CentOS	CentOS 6.8 64bit	2.6.32-642.el6.x86_64
	CentOS 6.9 64bit	2.6.32-696.el6.x86_64
	CentOS 7.2 64bit	3.10.0-327.el7.x86_64
	CentOS 7.3 64bit	3.10.0-514.el7.x86_64
	CentOS 7.4 64bit	3.10.0-693.el7.x86_64
	CentOS 7.5 64bit	3.10.0-862.el7.x86_64
Ubuntu	Ubuntu 16.04 LTS 64bit	4.4.0-21-generic x86_64
	Ubuntu 14.04 LTS 64bit	3.13.0-24-generic
Debian	Debian 8.6 64bit	3.16.0-4-amd64
Windows	Windows Server 2012 R2 Standard 64bit	-
	Windows Server 2016 Standard 64bit	-

表 1-3 ARM 操作系统列表

操作系统类型	操作系统版本	内核版本
CentOS	CentOS 7.6 ARM	4.14.0-115.el7a.0.1.aarch64
EulerOS	EulerOS 2.8 ARM	4.19.36-vhulk1907.1.0.h475.eulerosv2r8.aarch64

 说明

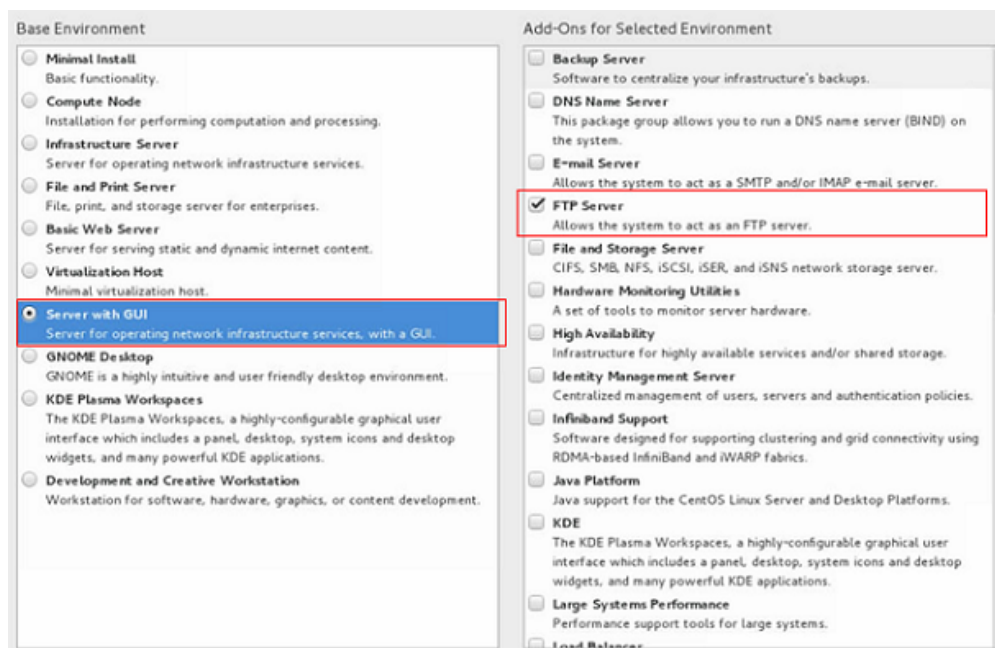
当前SDI驱动仅支持上述内核版本，下载SDI卡驱动时，请确保与内核版本对应。
操作系统内核版本查询命令：**uname -r**。

2 制作环境准备

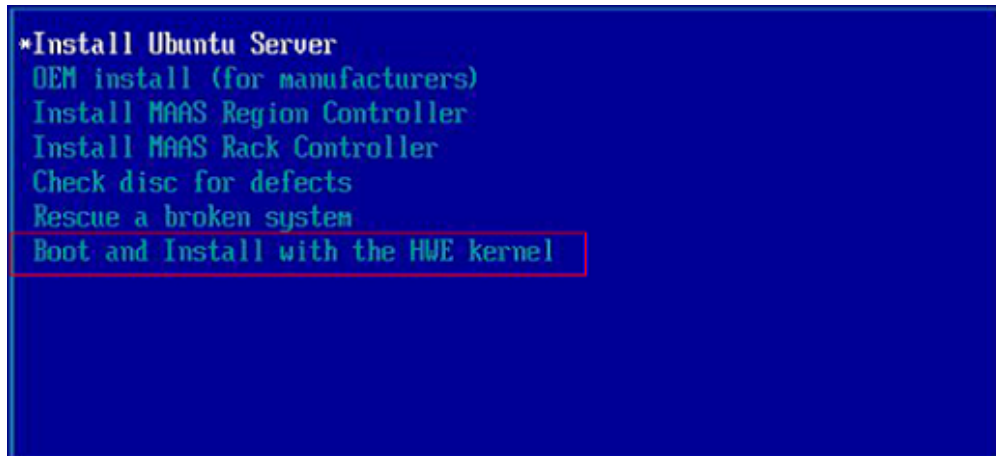
- 需要准备一台Linux系统的物理服务器或虚拟机作为镜像制作服务器，即宿主机，并为宿主机安装Linux操作系统。

以CentOS 7.3 64bit操作系统为例，安装时必须选择安装GUI桌面和虚拟化环境。

选择“Server with GUI”并勾选“FTP Server”、“Virtualization Client”、“Virtualization Hypervisor”及“Virtualization Tools”。



制作ARM64系统镜像时，推荐给宿主机安装Ubuntu 16.04 Server-ARM64系统，安装时，按照下图选择安装HWE kernel。



宿主机安装系统后，需要执行以下指令安装Ubuntu桌面和虚拟化软件，重启生效。

```
sudo apt-get update
```

```
sudo apt-get dist-upgrade
```

```
sudo apt-get install ubuntu-desktop
```

```
sudo apt-get install qemu
```

```
sudo apt-get install qemu-kvm
```

```
sudo apt-get install libvirt0
```

```
sudo apt-get install virt-manager
```

```
sudo apt-get install qemu-efi
```

- 配置宿主机网络和远程连接。
 - a. 宿主机能够连接到互联网，目的是虚拟机通过宿主机可以在线安装软件。
 - b. 通过桌面连接到宿主机。如果使用Windows系统连接宿主机，需要进行相关配置。
- 开启vsftpd、sshd及VNC服务并进行相关配置。

此项配置是为了向宿主机远程传输软件包、镜像ISO、脚本等文件，以及用于远程桌面连接。配置方法参见[配置SSH服务](#)、[配置vsftpd服务](#)及[配置VNC服务](#)。
- CentOS 7.x操作系统需要关闭防火墙，避免无法使用VNC Viewer登录宿主机。命令如下：

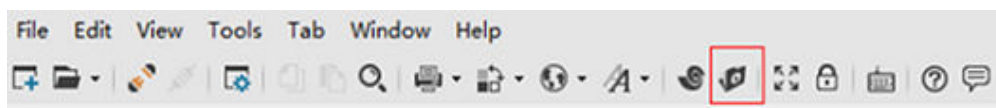
```
systemctl disable firewalld.service
```

```
systemctl stop firewalld.service
```

- 传输需要的文件到宿主机。

配置完成后，通过vsftp服务将虚拟机所需的文件上传到宿主机。至此，宿主机环境配置完成。

采用Xftp工具传输，在用ssh通过Xshell登录宿主机成功后，单击下图所示的图标：



3Linux 镜像制作流程

3.1 软件准备

需要准备[表3-1](#)所示的软件，并且参考[裸金属服务器规格与驱动对应关系](#)确认对应机型/操作系统需要安装的驱动。

表 3-1 软件列表

序号	名称	说明	获取路径	提供方
1	操作系统ISO文件	要制作镜像的ISO文件。 说明 Windows操作系统目前仅支持Standard版本。	建议从各操作系统官方网站获取。	对应OS官方网站

序号	名称	说明	获取路径	提供方
2	SDI卡驱动程序软件包	镜像中安装SDI卡前端驱动插件，使裸金属服务器能够支持挂载云硬盘，并且支持从云硬盘启动，实现快速发放能力。 说明 仅带有存储SDI卡的服务器镜像需要安装该软件包，SDI 2.2和SDI 3.0均无需安装。	linux操作系统： 访问https://support.huawei.com/enterprise/zh/intelligent-accelerator-components/sd100-pid-22040214/software/253495955?idAbsPath=fixnode01%7C23710424%7C251364409%7C21782478%7C22040214，单击下载“SD100-2.0.2.SPC15-DRIVER.zip”软件包并解压，根据镜像OS类型，获取对应的驱动包。比如centos7.6对应的驱动包为“kmod-scsi_ep_front-centos_7.6_1.0.18-3.10.0_957.el7.centos.x86_64.rpm”。 windows操作系统： 访问https://support.huawei.com/enterprise/zh/intelligent-accelerator-components/sd100-pid-22040214/software/250607152?idAbsPath=fixnode01%7C23710424%7C251364409%7C21782478%7C22040214，单击下载“SD100-2.0.2.SPC11-DRIVER-V010.zip”软件包并解压，比如windows2016对应的驱动包为“sdidriver_win-2016-installpack.zip”	华为
3	bms-network-config网络配置程序软件包	用于网络自动化配置。	软件：https://bms-image-package.obs.cn-north-1.myhuaweicloud.com/bms-network-config-23.8.0.zip SHA256校验码：https://bms-image-package.obs.cn-north-1.myhuaweicloud.com/bms-network-config-23.8.0.zip.sha256	华为
4	Cloud-Init初始化工具	安装方式为在线安装，请确保制作镜像的服务器能够连接上互联网。	建议从官方网站获取。	Cloud-Init

序号	名称	说明	获取路径	提供方
5	Cloudbase-Init 初始化工具	Windows密码注入插件，建议下载华为云指定的软件版本。	软件： https://bms-image-package.obs.cn-north-1.myhuaweicloud.com/windows/cloudbase_init/CloudbaseInitSetup_x64.msi SHA256校验码： https://bms-image-package.obs.cn-north-1.myhuaweicloud.com/CloudbaseInitSetup_x64.msi.sha256	Cloudbase-Init
6	Hi1822 网卡驱动	安装Hi1822网卡驱动。	1. 访问 https://support.huawei.com/carrierindex/zh/hwe/index.html ，在首页搜索命名为“uNIC_GuestOS_Driver_BM_xxx.tar.gz”的软件包（“xxx”表示软件包版本），下载最新版本并解压。 2. 在解压后的软件包中，根据镜像的操作系统类型，获取对应操作系统的驱动包。	华为
		Hi1822网卡源码编译驱动包。	访问链接地址 https://support.huawei.com/enterprise/zh/intelligent-accelerator-components/in500-solution-pid-23507369/software ，单击进入最新的版本，下载hinic驱动源码包，上传os内，解压并编译驱动。 例如： 进入“IN500 solution 5.1.0.SPC213”版本，下载软件 hinic-5.5.0.13.zip，解压后，根据 Readme.txt步骤编译驱动和安装。	华为
7	IB网卡驱动	100G Infiniband网卡驱动。	访问 https://network.nvidia.com/products/infiniband-drivers/linux/mlnx_ofed/ ，单击页面正下方的“Download”页签并选择对应操作系统进行下载。	Mellanox
8	FusionServer服务器驱动	FusionServer服务器的板载网卡、RAID卡、磁盘驱动等。	访问 https://support.xfusion.com/support/#/zh/idriver/fusionserver-idriver-pid-21588909/software ，单击进入推荐版本，根据镜像OS类型，下载对应的驱动包。	华为

序号	名称	说明	获取路径	提供方
9	TaiShan服务器驱动	TaiShan服务器的网卡、RAID卡驱动。	访问 https://support.huawei.com/enterprise/zh/management-software/taishanserver-idriver-pid-251215329/software ，单击进入推荐版本，根据镜像OS类型，下载对应的驱动包。	华为
10	UltraPath软件	HBA卡挂载FC存储卷需要的多路径软件。	访问 https://support.huawei.com/enterprise/zh/cloud-storage/ultrapath-pid-8576127/software ，获取对应版本软件及文档。	华为
11	MLNX_OFED驱动	vroce能力依赖OFED驱动能力（指定5.8-3.0.7.0-LTS版本）	访问 https://network.nvidia.com/products/infiniband-drivers/linux/mlnx_ofed/ ，单击页面正下方的“Download”页签，点击“Archive Versions”页签并选择5.8-3.0.7.0-LTS版本，根据对应操作系统进行下载驱动。	Mellanox
	vroce驱动	支持vroce协议的网卡驱动 （使用和hpd版本一致的vroce驱动） 根据驱动链接下载以下4个驱动包以及HPD驱动 hiroce3-*.rpm kmod-hinic3-*.rpm kmod-hisdk3-*.rpm kmod-sssdk-*.rpm （Ubuntu格式包后缀为.deb）	- SDI5.0(kc2, arm机型)：下载arm64格式包 - SDI5.1/5.0(c7e/c7h/AC8, x86机型)：下载amd64格式包 support链接： https://support.huawei.com/carrier/navi?coltype=software#col=software&detailId=PBI1-265311721&path=PBI1-253383977/PBI1-23710112/PBI1-23710137/PBI1-255213015	华为

序号	名称	说明	获取路径	提供方
12	HPD网卡热插拔驱动 (所有涉及sdi5.0、sdi5.1的机型都需要安装)	hotplug-daemon-*.rpm 注意 未安装或者停止hotplug服务，会导致热插拔网卡失败。	- SDI5.1/5.0(arm机型，安装 hotplug-daemon-25.6.11.1-1.aarch64.rpm/hotplug-daemon-25.6.11.1.ubuntu.arm64.deb) - SDI5.1/5.0(x86机型，安装 hotplug-daemon-25.6.11.1-1.x86_64.rpm/hotplug-daemon-25.6.11.1.ubuntu.amd64.deb) support链接： https://support.huawei.com/carrier/navi?coltype=software#col=software&detailId=PBI1-265311721&path=PBI1-253383977/PBI1-23710112/PBI1-23710137/PBI1-255213015	华为

📖 说明

在操作系统安装过程中，请务必按需选择操作系统中所需的软件包，安装的软件越多可能导致最终的镜像文件越大。

裸金属服务器的内存大小最低要求为：比所制作的镜像文件大150MB。请确保所制作的镜像满足此要求。

3.2 工具准备

需要准备的工具如表3-2所示。

📖 说明

qemu-img-hw工具和VNC工具仅在第二种方式（即使用物理机virt-manager工具制作镜像）下需要。

表 3-2 工具列表

名称	说明	获取路径
跨平台远程访问工具	制作镜像过程中，用于访问宿主机的工具，如“Xshell”。	建议从官方网站获取。
文件传输工具	制作镜像过程中，用于为虚拟机传输文件的工具，如“Xftp”。	建议从官方网站获取。

名称	说明	获取路径
qemu-img-hw工具	用于镜像的格式转换。	获取路径： https://cn-south-1-cloud-reset-pwd.obs.cn-south-1.myhuaweicloud.com/imageImportTools/qemu-img-hw.zip
VNC工具	制作镜像过程中，用于登录虚拟机的工具，如“VNC Viewer”。 说明 如果VNC Viewer为5.3.2版本，请在连接页面选择“Options > Expert > ColorLevel”，将ColorLevel值设置为“rgb222”，否则可能连接不上远端操作系统。	建议从官方网站获取。

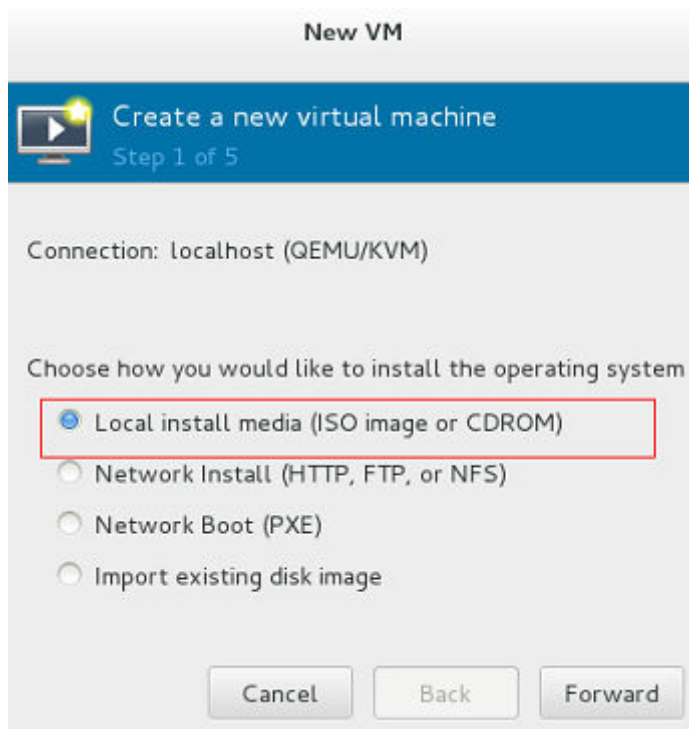
3.3 创建 linux 虚拟机

3.3.1 使用 virt-manager 工具创建虚拟机

3.3.1.1 创建虚拟机（Linux）

使用virt-manager创建虚拟机，以Oracle Linux 6.8虚拟机操作系统为例，其他操作系统类似。以下步骤图示仅供参考，视实际情况而定。

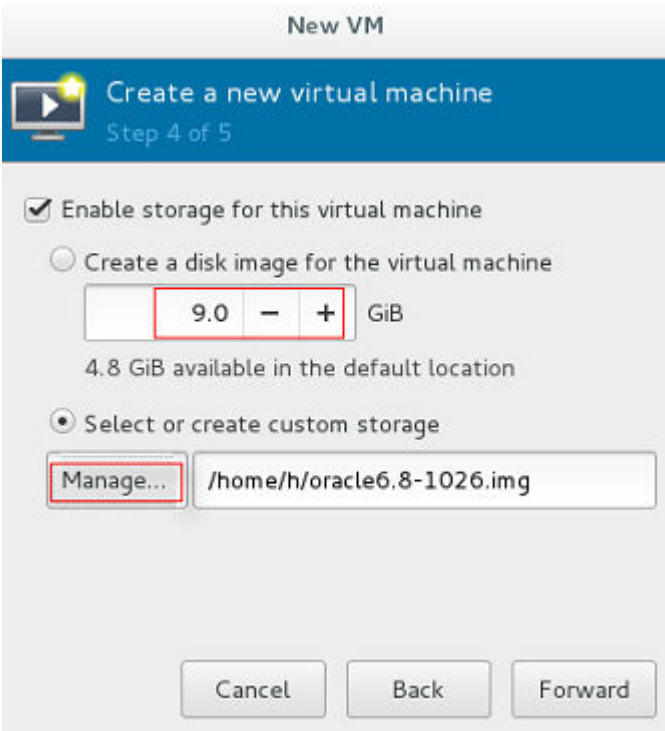
1. 下载所需ISO镜像到宿主机，或者在本地下载后通过Xshell连接宿主机，将ISO文件上传。
2. 登录宿主机，打开终端，输入**virt-manager**命令，弹出virt-manager的软件界面。
3. 新建虚拟机，选择“Local install media (ISO image or CDROM)”。单击“Forward”。



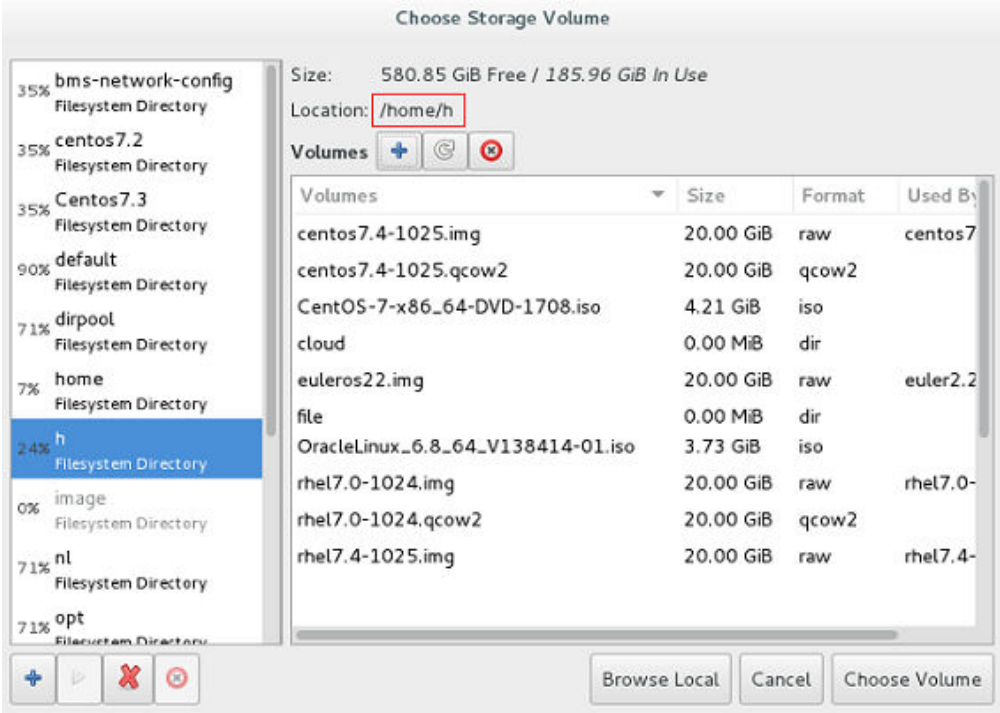
4. 选择ISO镜像，并选择相应的操作系统类型和版本。单击“Forward”。
5. 配置虚拟机内存大小和CPU。单击“Forward”。
 - Memory (RAM): 4096 MiB
 - CPUs: 4
6. 配置创建的磁盘映像大小。

说明

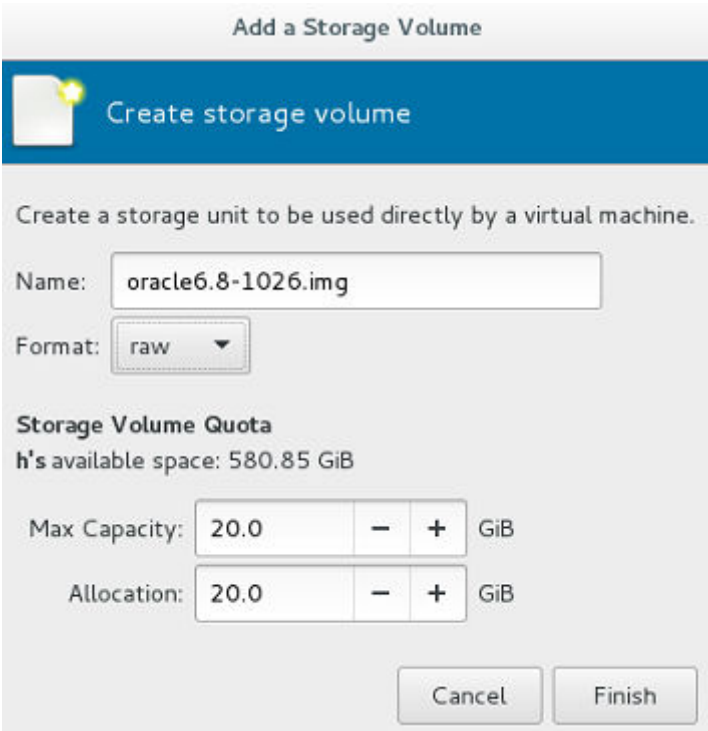
- 磁盘映像不能配置太大，避免生成的镜像文件过大。Oracle Linux 7.3/Red Hat 7.3等Linux系统的镜像大小，建议选择在6G以内，甚至更小，只要能满足要求即可。Ubuntu 16.04 for ARM镜像建议选择手动创建，勾选“Select managed or other existing storage”，然后创建150GB大小的qcow2镜像并选择该镜像。
- 镜像文件大小加预留的内存大小（预留内存为150M）不能超过裸金属服务器的物理内存大小，否则会导致裸金属服务器发放失败。



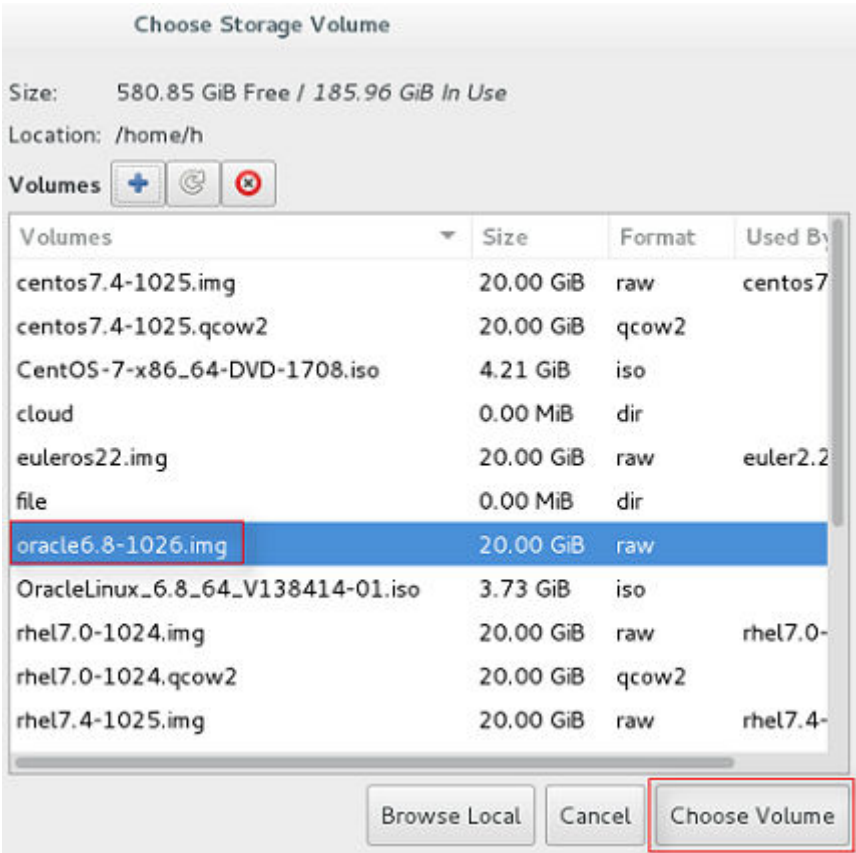
7. 单击上一步的“Manage”，在左侧目录列表选择镜像最终的保存位置，如本例选择“/home/h”。



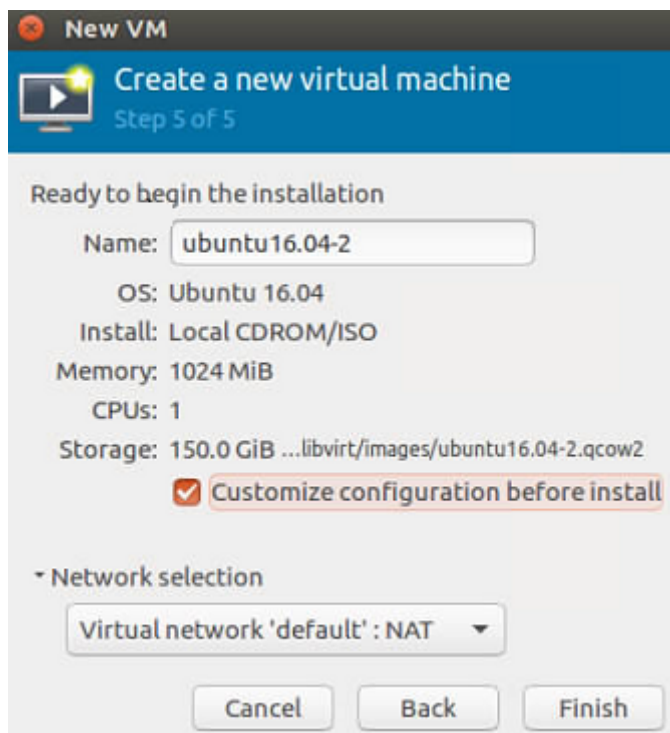
单击 图标创建存储卷，建议Format选择raw，名称可自定义，建议为img格式，方便镜像太大时进行压缩。



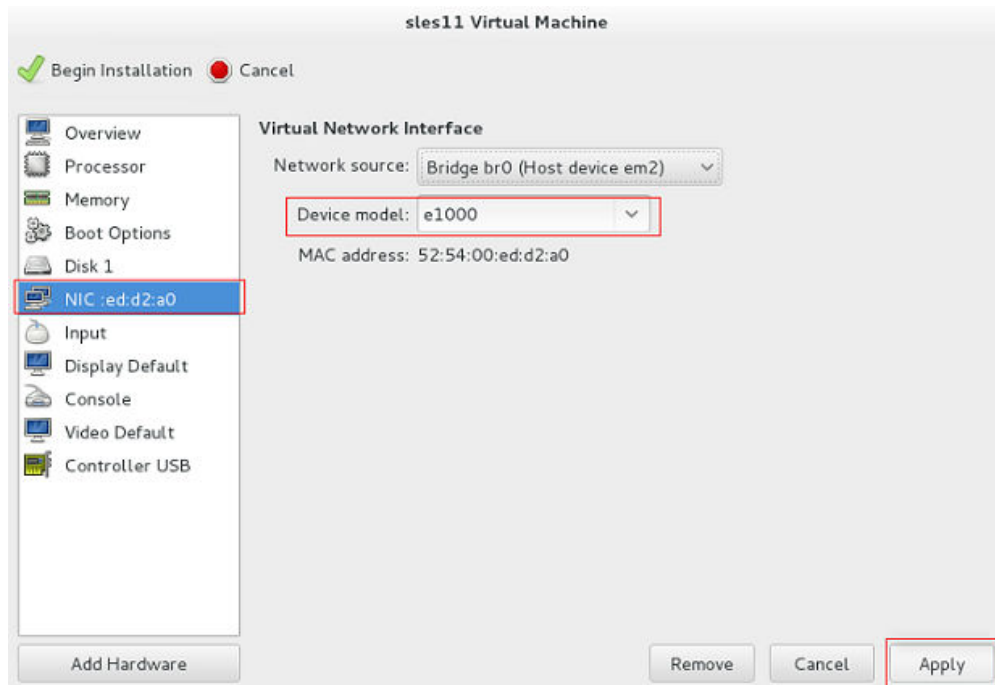
单击“Finish”，返回存储卷列表，选择刚刚创建的卷，单击“Choose Volume”。



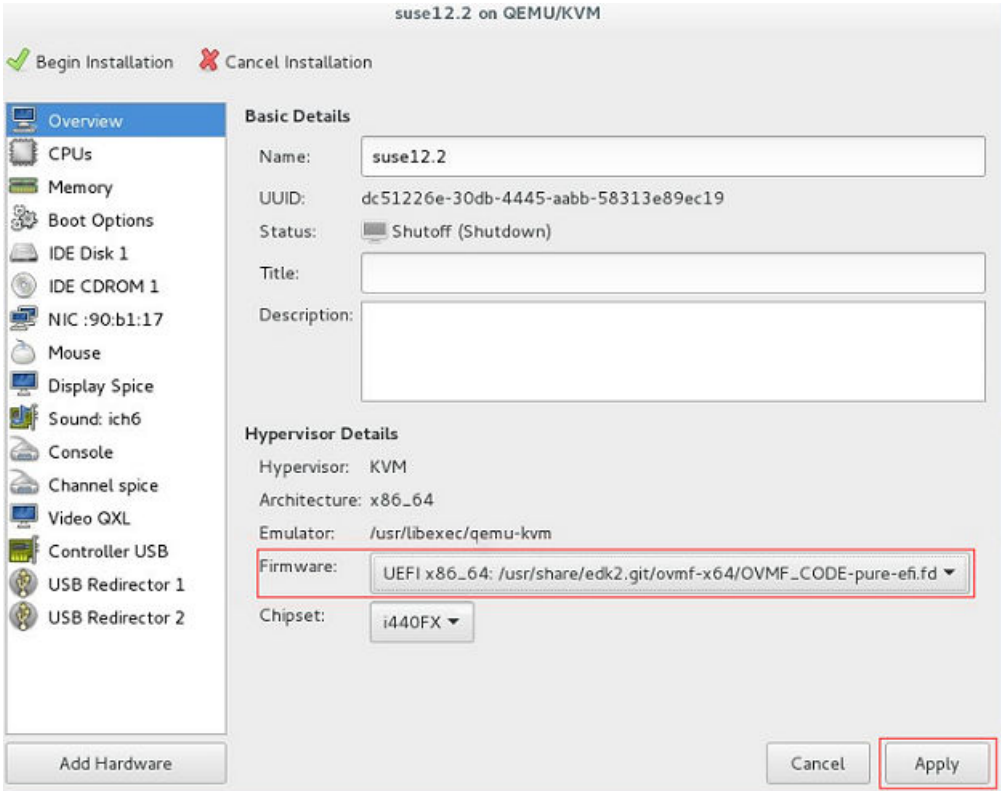
8. 设置虚拟机名称，比如“oracle6.8”，并勾选自定义配置“Customize configuration before install”。Ubuntu 16.04 for ARM在Advanced options选项建议选择NAT类型。单击“Finish”。



9. 如果是Ubuntu 16.04 for ARM镜像，则跳过本步骤。选择网卡配置，选择设备型号为e1000（千兆网卡），单击“Apply”。



10. 如果是SUSE 12镜像，而且是针对HANA解决方案的KunLun服务器，则按照本步骤设置引导方式为UEFI。在左侧导航中选择“Overview”，在右侧区域的“Fireware”选项中选择UEFI方式。



11. 单击“Begin Installation”开始安装。
12. 等待虚拟机启动，然后安装操作系统。按照实际需要配置操作系统语言、时区等。

3.3.2 使用镜像服务的 ISO 镜像功能创建虚拟机

3.3.2.1 注册 ISO 文件（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。

约束与限制

- 如果制作的是鲲鹏服务器镜像，“架构类型”必须选择“ARM”，并且请确保所在Region可以创建鲲鹏云服务器。
- 如果制作的是x86服务器镜像，若用于支持V6 CPU，请设置启动方式为“UEFI”。

前提条件

- 待注册文件必须为iso格式。
- 已上传ISO镜像文件到个人OBS桶中。

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。

操作步骤

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。

3. 单击右上角的“创建私有镜像”，进入创建私有镜像页面。

4. 在“镜像类型和来源”区域，选择镜像的创建方式为“ISO镜像”。

5. 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。

图 3-1 使用 ISO 文件创建私有镜像



6. 在“配置信息”区域，填写镜像的基本信息。

- 启动方式：取值为“BIOS”和“UEFI”。请确保选择的启动方式与对应裸金属服务器机型的启动方式一致，否则，制作出的裸机镜像创建的裸金属服务器无法启动。

- 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，要求不小于镜像文件的系统盘大小。

- 名称：输入镜像的名称。

- 企业项目：选择资源所属的企业项目。

- 标签：可选配置，根据需要为待创建私有镜像添加资源标签。

- 描述：可选配置，根据需要输入描述信息。

7. 单击“立即创建”。

8. 根据界面提示，确认镜像参数。阅读并勾选《镜像制作承诺书》和《华为镜像免责声明》，单击“提交申请”。

9. 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。

3.3.2.2 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。更多信息详见[专属云](#)。

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”，创建弹性云服务器。

由于此云服务器仅作为临时云服务器使用，最终需要删除。因此，系统会默认创建一个固定规格且“按需付费”的云服务器，使用该云服务器创建的私有镜像再次创建云服务器时不会限制规格和付费方式。

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. 根据界面提示完成弹性云服务器的配置，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。

3.4 安装虚拟机

3.4.1 安装虚拟机说明

请根据操作系统类型安装对应类型的虚拟机。由于SUSE 11 SP4、Ubuntu 18.04、Ubuntu 16.04、Ubuntu 14.04、Debian 8.6、SUSE 12系列等操作系统在安装时有一些特殊配置，本章节着重介绍这几种操作系统的安装过程。

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，并根据实际情况完成时区、KMS地址、补丁服务器和Repo源更新地址、输入法、语言等相关配置。

须知

- 裸金属服务器BIOS镜像制作需要使用MBR分区表格式，另外需要为裸金属服务器发放预留一个主分区（Primary Partition），发放后系统会自动生成一个64M的config drive分区。因为MBR格式中主分区和扩展分区总共最多支持4个，故使用的镜像主分区数量不能超过3个，否则会导致裸金属服务器下发失败。
- 如果要使用自动扩盘，根分区必须为最后一个分区，且为主分区。
- 如果业务需要的分区数较多，则需要在扩展分区基础上划分LVM来实现。
- UEFI启动的虚拟机操作系统安装完成后，请勿重启，需要先执行[启动引导文件修改（仅UEFI启动场景涉及）](#)修改启动引导文件，修改完成后再重启。

3.4.2 操作系统分区建议

场景一：裸金属服务器使用 BIOS 启动

如果裸金属服务器使用BIOS启动，则需要BIOS镜像，且采用MBR分区表格式。详细分区建议如下：

- 如果主分区已经够用，

A：需要boot、swap单独分区，选择如下分区方式：

boot-swap-根分区：

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINT
sda	8:0	0	20G	0	disk	
└sda1	8:1	0	500M	0	part	/boot
└sda2	8:2	0	5G	0	part	[SWAP]
└sda3	8:3	0	14.5G	0	part	/

B：不需要boot或swap单独分区，选择如下几种分区方式：

swap-根分区：

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINT
sda	8:0	0	20G	0	disk	
└sda1	8:1	0	5G	0	part	[SWAP]
└sda2	8:2	0	15G	0	part	/

根分区：

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINT
sda	8:0	0	10G	0	disk	
└sda1	8:1	0	10G	0	part	/

- 如果需要使用扩展分区（比如需要使用lvm），则可以选择如下分区方式：
扩展分区（可组lvm）-swap-根分区：

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINT
sda	8:0	0	400G	0	disk	
└─sda1	8:1	0	350G	0	part	
├─euler2.5-var_log_npu	253:0	0	20G	0	lvm	/var/log/npu/oplog
├─euler2.5-var_log_npu_slog	253:1	0	200G	0	lvm	/var/log/npu/slog
├─euler2.5-var_log_npu_hisi_log	253:2	0	20G	0	lvm	/var/log/npu/hisi_log
├─euler2.5-var_log_npu_profiling	253:3	0	100G	0	lvm	/var/log/npu/profiling
├─euler2.5-var_log_npu_dump	253:4	0	10G	0	lvm	/var/log/npu/dump
└─sda2	8:2	0	8G	0	part	[SWAP]
└─sda3	8:3	0	42G	0	part	/

扩展分区（可组lvm）-根分区：

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINT
sda	8:0	0	10G	0	disk	
└─sda1	8:1	0	3G	0	part	
├─euleros2.5-var_log_npu_oplog	253:0	0	1G	0	lvm	/var/log/npu/oplog
├─euleros2.5-var_log_npu_slog	253:1	0	1G	0	lvm	/var/log/npu/slog
├─euleros2.5-var_log_npu_dump	253:2	0	1G	0	lvm	/var/log/npu/dump
└─sda2	8:2	0	7G	0	part	/

boot-扩展分区（可组lvm）-根分区：

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINT
sda	8:0	0	10G	0	disk	
└─sda1	8:1	0	1G	0	part	/boot
└─sda2	8:2	0	3G	0	part	
├─euleros-var_log_npu_oplog	253:0	0	1G	0	lvm	/var/log/npu/oplog
├─euleros-var_log_npu_slog	253:1	0	1G	0	lvm	/var/log/npu/slog
├─euleros-var_log_npu_dump	253:2	0	1G	0	lvm	/var/log/npu/dump
└─sda3	8:3	0	6G	0	part	/

场景二：裸金属服务器使用 UEFI 启动

如果裸金属服务器使用UEFI启动，则需要UEFI镜像。x86裸金属服务器如果使用UEFI启动，则仍然需要采用MBR分区表格式，UEFI要求必须有boot_efi分区，故详细分区建议如下：

- 如果主分区已经够用，
A：需要swap单独分区，选择如下分区方式：
boot_efi-swap-根分区：

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINT
sda	8:0	0	20G	0	disk	
└─sda1	8:1	0	500M	0	part	/boot/efi
└─sda2	8:2	0	5G	0	part	[SWAP]
└─sda3	8:3	0	14.5G	0	part	/

- B：如果不需要swap单独分区，选择如下分区方式：
boot_efi-根分区：

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINT
sda	8:0	0	20G	0	disk	
└─sda1	8:1	0	500M	0	part	/boot/efi
└─sda2	8:2	0	19.5G	0	part	/

- 如果需要使用扩展分区，则使用如下分区方式。
boot_efi-扩展分区（可组lvm）-根分区：

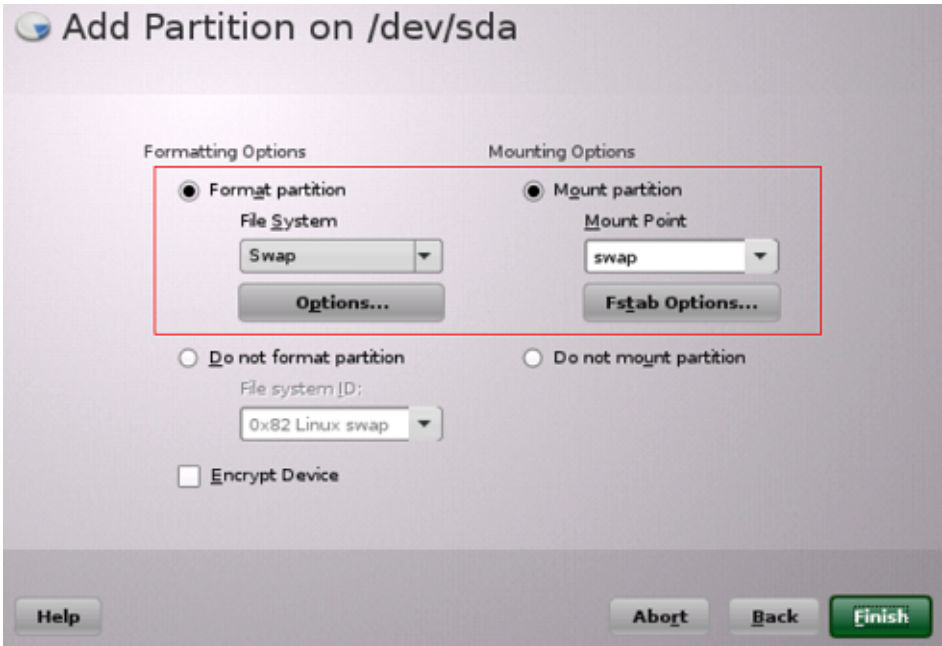
NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINT
sda	8:0	0	10G	0	disk	
├─sda1	8:1	0	1G	0	part	/boot/efi
├─sda2	8:2	0	3G	0	part	
│ └─rhel-var_log_npu_oplog	253:0	0	1G	0	lvm	/var/log/npu/oplog
│ └─rhel-var_log_npu_slog	253:1	0	1G	0	lvm	/var/log/npu/slog
│ └─rhel-var_log_npu_dump	253:2	0	1G	0	lvm	/var/log/npu/dump
└─sda3	8:3	0	6G	0	part	/

3.4.3 SUSE 11 SP4

1. 在安装到 “Installation Settings” 阶段时，切换到 “Expert” 页签，选择 “Change > Partitioning” 创建分区。

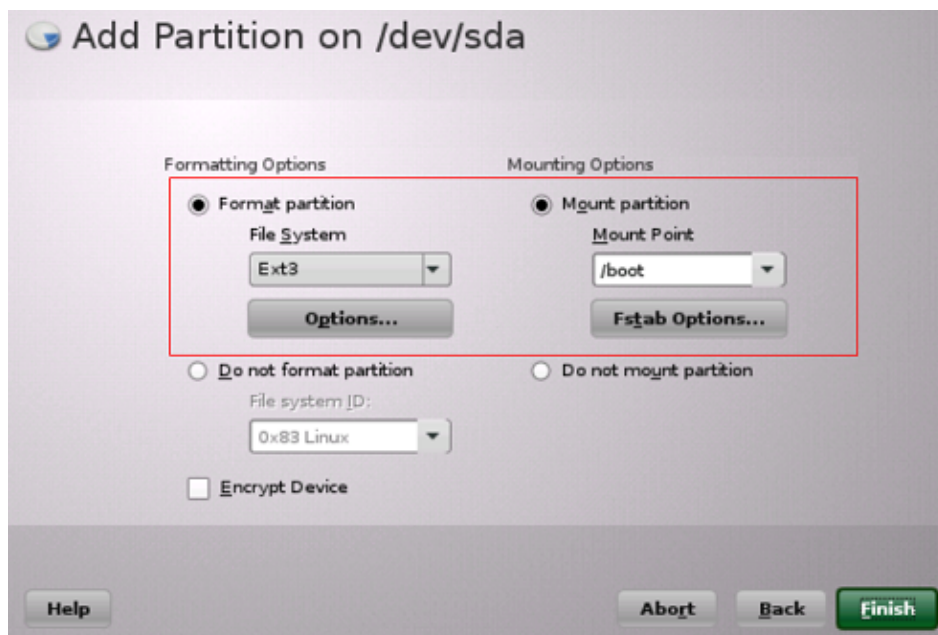
创建Swap分区

- a. 在 “Preparing Hard Disk” 页面，选择 “Custom Partitioning (for experts)”，单击 “Next”。
- b. 在左侧导航中选择 “Hard Disks”，单击 “Add Partition”。
- c. 在 “Add Partition on /dev/sda” 页面，选择 “Primary Partition”，单击 “Next”。
- d. 选择 “Custom Size” 并设置 “Size” 为 “10GB”，单击 “Next”。
- e. 设置 “Formatting Options” 和 “Mounting Options”，单击 “Finish” 完成分区。



创建boot分区

- a. 在左侧导航中选择 “Hard Disks”，单击 “Add Partition”。
- b. 在 “Add Partition on /dev/sda” 页面，选择 “Primary Partition”，单击 “Next”。
- c. 选择 “Custom Size” 并设置 “Size” 为 “5.00GB”，单击 “Next”。
- d. 设置 “Formatting Options” 和 “Mounting Options”，单击 “Finish” 完成分区。

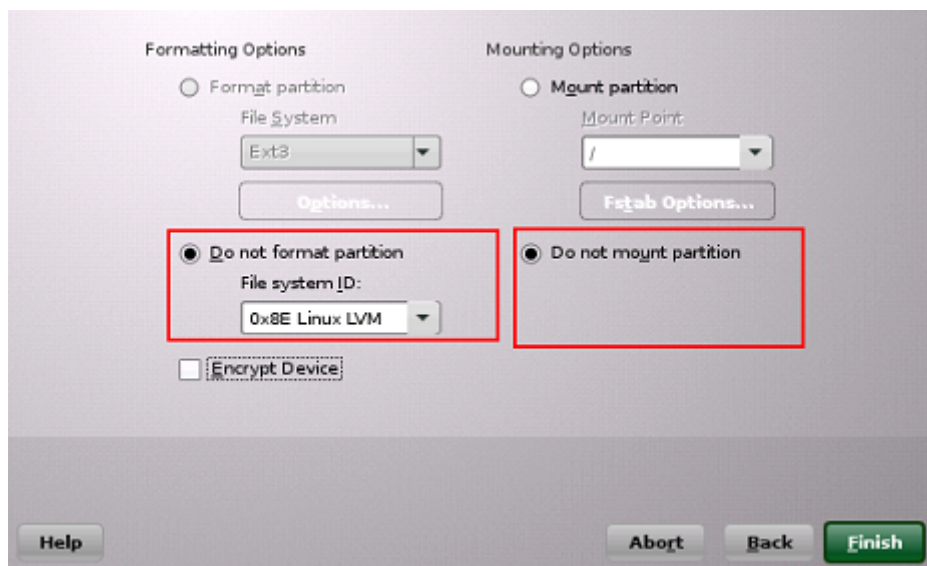


至此，Swap分区和boot分区创建完成。

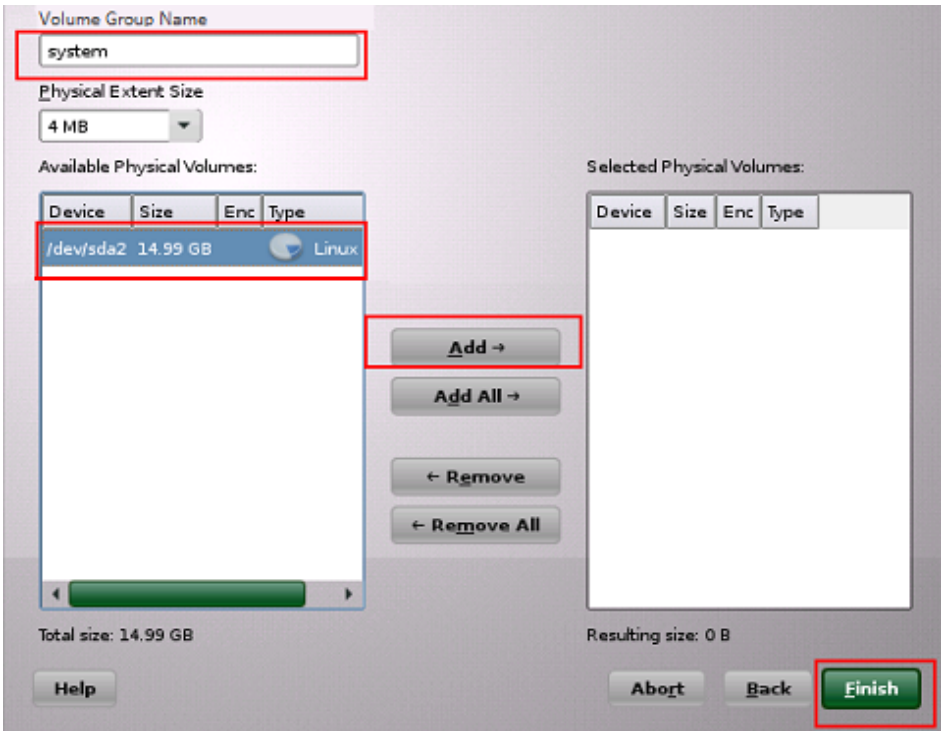
2. 安装设置（创建分卷）。

如果Swap分卷和根分卷均采用LVM模式，则按照下述方法创建分卷：

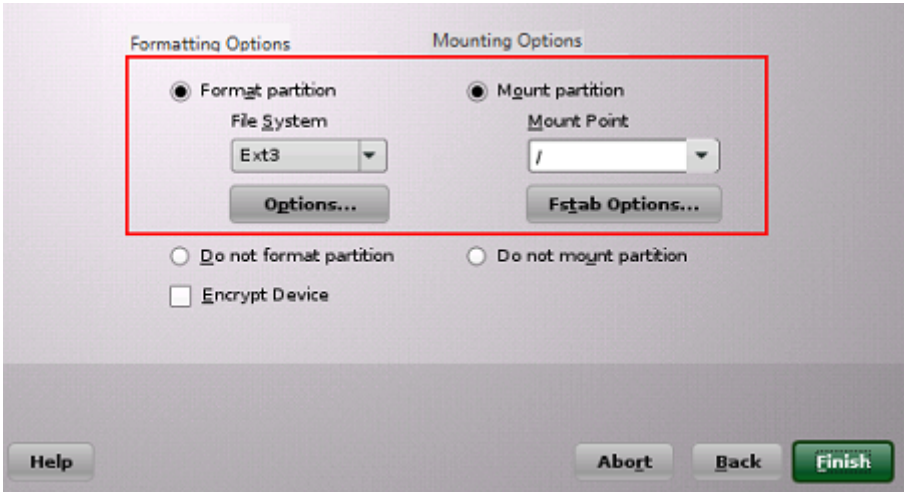
- 在左侧导航中选择“Hard Disks”，单击“Add Partition”。
- 在“Add Partition on /dev/sda”页面，选择“Primary Partition”，单击“Next”。
- 选择“Custom Size”并设置“Size”为“14.99GB”，单击“Next”。
- 设置“Formatting Options”和“Mounting Options”，单击“Finish”。



- 在左侧导航中选择“Volume Management”，单击“Add > Volume Group”添加磁盘组。
- 按如下要求设置相应参数，设置完成后单击“Finish”。



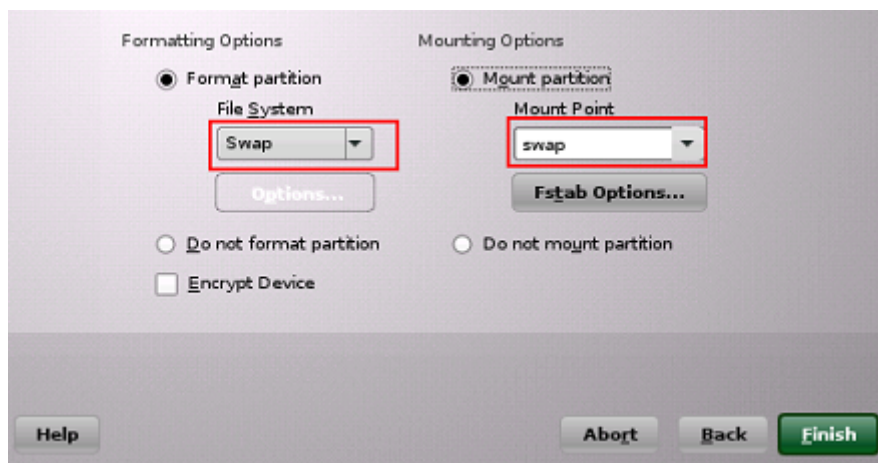
- g. 在左侧导航中选择“Volume Management”，单击“Add > Logical Volume”创建逻辑卷。
- h. 设置“Name”为“root”，“Type”为“Normal Volume”，单击“Next”。
- i. 设置“Custom Size”为“10.00 GB”，单击“Next”。
- j. 设置“Formatting Options”和“Mounting Options”，单击“Finish”完成根分卷的创建。



至此，根分卷已经创建完成，后续步骤介绍如何创建Swap分卷。

- k. 在左侧导航中选择“Volume Management”，单击“Add > Logical Volume”创建逻辑卷。
- l. 设置“Name”为“swap”，“Type”为“Normal Volume”，单击“Next”。
- m. 设置“Size”为“Maximum Size (4.99 GB)”，单击“Next”。

- n. 设置“Formatting Options”和“Mounting Options”，单击“Finish”完成Swap分卷的创建。



单击“Accept”完成配置。

至此，根分卷和Swap分卷已经划分完毕。

3. 返回安装设置界面，查看当前的分区和启动配置。确认无误后，单击“Install”。
4. 在弹出的页面中单击“I Agree”，然后单击“Install”。
5. 设置root用户密码。单击“Next”。
6. 设置Hostname、Domain Name，单击“Next”。
7. 在“Network Configuration”页面，选择“Use Following Configuration”，单击“Next”。
- 在“Test Internet Connection”页面，勾选“No, Skip This Test”跳过网络连通性测试。
8. 在“Network Services Configuration”页面，选择“Use Following Configuration”，单击“Next”。
9. 用户认证方法保持默认配置，单击“Next”。
10. 创建本地用户，比如suse。单击“Next”。

说明

SUSE 11 SP4安装过程要求创建本地用户，在安装完成后，如果不需要该用户，可参考“[虚拟机环境配置](#)”章节的步骤进行删除。

11. 在“Hardware Configuration”页面，选择“Use Following Configuration”，单击“Next”。
12. 单击“Finish”完成安装。

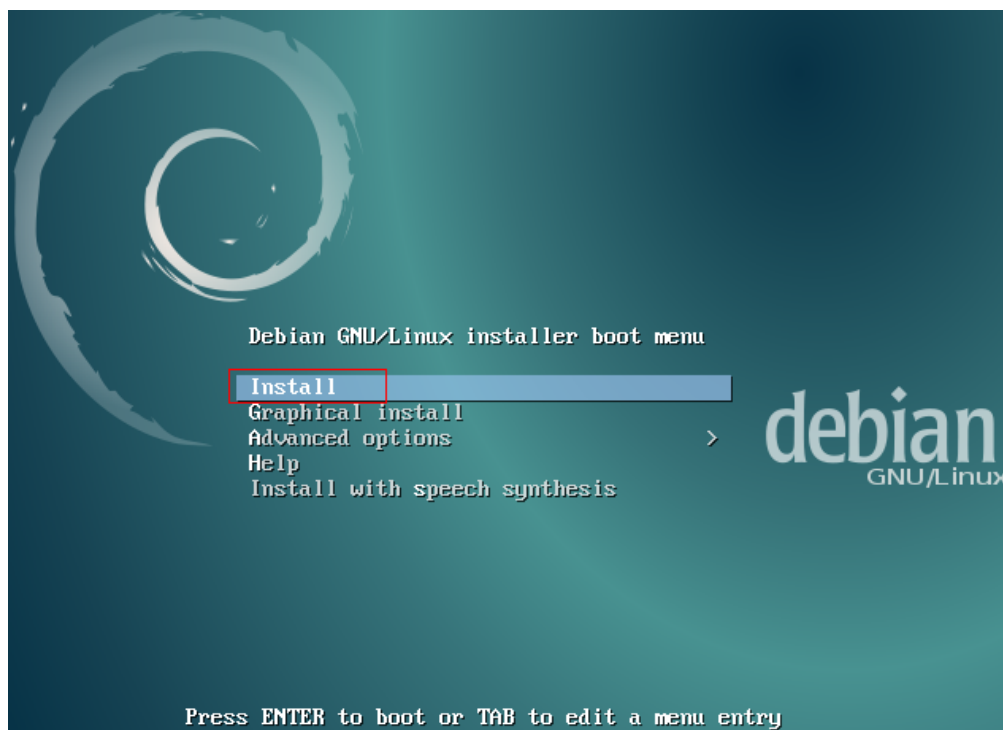
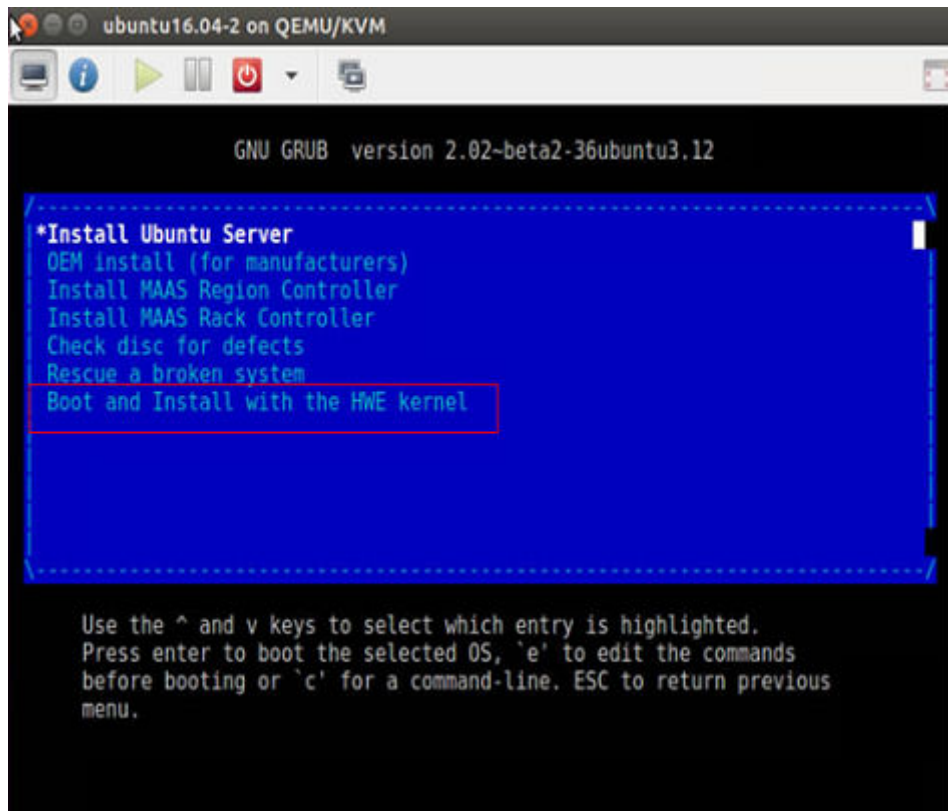
3.4.4 Ubuntu 18.04/Ubuntu 16.04/Ubuntu 14.04/Debian 8.6

注意

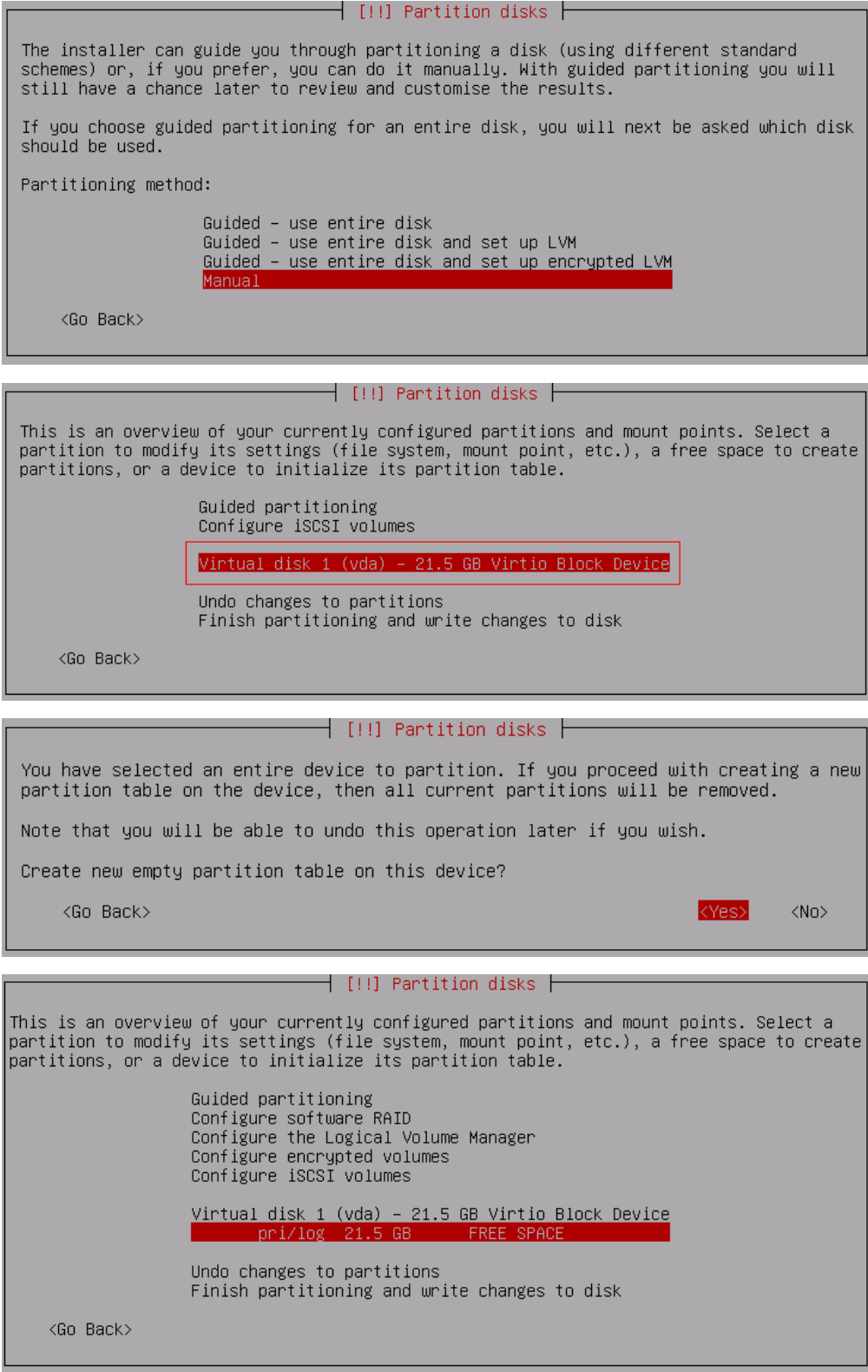
请使用ubuntu-xx.xx.x-server-amd64.iso格式的ubuntu ISO，请勿使用ubuntu-xx.xx.x-live-server-amd64.iso格式的。

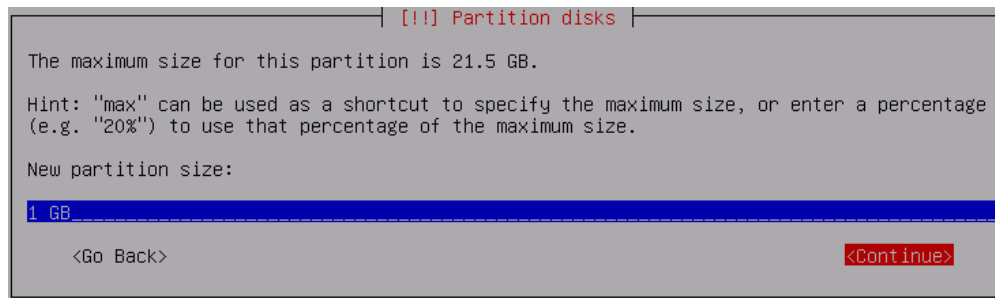
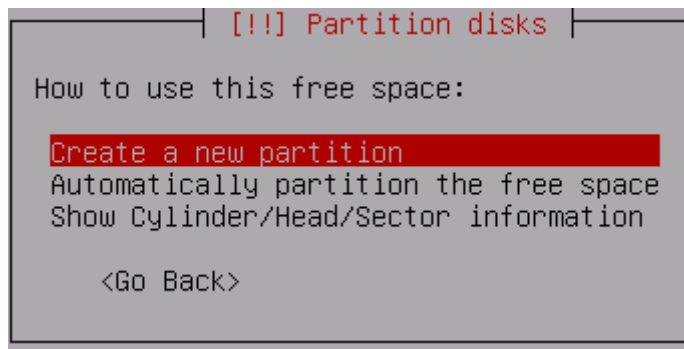
1. Ubuntu 18.04、Ubuntu 16.04 for ARM虚拟机内安装系统时，进入ISO后，按照下图选择“Boot and Install with the HWE kernel”，然后再选择“Install”

Ubuntu Server”开始安装。（制作Ubuntu 18.04、Ubuntu 16.04/Ubuntu 14.04/Debian 8.6镜像时，直接选择“Install”进行安装。）

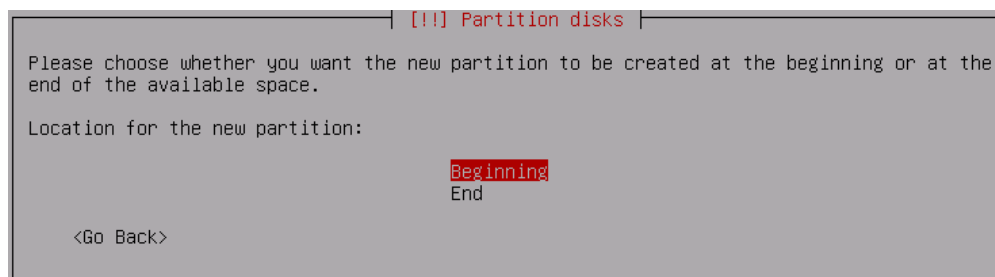
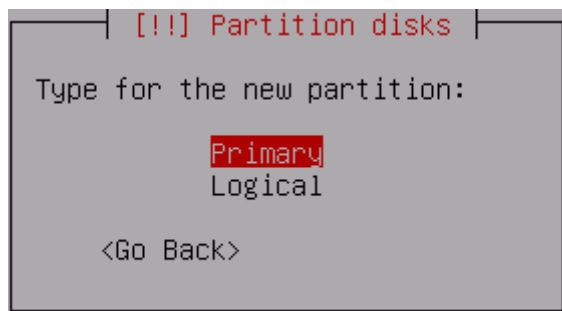


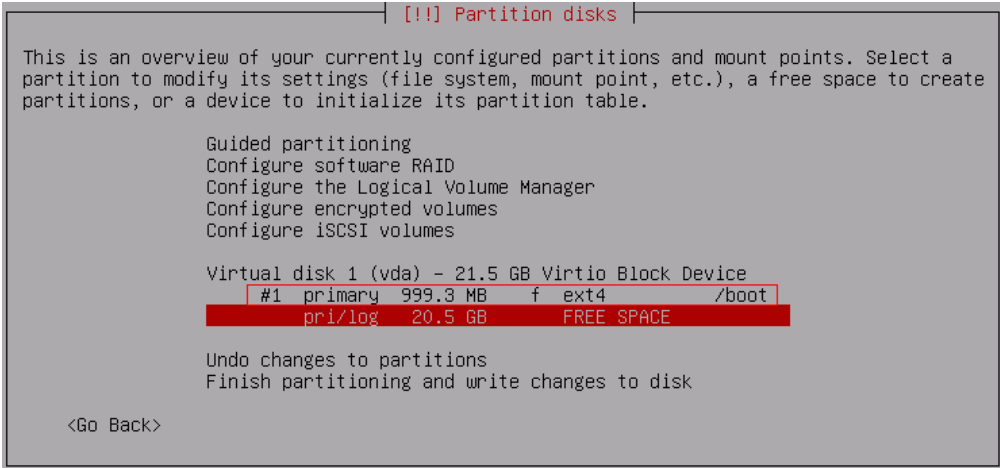
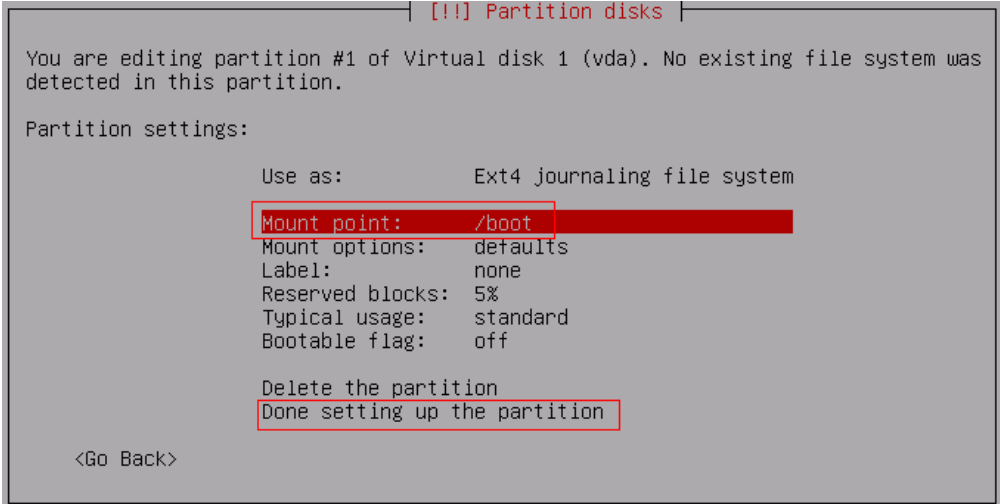
2. 在安装Ubuntu 18.04、Ubuntu 16.04 for ARM镜像的时候，选择自动分区；在安装Ubuntu 18.04、Ubuntu 16.04/Ubuntu 14.04/Debian 8.6镜像的过程中需要进行手动分区，步骤如下：



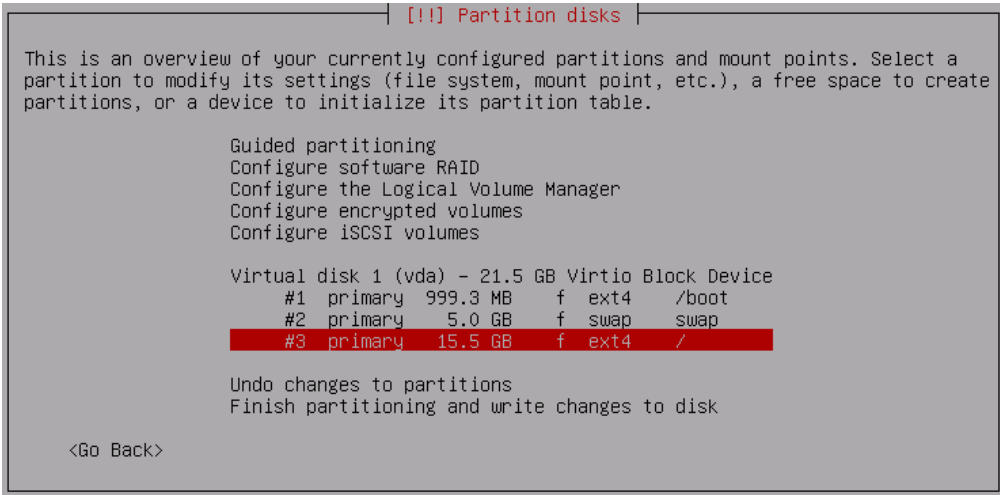


下图新分区类型不能选择逻辑分区，应该选择“Primary”选项。

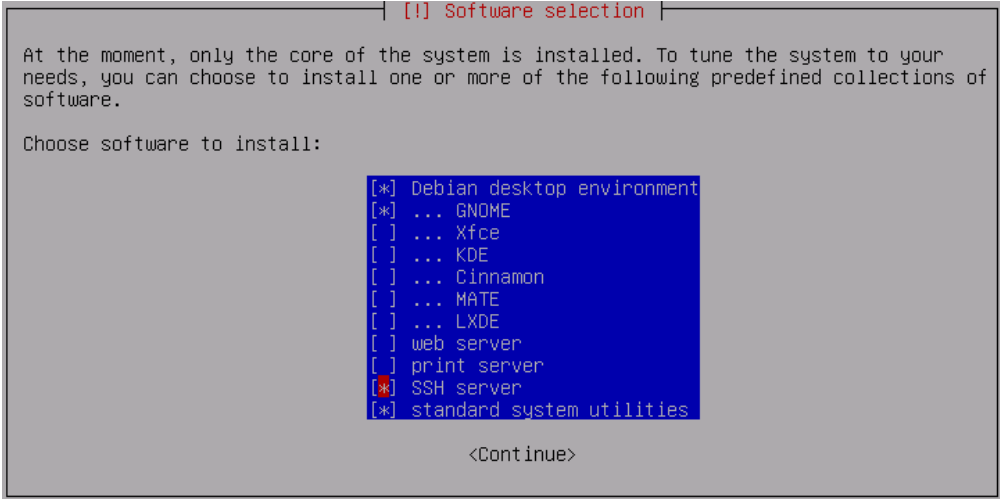




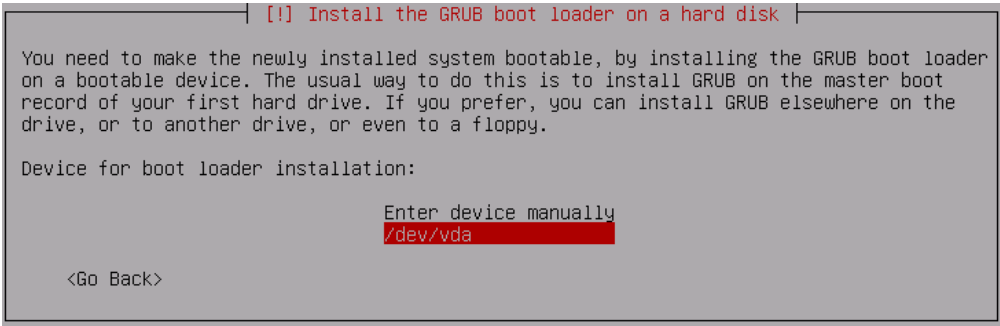
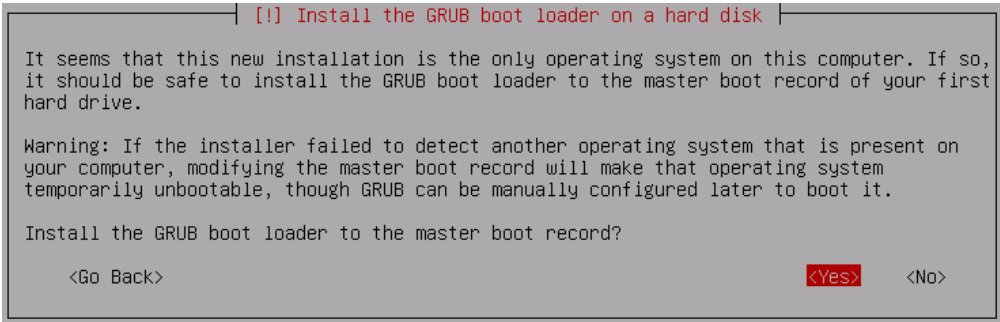
以上为boot分区的方法，下面的swap分区、/分区参考boot分区即可，下面为最终分区结果示意（根分区为最后一个分区）：



安装额外软件：

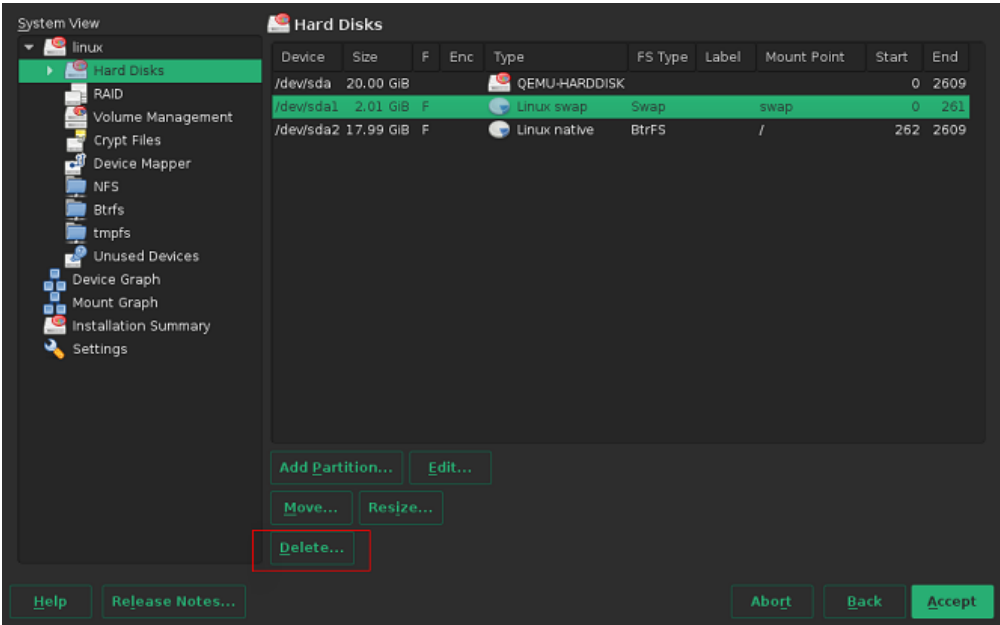


将GRUB安装至硬盘：

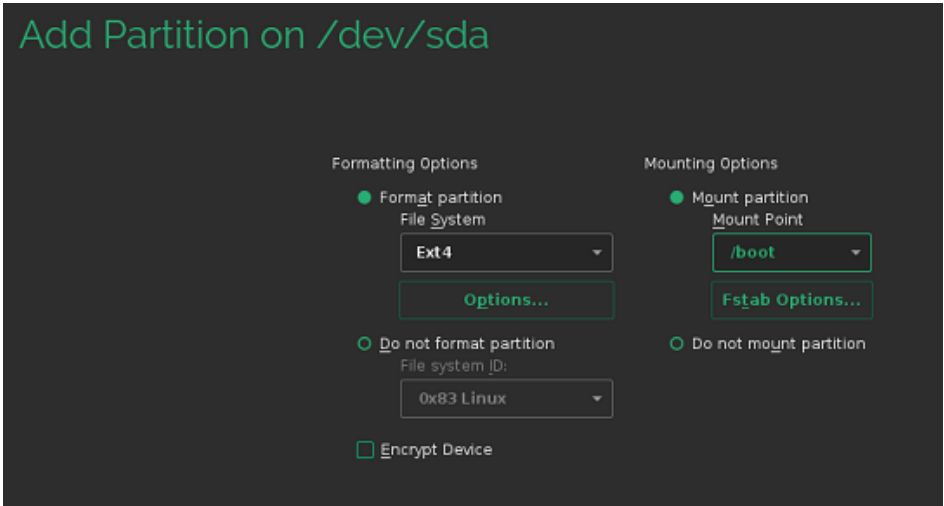


3.4.5 SUSE 12 系列

1. 在安装到磁盘分区时，单击“Expert Partitioner”。
2. 选择下面的默认分区，鼠标右键单击“Delete”进行删除。

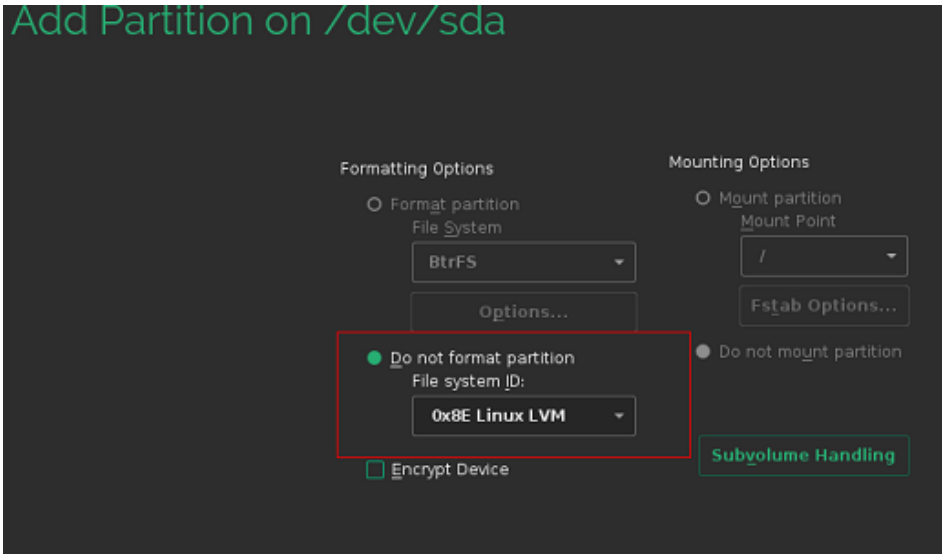


3. 按照实际业务要求进行分区，示例如下：
- a. 在左侧导航中选择“Hard Disks”，单击“Add Partition”，创建分区。
 - b. 选择“Primary Partition”，然后单击“Next”。
 - c. 选择自定义大小（例如500 MiB），如果有要求请按照实际要求设置大小。
 - d. 角色设置为“Operating System”，然后单击“Next”。
 - e. 选择文件系统和挂载点，单击“Finish”。

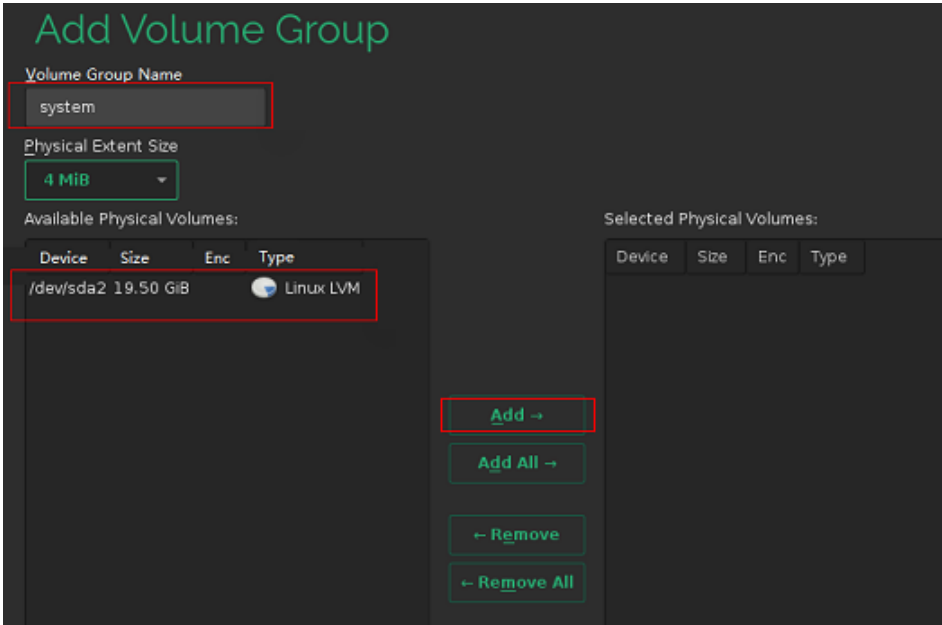


至此，boot分区创建完成。

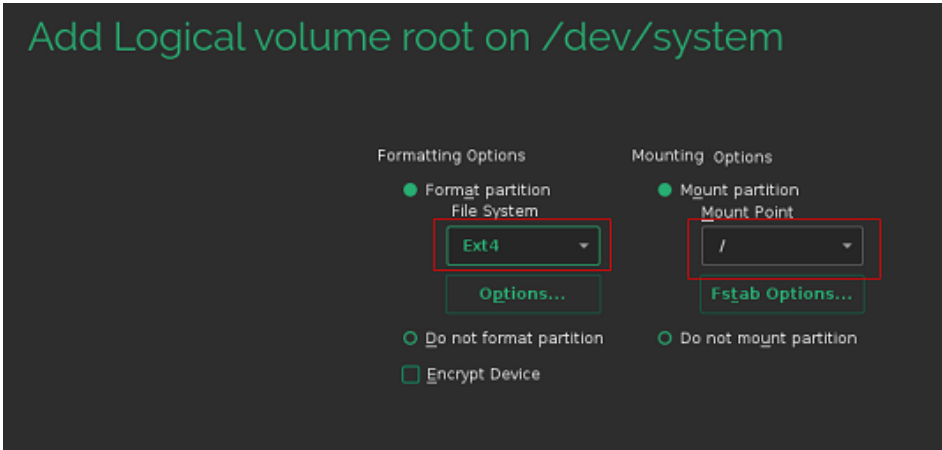
4. 如果Swap分卷和根分卷均采用LVM模式，则按照下述方法创建分卷：
- a. 在左侧导航中选择“Hard Disks”，单击“Add Partition”。
 - b. 选择“Primary Partition”，单击“Next”。
 - c. 选择自定义大小，设置“Size”为“19.50 GB”，单击“Next”。
 - d. 角色选择“Operating System”，单击“Next”。
 - e. 设置“Formatting Options”，单击“Finish”完成分区创建。



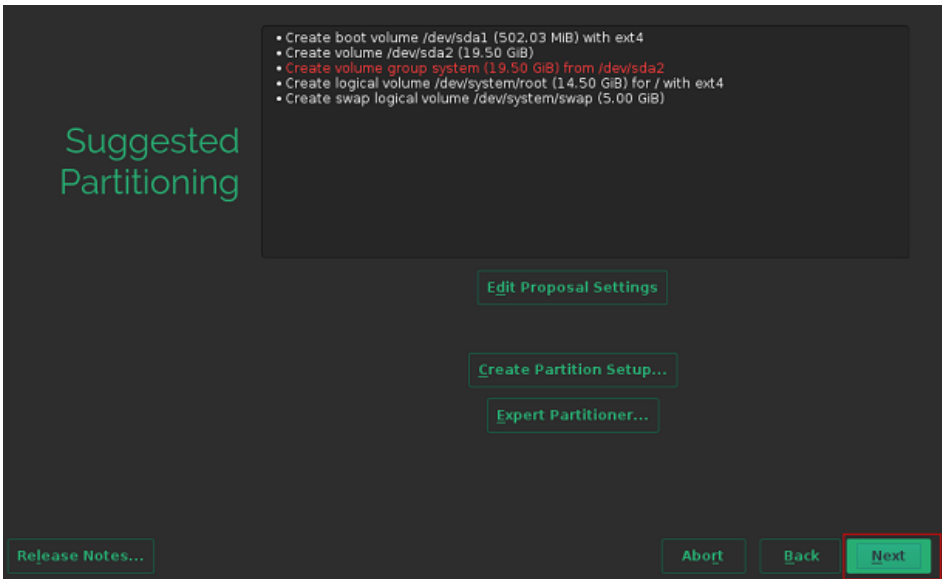
- f. 在左侧导航中选择“Volume Management”，单击“Add > Volume Group”添加卷组。
- g. 按照下图设置相关参数，单击“Add”将可用物理卷加入“Selected Physical Volumes”区域。
设置完成后单击“Finish”完成卷组创建。



- h. 在左侧导航中选择“Volume Management”，单击“Add > Logical Volume”添加Swap逻辑卷。
- i. 设置逻辑卷名称为“swap”，大小为“5 GiB”：
- j. 角色选择“Operating System”。
- k. 设置“Formatting Options”和“Mounting Options”，单击“Finish”完成Swap分卷的创建。
- l. 创建根分卷的方法与Swap分卷类似，需要将逻辑卷名称设置为“root”，大小设置为“14.50 GiB”。
“Formatting Options”和“Mounting Options”按下图设置，单击“Finish”完成root分卷的创建。



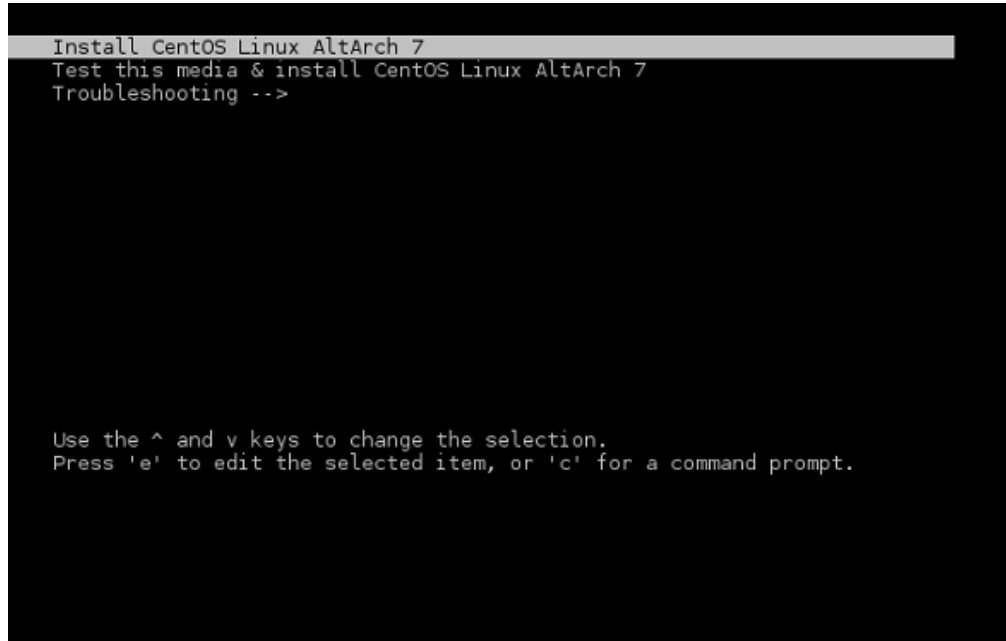
确认分区和分卷信息：



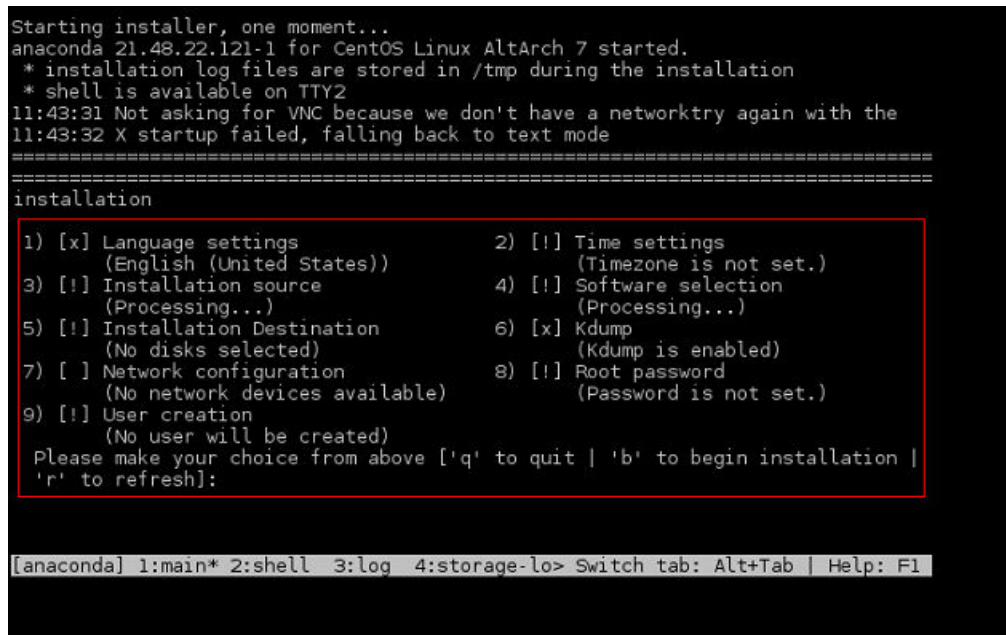
至此，分区和分卷划分完成。单击“Next”，继续按照提示进行操作系统的安装。

3.4.6 CentOS 7.4 for ARM

1. 选择“Install CentOS Linux AltArch 7”开始安装。



2. 根据页面提示的选项进行安装。



3. 设置时区。

```
(English (United States))
3) [!] Installation source (Processing...)
5) [!] Installation Destination (No disks selected)
7) [ ] Network configuration (Not connected)
9) [!] User creation (No user will be created)
Please make your choice from above ['q' to quit | 'b' to begin installation |
'r' to refresh]: 2
=====
Time settings
Timezone: not set
NTP servers: not configured
1) Set timezone
2) Configure NTP servers
Please make your choice from above ['q' to quit | 'c' to continue |
'r' to refresh]: 1
[anaconda] 1:main* 2:shell 3:log 4:storage-lo> Switch tab: Alt+Tab | Help: F1

=====
Time settings
Timezone: not set
NTP servers: not configured
1) Set timezone
2) Configure NTP servers
Please make your choice from above ['q' to quit | 'c' to continue |
'r' to refresh]: 1
=====
Timezone settings
Available regions
1) Europe
2) Asia
3) America
4) Africa
5) Antarctica
6) Pacific
7) Australia
8) Atlantic
9) Indian
10) Arctic
11) US
12) Etc
Please select the timezone.
Use numbers or type names directly [b to region list, q to quit]: 2
[anaconda] 1:main* 2:shell 3:log 4:storage-lo> Switch tab: Alt+Tab | Help: F1
```

4. 根据需要选择安装软件。

```

24) Dushanbe          52) Novosibirsk       79) Yakutsk
25) Famagusta         53) Omsk               80) Yangon
26) Gaza              54) Oral               81) Yekaterinburg
27) Hebron            55) Phnom_Penh         82) Yerevan
28) Ho_Chi_Minh
Please select the timezone.
Use numbers or type names directly [b to region list, q to quit]: 64
=====
Installation

1) [x] Language settings          2) [x] Time settings
   (English (United States))      (Asia/Shanghai timezone)
3) [x] Installation source        4) [x] Software selection
   (Local media)                  (Minimal Install)
5) [!] Installation Destination   6) [x] Kdump
   (No disks selected)            (Kdump is enabled)
7) [ ] Network configuration      8) [!] Root password
   (Not connected)               (Password is not set.)
9) [!] User creation
   (No user will be created)
Please make your choice from above ['q' to quit | 'b' to begin installation |
'r' to refresh]: 4
[anaconda] 1:main* 2:shell 3:log 4:storage-lo> Switch tab: Alt+Tab | Help: F1

```

```

=====
Base environment
Software selection
Base environment

1) [x] Minimal Install          6) [ ] Server with GUI
2) [ ] Compute Node           7) [ ] GNOME Desktop
3) [ ] Infrastructure Server    8) [ ] KDE Plasma Workspaces
4) [ ] File and Print Server    9) [ ] Development and Creative
5) [ ] Basic Web Server         Workstation
Please make your choice from above ['q' to quit | 'c' to continue |
'r' to refresh]: 5
[anaconda] 1:main* 2:shell 3:log 4:storage-lo> Switch tab: Alt+Tab | Help: F1

```

5. 设置系统密码。

```

Current host name: localhost

1) Set host name
2) Configure device eth0
Please make your choice from above ['q' to quit | 'c' to continue |
'r' to refresh]: c
=====
Installation

1) [x] Language settings          2) [x] Time settings
   (English (United States))      (Asia/Shanghai timezone)
3) [x] Installation source        4) [x] Software selection
   (Local media)                  (Basic Web Server)
5) [x] Installation Destination   6) [x] Kdump
   (Automatic partitioning        (Kdump is enabled)
   selected)                      8) [!] Root password
7) [ ] Network configuration      (Password is not set.)
   (Not connected)
9) [!] User creation
   (No user will be created)
Please make your choice from above ['q' to quit | 'b' to begin installation |
'r' to refresh]: 8
[anaconda] 1:main* 2:shell 3:log 4:storage-lo> Switch tab: Alt+Tab | Help: F1

```

6. 完成配置项设置后，输入“b”，按“Enter”，开始安装。

```
1) Set host name
2) Configure device eth0
   Please make your choice from above ['q' to quit | 'c' to continue |
   'r' to refresh]: c
=====
Installation
=====
1) [x] Language settings                2) [x] Time settings
   (English (United States))           (Asia/Shanghai timezone)
3) [x] Installation source              4) [x] Software selection
   (Local media)                       (Basic Web Server)
5) [x] Installation Destination         6) [x] Kdump                enabled
   (Automatic partitioning             (Kdump is enabled
   selected)
7) [ ] Network configuration            8) [!] Root password
   (Not connected)                   (Password is not set.)
9) [!] User creation
   (No user will be created)
   Please make your choice from above ['q' to quit | 'b' to begin installation |
   'r' to refresh]: 8
[anaconda] 1:main* 2:shell 3:log 4:storage-lo> Switch tab: Alt+Tab | Help: F1
```

3.5 启动引导文件修改（仅 UEFI 启动场景涉及）

UEFI启动场景下，需要修改启动引导文件，否则，使用该虚拟机镜像发放的裸金属服务器强制重启后会概率性启动失败。

安装完操作系统后，先不要重启，按Ctrl + Alt + F2进入命令行模式，按如下指导修改完成后，再重启操作系统。

说明

- 启动失败原因：制作镜像过程通过虚拟化制作，在UVP底层转化过程中，可能概率性会遇到转换格式差异，导致启动文件转换失败，从而在使用该虚拟机镜像发放的裸金属服务器强制重启后会概率性启动失败，所以需要手动固化grub，保证启动不会失败。
- 是否与操作系统有关：与操作系统无关，UEFI启动的系统建议都做替换操作，固化grub。
- 影响面：如果不作处理，会有概率性启动失败，会影响业务面操作，导致裸机上部署的业务无法继续运行。

ARM 镜像

先执行**find / -name "boot/efi/EFI"**找到启动文件路径，再使用ARM镜像启动引导文件“boot/efi/EFI/\$os_version/grubaa64.efi”替换“boot/efi/EFI/BOOT/BOOTAA64.EFI”。

以CentOS 7.4系统为例，\$os_version为centos，按下图操作使用“boot/efi/EFI/centos/grubaa64.efi”替换“boot/efi/EFI/BOOT/BOOTAA64.EFI”。

```
[root@localhost EFI]# cp BOOT/BOOTAA64.EFI BOOT/BOOTAA64.EFI.backup
[root@localhost EFI]# cp centos/grubaa64.efi BOOT/BOOTAA64.EFI
cp: overwrite 'BOOT/BOOTAA64.EFI'? y
[root@localhost EFI]# pwd
/boot/efi/EFI
```

不同操作系统的ARM镜像启动引导文件grubaa64.efi所在位置不同，请参见[表3-3](#)查阅。

表 3-3 ARM 镜像启动引导文件 grubaa64.efi 所在位置

OS	文件位置
CentOS	/boot/efi/EFI/centos/grubaa64.efi
EulerOS	/boot/efi/EFI/euleros/grubaa64.efi
SUSE	/boot/efi/EFI/sles/grubaa64.efi
Ubuntu	/boot/efi/EFI/ubuntu/grubaa64.efi
Red Hat	/boot/efi/EFI/redhat/grubaa64.efi

x86 镜像

先执行find / -name "boot/efi/EFI"找到启动文件路径，比如EulerOS 2.5系统的文件路径为“/boot/efi/EFI”，使用x86镜像启动引导文件“/boot/efi/EFI/\$os_version/grubx64.efi”替换“/boot/efi/EFI/BOOT/BOOTX64.EFI”。

以EulerOS 2.5系统为例，\$os_version为euleros，按下图操作使用“/boot/efi/EFI/euleros/grubx64.efi”替换“/boot/efi/EFI/BOOT/BOOTX64.EFI”。

```
[anaconda root@euleros2u5-uefi euleros]# ls -l ../BOOT/
total 1264
-rwx-----. 1 root root 1293149 Feb 12 00:00 BOOTX64.EFI
[anaconda root@euleros2u5-uefi euleros]# cp ../BOOT/BOOTX64.EFI ./
[anaconda root@euleros2u5-uefi euleros]# ls -l
total 6080
-rwx-----. 1 root root      136 Feb 12 00:00 BOOT.CSV
-rwx-----. 1 root root      136 Feb 12 00:00 BOOTX64.CSV
-rwx-----. 1 root root 1293149 Aug  9 13:03 BOOTX64.EFI
drwx-----. 2 root root    4096 Aug  9 12:45 fonts
-rwx-----. 1 root root    5657 Aug  9 12:49 grub.cfg
-rwx-----. 1 root root    1024 Aug  9 12:49 grubenv
-rwx-----. 1 root root 1053832 Feb 12 00:00 grubx64.efi
-rwx-----. 1 root root 1261168 Feb 12 00:00 mmx64.efi
-rwx-----. 1 root root 1293149 Feb 12 00:00 shim.efi
-rwx-----. 1 root root 1293149 Feb 12 00:00 shimx64.efi
[anaconda root@euleros2u5-uefi euleros]# rm -rf BOOTX64.EFI
[anaconda root@euleros2u5-uefi euleros]# cd ../BOOT/
[anaconda root@euleros2u5-uefi BOOT]# cp BOOTX64.EFI BOOTX64.EFI.bak
[anaconda root@euleros2u5-uefi BOOT]# cp ../euleros/grubx64.efi ./BOOTX64.EFI
[anaconda root@euleros2u5-uefi BOOT]# ls -l
total 2296
-rwx-----. 1 root root 1053832 Aug  9 13:05 BOOTX64.EFI
-rwx-----. 1 root root 1293149 Aug  9 13:05 BOOTX64.EFI.bak
[anaconda root@euleros2u5-uefi BOOT]# pwd
/mnt/sysimage/boot/efi/EFI/BOOT
[anaconda root@euleros2u5-uefi BOOT]#
```

不同操作系统的x86镜像启动引导文件grubx64.efi所在位置不同，请参见表3-4。

表 3-4 x86 镜像启动引导文件 grubx64.efi 所在位置

OS	文件位置
CentOS	/boot/efi/EFI/centos/grubx64.efi
EulerOS	/boot/efi/EFI/euleros/grubx64.efi

OS	文件位置
SUSE	/boot/efi/EFI/sles/grubx64.efi
Ubuntu	/boot/efi/EFI/ubuntu/grubx64.efi
Red Hat	/boot/efi/EFI/redhat/grubx64.efi

3.6 虚拟机环境配置

3.6.1 概述

为虚拟机配置网络环境使其能够连接互联网，上传安装包到虚拟机。

前提条件

- 宿主机可以连接互联网。
- 已经通过远程桌面工具VNC Viewer等软件连接到宿主机。
- 所需安装包已经上传至宿主机。

3.6.2 安装基础组件包（可选）

操作场景

仅Debian系列操作系统需要执行此步骤，其他操作系统请跳过此步骤。安装的基础组件包包括vim软件、dkms包、linux--headers-xxx-common和linux-headers-xxx-amd64包。

操作步骤

1. 安装vim软件。
 - a. 配置apt源。

执行**vi /etc/apt/sources.list**命令，添加apt源。不同版本的Debian操作系统所添加的内容不一样，以下文本以Debian 8.6为例。

```
deb http://mirrors.ustc.edu.cn/debian jessie main contrib non-free
deb-src http://mirrors.ustc.edu.cn/debian jessie main contrib non-free
deb http://mirrors.ustc.edu.cn/debian jessie-proposed-updates main contrib non-free
deb-src http://mirrors.ustc.edu.cn/debian jessie-proposed-updates main contrib non-free
deb http://mirrors.ustc.edu.cn/debian jessie-updates main contrib non-free
deb-src http://mirrors.ustc.edu.cn/debian jessie-updates main contrib non-free
```

输入:**wq!**保存文件，然后执行**apt update**命令。
 - b. 执行**apt-get install vim**安装vim软件。
2. 执行**apt-get install dkms**安装dkms包。
3. 执行**apt-get install linux--headers-xxx-common**安装linux--headers-xxx-common包。

xxx代表具体内核版本号，如Debian 8.6的内核版本为3.16.0-4，请执行**apt-get install linux--headers-3.16.0-4-common**。

4. 执行**apt-get install linux-headers-xxx-amd64**安装linux-headers-xxx-amd64包。

xxx代表具体内核版本号，如Debian 8.6的内核版本为3.16.0-4，请执行**apt-get install linux-headers-3.16.0-4-amd64**。

5. 删除“/etc/network/interfaces”文件的配置项。

执行命令**vi /etc/network/interfaces**，如果存在eth0的配置，删除最后两行配置项。

```
...
# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
# 以下为需要删除的配置项：
allow-hotplug eth0
iface eth0 inet dhcp
```

输入:**wq!**保存文件。

3.6.3 虚拟机网络配置

操作场景

为虚拟机配置一个可用的IP地址，使虚拟机和宿主机网络互通。

suse15系列若是不存在ifconfig及route命令，使用zypper install net-tools-deprecated命令，安装对应工具包，即可解决

操作步骤

1. 在虚拟机上，执行以下命令，查看网卡名称。

ifconfig -a

2. 执行以下命令，查看网卡是否获取到IP地址。

ifconfig

显示如下回显信息（xxx表示自动获取的IP地址，XX表示MAC地址）：

```
eth0  Link encap:Ethernet  HWaddr XX:XX:XX:XX:XX:XX
inet addr:xxx.xxx.xxx.xxx  Bcast:xxx.xxx.x.xxx  Mask:xxx.xxx.xxx.xxx
```

3. 如果没有获取到（否则跳过此步骤），执行以下命令，使网卡动态获取IP地址。

ifup 网卡名称

示例：

ifup eth0

显示如下回显信息：

```
Determining IP information for eth0... done
```

若提示无法ifup，则可以使用如下命令：

ip link set 网卡名称 up

dhclient 网卡名称

也可以执行下列命令：

ifconfig eth0 up

一般无回显信息。

3.6.4 设置 systemd 超时时间参数默认值

操作场景

设置时间参数，防止裸金属服务器下发超时。

操作步骤

对于Red Hat 7系列/EulerOS/CentOS 7系列/CentOS 8系列/Oracle Linux 7系列/Ubuntu 16.04/Ubuntu 18.04/SUSE 12 SP2/SUSE 12 SP3/SUSE 15/Debian操作系统，执行以下命令：

vi /etc/systemd/system.conf

去掉“DefaultTimeoutStartSec”和“DefaultTimeoutStopSec”前的#号注释，将时间修改为300s，修改后的配置文件如下所示：

```
#TimeSlackNSec=  
#DefaultTimerAccuracySec=1min  
#DefaultStandardOutput=journal  
#DefaultStandardError=inherit  
DefaultTimeoutStartSec=300s  
DefaultTimeoutStopSec=300s  
#DefaultRestartSec=100ms  
#DefaultStartLimitInterval=10s  
#DefaultStartLimitBurst=5  
#DefaultEnvironment=  
#DefaultCPUsAccounting=no  
#DefaultBlockIOAccounting=no
```

3.6.5 关闭防火墙

操作场景

关闭虚拟机防火墙。防火墙会阻止远程SSH登录等连接，需要关闭。

操作步骤

- 对于Red Hat 7.0/Red Hat 7.2/Red Hat 7.3/Red Hat 7.4/Oracle Linux 7系列/EulerOS/CentOS 7系列/CentOS 8/SUSE15系列操作系统，依次执行以下命令：

systemctl disable firewalld.service

systemctl stop firewalld.service

查看服务状态：**systemctl status firewalld.service**

- 对于Red Hat 6.7/Red Hat 6.8/Red Hat 6.9/CentOS 6.8/CentOS 6.9/Oracle Linux 6.8/Oracle Linux 6.9操作系统，依次执行以下命令：

chkconfig iptables off

service iptables stop

查看服务状态：**service iptables status**

```
[root@localhost ~]# service iptables status  
iptables: Firewall is not running.
```

- 对于SUSE 12操作系统，依次执行以下命令：

systemctl disable SuSEfirewall2.service

systemctl stop SuSEfirewall2.service

查看服务状态：**service SuSEfirewall2 status**

- 对于SUSE 11操作系统，先执行：
rcSuSEfirewall2 stop
再执行下面命令：
 - a. **yast**
 - b. 选择“Security and Users > Firewall > Disble Firewall Automatic Starting”。
 - c. 查看是否为“Disble Firewall Automatic Starting”。
 - 是，单击“Cancel > Quit”。
 - 否，选择它，单击“Next > Finish > Quit”。
- 对于Ubuntu 18.04/Ubuntu 16.04/Ubuntu 14.04/Debian操作系统，执行下面命令关闭防火墙：
ufw disable
如果没有ufw命令，可以在官网下载防火墙管理软件ufw并安装（下载地址参考：<https://packages.ubuntu.com/>）。
deb安装包如下（版本号仅供参考）：
ufw_0.35-0ubuntu2_all.deb
或者参考“[SUSE/Red Hat/CentOS/Oracle Linux/Ubuntu/Debian系列](#)”章节配置完apt源后，使用命令**apt-get install ufw**进行安装，之后再执行**ufw disable**关闭防火墙。

3.6.6 升级 wicked 组件（可选）

操作场景

对于SUSE 12 SP1，需要升级wicked组件。其他操作系统请跳过此步骤。

操作步骤

1. 执行**rpm -ivh --nodeps --force *rpm**命令强制安装wicked软件包。
2. 查看当前已安装的所有wicked软件包。

```
linux-locc:/home/fsp/Desktop # rpm -qa | grep wick
wicked-service-0.6.28-1.1.x86_64
libwicked-0-6-0.6.28-1.1.x86_64
libwicked-0-6-0.6.40-28.6.1.x86_64
wicked-0.6.28-1.1.x86_64
wicked-service-0.6.40-28.6.1.x86_64
wicked-0.6.40-28.6.1.x86_64
```
3. 卸载旧版本的wicked软件包。

```
linux-locc:/home/fsp/Desktop # rpm -e wicked-service-0.6.28-1.1.x86_64
linux-locc:/home/fsp/Desktop # rpm -e libwicked-0-6-0.6.28-1.1.x86_64
linux-locc:/home/fsp/Desktop # rpm -e wicked-0.6.28-1.1.x86_64
```

3.6.7 关闭虚拟机的网络管理工具（可选）

操作场景

对于集中式裸金属服务器，系统的网络管理可能与network-config配置冲突，需要关闭。分布式裸金属服务器请跳过此步骤。

说明

SUSE 11 SP4、Ubuntu 16.04和Ubuntu 14.04不需要执行此步骤；带SDI 3.0卡（存储和网络归一）或SDI 2.2网络卡的机型由于使用NetworkManager管理网络，故不需要执行此步骤。

注：

由NetworkManager管理网络的裸机镜像配置（仅ubuntu操作系统涉及，其他操作系统跳过）：

以ubuntu1804.2为例：

1. 执行systemctl status NetworkManager，查看服务状态设置为开机自启动

```
root@bms-ubuntu:~# systemctl status NetworkManager
● NetworkManager.service - Network Manager
   Loaded: loaded (/lib/systemd/system/NetworkManager.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2022-10-09 10:26:45 CST; 17min ago
     Docs: man:NetworkManager(8)
   Main PID: 3685 (NetworkManager)
      Tasks: 4 (limit: 13510)
    CGroup: /system.slice/NetworkManager.service
            └─3685 /usr/sbin/NetworkManager --no-daemon
              └─4420 /sbin/dhclient -d -q -sf /usr/lib/NetworkManager/nm-dhcp-helper -pf /run/dhclient-eth1.pid -lf /var/lib/NetworkManager/dhclient-433e484b-3493

Oct 09 10:26:45 localhost NetworkManager[3685]: <info> [1665282405.3567] device (eth1): state change: secondaries -> activated (reason 'none', sys-iface-state
Oct 09 10:26:45 localhost NetworkManager[3685]: <info> [1665282405.3567] manager: NetworkManager state is now CONNECTED_LOCAL
Oct 09 10:26:45 localhost NetworkManager[3685]: <info> [1665282405.3578] manager: NetworkManager state is now CONNECTED_SITE
Oct 09 10:26:45 localhost NetworkManager[3685]: <info> [1665282405.3579] policy: set 'netplan-eth1' (eth1) as default for IPv4 routing and DNS
Oct 09 10:26:45 localhost NetworkManager[3685]: <info> [1665282405.3573] device (eth1): Activation: successful, device activated.
Oct 09 10:26:45 localhost NetworkManager[3685]: <info> [1665282405.3579] manager: NetworkManager state is now CONNECTED_GLOBAL
Oct 09 10:26:45 localhost dhclient[4420]: bound to 192.168.24.87 -- renewal in 1368147 seconds.
Oct 09 10:26:51 localhost NetworkManager[3685]: <info> [1665282411.2169] manager: startup complete
Oct 09 10:28:59 bms-test NetworkManager[3685]: <info> [1665282539.8826] hostname: hostname changed from "localhost" to "bms-test"
Oct 09 10:29:24 bms-ubuntu NetworkManager[3685]: <info> [1665282564.3807] hostname: hostname changed from "bms-test" to "bms-ubuntu"
lines 1-29/29 (END)
```

若该服务不存在，则执行apt-get install network-manager命令安装

2. 需要修改/etc/netplan/01*yaml文件，添加网络管理服务及网卡信息，如下

```
# This file describes the network
# For more information, see netplan
network:
  version: 2
  renderer: NetworkManager
  ethernets:
    ens3:
      dhcp4: yes
```

然后执行 :wq! 保存退出，再执行netplan try刷新网络配置

操作步骤

- 对于Red Hat 7系列/Oracle Linux 7系列/Debian/EulerOS/CentOS 7/CentOS 8系列操作系统，依次执行以下命令：

systemctl disable NetworkManager.service

systemctl stop NetworkManager.service

查看服务状态：**service NetworkManager.service status**

```
[root@localhost ~]# service NetworkManager.service status
redirecting to /bin/systemctl status NetworkManager.service
NetworkManager.service - Network Manager
   Loaded: loaded (/usr/lib/systemd/system/NetworkManager.service; disabled; vendor preset: enabled)
   Active: inactive (dead) since Mon 2017-11-13 19:06:18 CST; 1 min 17s ago
```

对于EulerOS操作系统，如果有以下两个服务，还需要关闭这两个服务：

systemctl disable euleros-security

systemctl disable NetworkManager-wait-online

- 对于Red Hat 6.7/Red Hat 6.8/Red Hat 6.9/CentOS 6.8/CentOS 6.9/Oracle Linux 6.8/Oracle Linux 6.9操作系统，依次执行以下命令：

说明

对于Red Hat 6.7/Red Hat 6.8/Red Hat 6.9/CentOS 6.8/CentOS 6.9/Oracle Linux 6.8/Oracle Linux 6.9操作系统，如果创建虚拟机时选择安装桌面，则需要执行如下关闭网络服务操作，若选择的是基础安装则不需要执行。

service NetworkManager stop

chkconfig NetworkManager off

- 对于SUSE 12操作系统，依次执行以下命令：

systemctl disable wicked

systemctl stop wicked

- 对于Ubuntu18.04操作系统依次执行以下命令：

apt-get install ifupdown --> 安装ifupdown，用于ifup网卡

apt-get install ifenslave --> 安装ifenslave，用于bond管理

apt-get --assume-yes purge nplan netplan.io --> 卸载netplan

3.6.8 删除虚拟机网络管理工具插件（可选）

说明

分布式裸机镜像，使用os自带的NetworkManager网络管理工具，需要删除NetworkManager-config-server插件，否则会影响网卡自动获取ip。集中式裸机镜像不适用NetworkManager工具管理网络，可跳过。

执行命令 **rpm -qa | grep NetworkManager-config-server** 查询。

若存在则执行**rpm -e NetworkManager-config-server** 进行删除。

3.6.9 删除虚拟机的本地用户（可选）

操作场景

安装虚拟机的过程需要创建本地用户，如果不需要可以进行删除。

操作步骤

执行命令：**userdel -rf xxx**

其中，xxx为本地用户的名称。“/home”目录下无该用户的文件夹表示删除成功。

说明

如果删除失败，请重启虚拟机并使用root用户登录，重新执行删除命令。

3.6.10 修改 DHCP 配置项（可选）

操作场景

对于SUSE系列操作系统，需要修改DHCP配置项，其他操作系统请跳过此步骤。

操作步骤

1. 在操作系统中打开命令行终端，执行**su - root**命令切换到root用户。
2. 执行**vi /etc/sysconfig/network/dhcp**命令以vi编辑器打开待修改的配置文件。
3. 输入“?DHCLIENT_PRIMARY_DEVICE”，按“Enter”定位到待修改的配置项。
按“i”进入编辑模式，将配置项的取值修改为“yes”，配置DHCP启动时获取默认网关。
按“Esc”退出编辑模式。

说明

如果配置文件中没有“DHCLIENT_PRIMARY_DEVICE”配置项，则忽略此配置。

4. 输入“?DHCLIENT_SET_HOSTNAME”，按“Enter”定位到待修改的配置项。
按“i”进入编辑模式，将配置项的取值修改为“no”，防止DHCP过程修改主机名。
按“Esc”退出编辑模式。
5. 输入“?DHCLIENT_USE_LAST_LEASE”，按“Enter”定位到待修改的配置项。
按“i”进入编辑模式，将配置项的取值修改为“no”，配置DHCP持续获取IP地址能力。
按“Esc”退出编辑模式。
6. 输入“?DHCLIENT6_MODE”，按“Enter”定位到待修改的配置项。
按“i”进入编辑模式，将配置项的取值修改为“managed”。
- 按“Esc”退出编辑模式。
7. 输入:wq命令，按“Enter”保存配置并退出。

3.6.11 设置 grub 配置文件超时参数（可选）

操作场景

对于Ubuntu 14.04/Debian，需要为grub配置文件设置超时参数，防止服务器异常掉电，造成系统无法正常进入。其他操作系统请跳过此步骤。

操作步骤

1. 使用vi编辑器打开“/etc/default/grub”，在GRUB_CMDLINE_LINUX字段内容后面增加GRUB_RECORDFAIL_TIMEOUT=10。

```
GRUB_DEFAULT=0
#GRUB_HIDDEN_TIMEOUT=0
GRUB_HIDDEN_TIMEOUT_QUIET=true
GRUB_TIMEOUT=2
GRUB_DISTRIBUTOR='lsb_release -i -s 2> /dev/null || echo Debian'
GRUB_CMDLINE_LINUX_DEFAULT=""
GRUB_CMDLINE_LINUX="console=tty0 console=ttyS0"
GRUB_RECORDFAIL_TIMEOUT=10
```
2. 执行以下命令刷新配置。
grub-mkconfig -o /boot/grub/grub.cfg

3.6.12 设置句柄链接数为 65535

- 一般linux 服务器默认的句柄数都是 1024，查看方法如下：

```
[root@platservice6~]# ulimit -n
```

1024

- 裸金属服务器镜像标准要求镜像的句柄链接数为65535，修改方法如下：

使用vim命令编辑 /etc/systemd/system.conf文件设置如下

DefaultLimitNOFILE=65535

DefaultLimitNPROC=65535

说明

按照以上步骤修改后，需要重启后登录重新查询，否则句柄数不会生效。

3.6.13 上传需要的软件包到虚拟机

操作场景

上传需要的软件包到虚拟机时，根据宿主机和虚拟机的网络连通情况，有三种方法，详见操作步骤。

操作步骤

- 方法一：宿主机和虚拟机网络连通的情况下，可以使用scp方式传输文件到虚拟机（推荐使用，因为此命令简单方便，可通过**scp -help**查询用法）。

例如，在虚拟机中执行以下命令（命令中的文件名以实际为准，此处仅为示例）：

```
scp fsp@xxx.xxx.xxx.xxx:/home/fsp/network-config-1.0-1.x86_64.rpm /home
```

说明：scp 用户名@宿主机ip:/自定义目录/所传文件名 /虚拟机目录

- 方法二：宿主机和虚拟机网络连通且可以通过Xshell登录虚拟机的情况下，可以使用Xftp进行文件传输。
- 方法三：如果宿主机和虚拟机网络不通，使用虚拟光驱挂载方式。

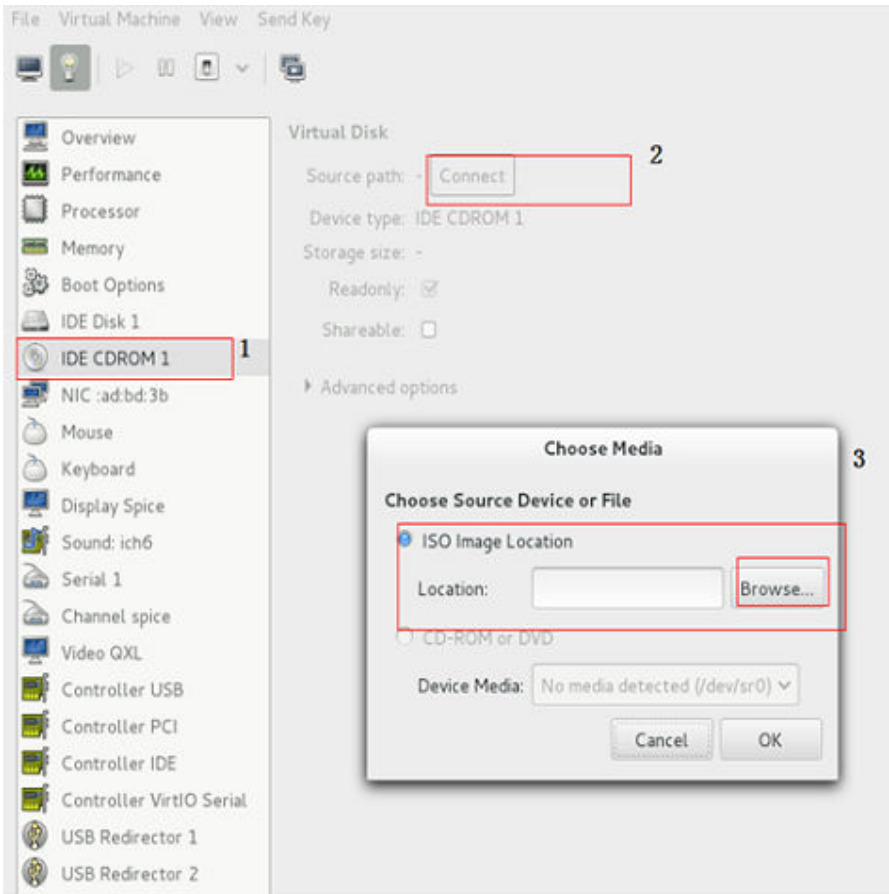
第一步：在宿主机中创建ISO文件

在Linux终端中执行：

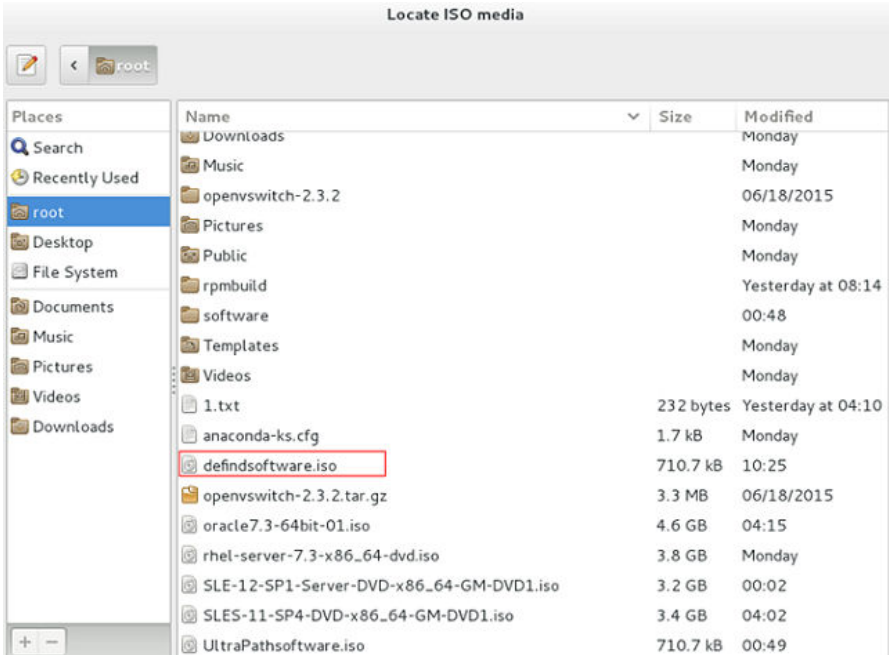
- 创建目录，执行**mkdir /root/software**命令。
- 将network-config和SDI卡驱动软件包放入software中。
获取软件包请参见[软件准备](#)章节。
- 执行**cd /root**，然后执行**mkisofs -L -R -J -T -V system-sp2 -o defindsoftware.iso /root/software**，建立ISO映像文件。
- 执行**ll**，结果显示的“defindsoftware.iso”即为制作的ISO文件。

第二步：使用virt-manager挂载ISO文件

- 在virt-manager界面选择“View > Details”。
- 选择“IDE CDROM 1”页签，单击右侧的“Connect”，按照下图的1、2、3依次完成操作。



- c. 单击“Browse Local”，选择/root目录。
- d. 找到“defindsoftware.iso”文件并双击，在弹出的页面中单击“OK”。



- e. 再次选择“View > Console”，选择对应正在操作的虚拟机。
- f. 进入虚拟机后打开终端，执行lsblk命令，查看ISO文件是否有挂载，例如挂载点在“/run/media/suse/system-sp2”。

- g. 执行`cd /run/media/suse/system-sp2`，将其中的文件复制到某个目录下，如“/home”。如果在挂载点处没有文件目录，就需要手动挂载，使用命令`mount /dev/sr0 /home`（挂载到/home目录下）。

3.7 安装 Cloud-Init

3.7.1 安装 Cloud-Init 说明

操作背景

Cloud-Init是为云环境中虚拟机或裸金属服务器的初始化而开发的工具，作用是在使用镜像创建虚拟机或裸金属服务器时，可自定义服务器的网络配置、hostname、hosts文件，以及用户名密码等。使用镜像创建的虚拟机如需由系统生成随机密码，也需要使用此工具实现。

使用镜像创建虚拟机或裸金属服务器时，没有上述需求，可以不安装Cloud-Init。由于Cloud-Init安装文件需要适配不同厂商的Linux版本，当前仅支持从网络安装，因此制作Linux镜像时，需要可接入Internet。

前提条件

- 已登录虚拟机。
- 宿主机可以连接互联网。
- 已使用VNC Viewer登录宿主机，已在启动的virt-manager中安装好相应操作系统。

操作说明

1. 以下安装Cloud-Init只是安装示例，建议从操作系统提供商的官网下载此软件。Cloud-Init版本会随官网实时更新，在安装时，使用官网的最新版本即可。
2. 在修改配置文件“/etc/cloud/cloud.cfg”时，请注意配置文件的格式（如对齐、空格）要与示例保持一致（此文件遵循yaml语法结构）。

3.7.2 SUSE/Red Hat/CentOS/Oracle Linux/Ubuntu/Debian 系列

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）](#)、[采用官方提供的Cloud-Init源码包通过pip方式安装Cloud-Init工具](#)和[采用源码编译安装方法](#)。

采用官方提供的包源安装 Cloud-Init 工具（优先推荐）

在不同操作系统的虚拟机上安装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、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：
<http://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. 执行以下命令，安装SUSE 12的网络安装源。
zypper ar http://ftp5.gwdg.de/pub/opensuse/repositories/Cloud:/Tools/SLE_12_SP3/Cloud:Tools.repo
- b. 执行以下命令，更新网络安装源。
zypper refresh
- c. 执行以下命令，安装Cloud-Init。
zypper install cloud-init
- d. 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

须知

对于SUSE和OpenSUSE操作系统，请执行以下步骤禁止动态修改名称。

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3-5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 3-5 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/Packages/e/

操作系统类型	版本	网络安装地址
	6系列64位	https://archives.fedoraproject.org/pub/archive/epel/6/x86_64/Packages/e/
	7系列64位	https://archives.fedoraproject.org/pub/archive/epel/7/x86_64/Packages/e/

此处以CentOS 6.5 64位为例，执行以下命令安装Cloud-Init。

```
yum install https://archives.fedoraproject.org/pub/archive/epel/6/x86_64/Packages/e/epel-release-xx-xx.noarch.rpm
```

```
yum install cloud-init
```

说明

xx-xx表示当前操作系统对应的epel版本号。

- **Debian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官网信息配置软件包安装源。

执行以下命令，安装Cloud-Init。

```
apt-get update
```

```
apt-get install cloud-init
```

Debian操作系统安装完Cloud-Init后，还需要安装vlan和ifenslave服务，请执行以下命令进行安装：

```
apt-get install vlan
```

```
apt-get install ifenslave
```

- **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Ubuntu官网信息配置软件包安装源。

执行以下命令，安装Cloud-Init。

```
apt-get update
```

```
apt-get install cloud-init
```

Ubuntu操作系统安装完Cloud-Init后，还需要安装ssh服务、dkms工具、vlan和ifenslave服务，以及ifupdown服务，请执行以下操作进行安装：

- a. 安装ssh服务。

对于x86镜像，执行命令：

```
apt-get install openssh-client
```

```
apt-get install openssh-server
```

对于ARM64镜像，执行命令：

```
apt install openssh-client
```

```
apt install openssh-server
```

b. 安装dkms工具。

为了防止SDI驱动异常，Ubuntu需要安装dkms，可以使用apt-get命令在线安装。

执行命令进行安装：**apt-get install dkms**

待安装完成后，执行命令：**vi /usr/sbin/dkms**

然后跳转到283行（同时按下“shift”和“:”进入命令行模式，输入283并按“Enter”），修改为：

```
invoke_command "$mkinitrd -f $initrd_dir/$initrd $1" "$mkinitrd" background
```

c. 安装vlan和ifenslave服务。

apt-get install vlan

apt-get install ifenslave

d. 安装ifupdown服务。

apt-get install ifupdown

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），上传到虚拟机指定目录“/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>

2. 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/>

3. 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

4. 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
```

5. 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
```

须知

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**）。

采用源码编译安装方法

由于Cloud-Init配置的相关内容已在源码包编译完成，执行以下操作步骤安装Cloud-Init成功即可，无需执行Cloud-Init配置操作。官方源码编译安装方法的Cloud-Init工具Github开源地址：<https://github.com/canonical/cloud-init/>

1. 执行以下命令，下载Cloud-Init压缩包，并将其复制至新建的“/tmp/CLOUD-INIT”文件夹。

说明

0.7.6版本安装包下载地址：<https://github.com/canonical/cloud-init/archive/refs/tags/0.7.6.zip>

0.7.9版本安装包下载地址：<https://github.com/canonical/cloud-init/archive/refs/tags/0.7.9.zip>

```
wget https://github.com/canonical/cloud-init/archive/refs/tags/0.7.9.zip
```

```
mkdir /tmp/CLOUD-INIT
```

```
cp cloud-init-0.7.9.zip /tmp/CLOUD-INIT
```

```
cd /tmp/CLOUD-INIT
```

2. 执行如下命令，解压Cloud-Init压缩包。

```
unzip cloud-init-0.7.9.zip
```

3. 执行如下命令进入cloud-init-0.7.9文件夹。

```
cd cloud-init-0.7.9
```

4. 按照操作系统类型，执行相应的命令安装Cloud-Init安装包。

- CentOS 6.x/SUSE 11.x:

- ```
python setup.py build
python setup.py install --init-system sysvinit
```
- CentOS 7.x/SUSE 12.x/EulerOS 2.8 ARM:

```
python setup.py build
python setup.py install --init-system systemd
```

#### 须知

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
```

5. 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
```
6. 执行如下命令检查Cloud-Init是否安装成功。

```
cloud-init -v
cloud-init init --local
```

回显如下类似信息所示表示安装Cloud-Init成功。

```
cloud-init 0.7.9
```

### 3.7.3 EulerOS/OpenEuler

1. 以EulerOS 2.2为例，首先进行EulerOS 2.2的yum源配置，编辑“/etc/yum.repos.d/EulerOS-base.repo”文件。以下为配置示例：

```
[EulerOS-base]
name=EulerOS-base
baseurl=https://repo.huaweicloud.com/euler/2.2/os/x86_64/
enabled=1
gpgcheck=1
gpgkey=https://repo.huaweicloud.com/euler/2.2/os/RPM-GPG-KEY-EulerOS
```

编辑后保存该文件。
2. yum源配置完成后，执行命令进行刷新：

```
yum repolist
```

然后进行cloud-init-0.7.6的安装：

**yum install cloud-init**

安装的过程会自动将Cloud-Init所需依赖一起安装。

```
Installed:
cloud-init.x86_64 0:0.7.6-2

Dependency Installed:
PyYAML.x86_64 0:3.10-11 audit-libs-python.x86_64 0:2.4.1-5
checkpolicy.x86_64 0:2.1.12-6 libsemanage-python.x86_64 0:2.1.10-18
libyaml.x86_64 0:0.1.4-11 policycoreutils-python.x86_64 0:2.2.5-15.h1
python-IPy.noarch 0:0.75-6 python-backports.x86_64 0:1.0-8
python-backports-ssl_match_hostname.noarch 0:3.4.0.2-4 python-jsonpatch.noarch 0:1.2-2
python-jsonpointer.noarch 0:1.9-2 python-prettytable.noarch 0:0.7.2-1
python-requests.noarch 0:2.6.0-1 python-six.noarch 0:1.9.0-2
python-urllib3.noarch 0:1.10.2-2 setools-libs.x86_64 0:3.3.7-46

Complete!
```

3. 如果要注入root密码，需要升级selinux-policy，通过下面命令，从h1升级至h2。  
**yum install selinux-policy**
4. 执行**cloud-init -v**命令，回显信息中包含cloud-init版本号，则表示安装完成。

## 3.8 配置 Cloud-Init

在安装完Cloud-Init后，需要配置cloud.cfg文件，用于定制Cloud-Init在裸金属服务器初始化时的功能配置。

使用vi编辑器修改“/etc/cloud/cloud.cfg”配置文件，以下章节内容仅为示例，应该以实际需求为准。

### 3.8.1 cloud-init-0.7.5 配置

1. 添加如下键值对，保持前后各空出一行：  
no\_ssh\_fingerprints: true
2. 设置ssh\_pwauth。该选项表示是否支持ssh密码登录。true为启用，false为禁用。  
ssh\_pwauth: true
3. 修改disable\_root为false。该参数表示是否禁用远程ssh root登录。  
disable\_root: false
4. 添加preserve\_hostname: false。  
preserve\_hostname: false
5. 使用“#”注释掉以下语句：  
mount\_default\_fields: [~, ~, 'auto', 'defaults,nofail', '0', '2']  
resize\_rootfs\_tmp: /dev  
ssh\_deletekeys: 0
6. 修改ssh\_genkeytypes为下面语句：  
ssh\_genkeytypes: ['rsa', 'dsa']
7. 修改syslog\_fix\_perms为下面语句：  
syslog\_fix\_perms: root:root
8. 添加下面语句：  
network:  
  config: disabled  
datasource\_list: [ OpenStack ]

## 须知

对于Ubuntu 14.04操作系统，不需要添加此行：

```
datasource_list: [OpenStack]
```

9. 在cloud\_final\_modules段的“- final-message”后面添加以下语句：

```
- power-state-change
```

10. 查看并修改system info中的信息，保持与下面内容一致：

```
system_info:
```

```
default_user:
```

```
name: root //登录操作系统使用的用户名
```

```
lock_passwd: False //True表示禁用密码登录方式，注意部分操作系统此处配置为1表示禁用
```

```
gecos: redhat
```

```
groups: [audio, cdrom, dialout, floppy] //可选项，将用户添加到其他组，groups必须是操作系统etc/group中已存在的组
```

```
sudo: ["ALL=(ALL) NOPASSWD:ALL"] //表示当前用户具有root用户的所有权限
```

```
shell: /bin/bash //shell执行采用bash方式
```

```
distro: sles
```

```
paths:
```

```
cloud_dir: /var/lib/cloud/
```

```
templates_dir: /etc/cloud/templates/
```

```
upstart_dir: /etc/init/
```

```
ssh_svcname: sshd
```

其中，distro参数需要根据具体操作系统类型修改，比如distro: sles、distro: rhel、distro: ubuntu、distro: debian、distro: fedora等。

11. （可选）对于Ubuntu 14.04操作系统，还需要执行以下操作：

- a. 使用vi编辑器打开“/etc/init/cloud-init-local.conf”配置文件，修改如下配置项：

```
cloud-init - the initial cloud-init job
```

```
crawls metadata service, emits cloud-config
```

```
start on mounted MOUNTPPOINT=/ and mounted MOUNTPPOINT=/run and stopped bms-network-config
```

- b. 执行以下命令，配置OpenStack源。

**dpkg-reconfigure cloud-init**

```
Configuring cloud-init
Cloud-init supports searching different "Data Sources" for information that it uses to configure a cloud instance.
Warning: Only select 'Ec2' if this system will be run on a system with the EC2 metadata service present. Doing so
incorrectly will result in a substantial timeout on boot.

Which data sources should be searched?

[] NoCloud: Reads info from /var/lib/cloud/seed only
[] ConfigDrive: Reads data from Openstack Config Drive
[] OpenNebula: read from OpenNebula context disk
[] DigitalOcean: reads data from Droplet datasource
[] Azure: read from MS Azure cdrom. Requires waiinux-agent
[] AltCloud: config disks for RHEUm and vSphere
[] OVF: Reads data from OVF Transports
[] MAAS: Reads data from Ubuntu MAAS
[] GCE: google compute metadata service
[*] OpenStack: native openstack metadata service
[] CloudSigma: metadata over serial for cloudsigma.com
[] SmartOS: Read from SmartOS metadata service
[] Bigstep: Bigstep metadata service
[] Scaleway: Scaleway metadata service
[] Aliyun: Alibaba metadata service
[] Ec2: reads data from EC2 Metadata service
[] CloudStack: Read from CloudStack metadata service
[] None: Failsafe datasource

<OK>
```

执行命令**vim /etc/cloud/cloud.cfg.d/90\_dpkg.cfg**打开配置文件，检查配置项是否正确。

```
to update this file, run dpkg-reconfigure cloud-init
```

```
datasource_list: [OpenStack]
```

```
~
~
~
```

配置文件和以上回显信息一致，表示配置成功。

### 3.8.2 cloud-init-0.7.6 配置

1. 添加如下键值对，保持前后各空出一行：  
`no_ssh_fingerprints: true`
  2. 修改users为default：  
`users:`  
`- default`
  3. 修改以下字段内容：  
`disable_root: False`  
`preserve_hostname: false`  
`syslog_fix_perms: root:root`
  4. 添加ssh\_pwauth设置。该选项表示是否支持ssh密码登录。true为启用，false为禁用。  
`ssh_pwauth: true`
  5. 使用“#”注释掉以下语句：  
`mount_default_fields: [~, ~, 'auto', 'defaults', '0', '2']`  
`manual_cache_clean: true`
  6. 添加network语句：  
`network:`  
`config: disabled`  
`datasource_list: [ OpenStack ]`
- 对于SUSE 11 SP4操作系统，还需要添加growpart语句：
- ```
growpart:
mode: false
```

须知

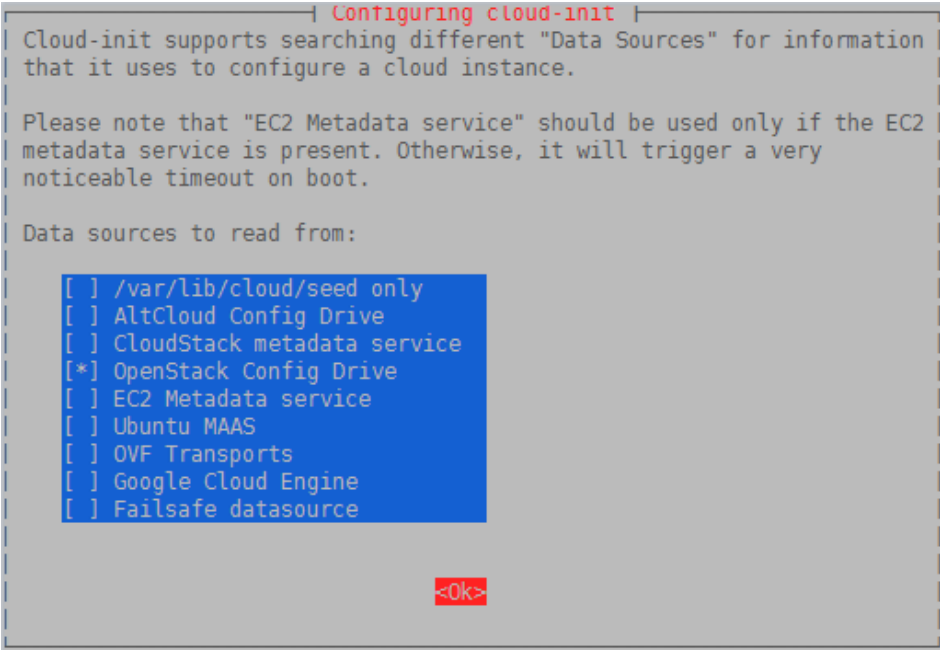
对于Debian 8.6操作系统，不需要添加此行：

```
datasource_list: [ OpenStack ]
```

7. 在cloud_final_modules段的“- final-message”后面添加以下语句：
`- power-state-change`
8. 查看并修改system info中的信息，保持与下面内容一致：
`system_info:`
`distro: sles`
`default_user:`
`name: root` //登录操作系统使用的用户名
`lock_passwd: False` //True表示禁用密码登录方式，注意部分操作系统此处配置为1表示禁用
`gecos: redhat`
`groups: [adm, audio, cdrom, dialout, dip, floppy, lxd, netdev, plugdev, sudo, video]` //可选项，将用户添加到其他组，groups必须是操作系统etc/group中已存在的组
`sudo: ["ALL=(ALL) NOPASSWD:ALL"]` //表示当前用户具有root用户的所有权限
`shell: /bin/bash` //shell执行采用bash方式
`paths:`
`cloud_dir: /var/lib/cloud/`
`templates_dir: /etc/cloud/templates/`
`ssh_svcname: sshd`

其中，distro参数需要根据具体操作系统类型修改，比如distro: sles、distro: rhel、distro: ubuntu、distro: debian、distro: fedora等。
9. 用“#”注释掉下面一行：
`ssh_genkeytypes: ['rsa', 'dsa']`
10. （可选）对于Debian 8.6操作系统，还需要执行以下操作：
 - a. 执行以下命令，配置OpenStack源。

dpkg-reconf igure cloud-init

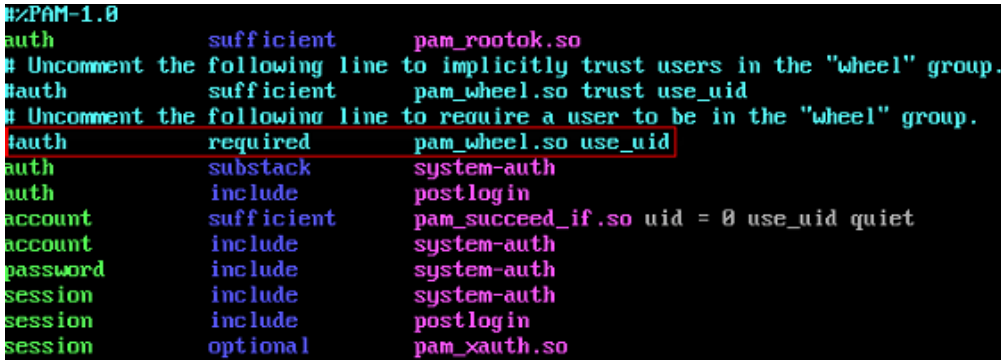


执行命令`vim /etc/cloud/cloud.cfg.d/90_dpkg.cfg`打开配置文件，检查配置项是否正确。

```
# to update this file, run dpkg-reconf igure cloud-init
datasource_list: [ OpenStack ]
~
~
~
```

配置文件和以上回显信息一致，表示配置成功。

11. （可选）对于EulerOS 2.2/EulerOS 2.3操作系统，还需要执行以下操作：
确认“/etc/pam.d/su”配置文件是否已通过“#”注释图示所在行，如果已经注释，则跳过此步骤。



3.8.3 cloud-init-0.7.9 及以上版本配置

- 1. 添加如下键值对，前后各空出一行：
no_ssh_fingerprints: true
- 2. 设置ssh_pwauth，该选项表示是否支持ssh密码登录。true为启用，false或0为禁用。
ssh_pwauth: true
- 3. 修改disable_root为false。该参数表示是否禁用远程ssh root登录。
disable_root: false

4. 添加preserve_hostname: false。
preserve_hostname: false
5. 使用“#”注释掉以下语句（不存在忽略）：
mount_default_fields: [~, ~, 'auto', 'defaults,nofail', '0', '2']
resize_rootfs_tmp: /dev
ssh_deletekeys: 0
6. 修改ssh_genkeytypes为下面语句（不存在添加）：
ssh_genkeytypes: ['rsa', 'dsa']
7. 修改syslog_fix_perms为下面语句（不存在添加）：
syslog_fix_perms: root:root
8. 添加下面语句：
network:
 config: disabled
datasource_list: [OpenStack]
datasource:
 OpenStack:
 metadata_urls: ['http://169.254.169.254']
 max_wait: 120
 timeout: 10
 retries: 5
9. （可选）在“/etc/cloud/cloud.cfg”中配置“apply_network_config: False”。
对于使用Cloud-Init 18.3及以上版本的用户，需执行本操作

```
network:  
  config: disabled  
datasource_list: [ OpenStack ]  
datasource:  
  OpenStack:  
    metadata_urls: ['http://169.254.169.254']  
    max_wait: 120  
    timeout: 10  
    retries: 5  
  apply_network_config: False
```

10. 在cloud_final_modules段的“- final-message”后面添加一句：
- power-state-change
11. 查看并修改system info中的信息：
system_info:
 distro: rhel
 default_user:
 name: root //登录操作系统使用的用户名
 lock_passwd: False //True表示禁用密码登录方式，注意部分操作系统此处配置为1表示禁用

其中，distro参数需要根据具体操作系统类型修改，比如distro: sles、distro: rhel、distro: ubuntu、distro: debian、distro: fedora等。

12. 除cloud-init 23.x以外的版本，修改如下源码：
源码路径：xxx/xxx/cloudinit/sources/DataSourceOpenStack.py
- 新增“return True”行。
- 删除“(DataSourceOpenStackLocal, (sources.DEP_FILESYSTEM,))”行。
def detect_openstack(accept_oracle=False):
 """Return True when a potential OpenStack platform is detected."""
 return True
 if not util.is_x86():

```
        return True # Non-Intel cpus don't properly report dmi product names
        product_name = util.read_dmi_data('system-product-name')
.....
# Used to match classes to dependencies
datasources = [
    (DataSourceOpenStack, (sources.DEP_FILESYSTEM, sources.DEP_NETWORK)),
]
```

13. 针对**cloud-init 23.x**版本及以上版本，修改如下源码：

- 源码路径：xxx/xxx/cloudinit/sources/DataSourceOpenStack.py，删除“(DataSourceOpenStackLocal, (sources.DEP_FILESYSTEM,))”行。

```
# Used to match classes to dependencies
datasources = [
    (DataSourceOpenStack, (sources.DEP_FILESYSTEM, sources.DEP_NETWORK)),
]
```

- 源码路径：xxx/xxx/cloudinit/sources/__init__.py，修改“return_value = self.check_and_get_data()”为“return_value = self.get_data()”。

```
def get_data(self) -> bool:
    """Datasources implement _get_data to setup metadata and userdata_raw.
    Minimally, the datasource should return a boolean True on success.
    """
    self.dirty_cache = True
    return_value = self.get_data()
    if not return_value:
        return return_value
    self.persist_instance_data()
    return return_value
```

14. （可选）对于SUSE 12 SP1和SUSE 12 SP2操作系统，还需要修改“/usr/lib/systemd/system/cloud-init-local.service”配置文件的[Unit]配置信息。

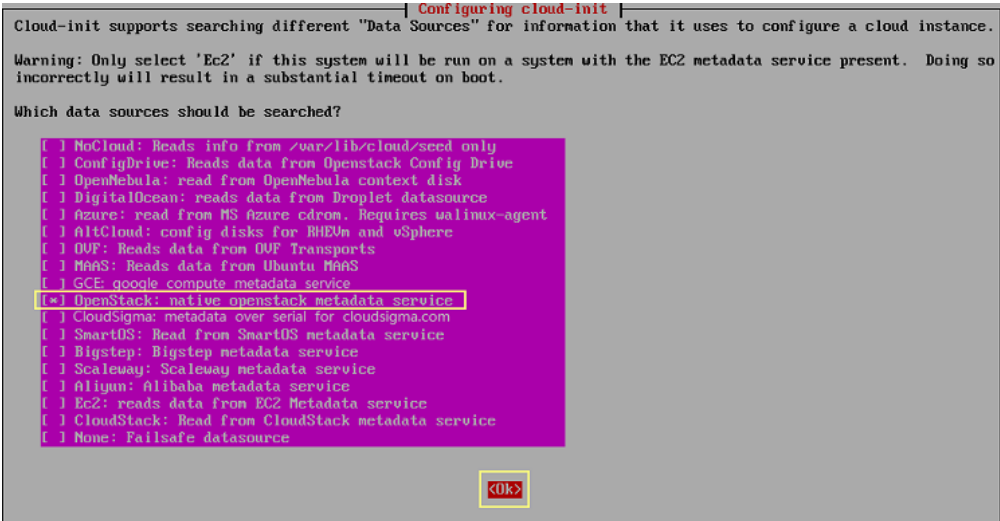
vi /usr/lib/systemd/system/cloud-init-local.service

保持[Unit]配置项与下面内容一致：

```
[Unit]
Description=Initial cloud-init job (pre-networking)
DefaultDependencies=no
Wants=network-pre.target
Wants=local-fs.target
After=local-fs.target
Before=network-pre.target
Before=shutdown.target
Before=basic.target
Conflicts=shutdown.target
# Other distros use Before=sysinit.target. There is not a clearly identified
# reason for usage of basic.target instead.
```

15. （可选）对于Ubuntu 16.04操作系统，还需要执行以下命令，配置OpenStack源。

dpkg-reconfigure cloud-init



执行命令`vim /etc/cloud/cloud.cfg.d/90_dpkg.cfg`打开配置文件，检查配置项是否正确。

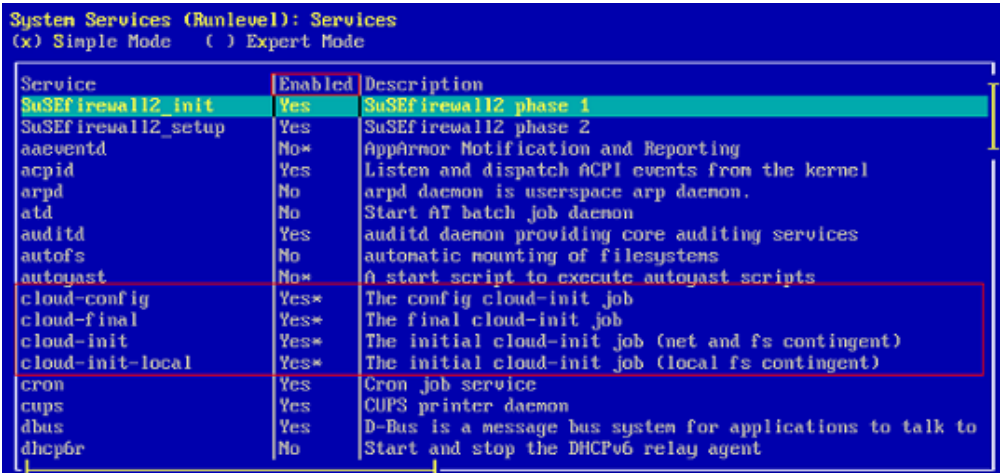
```
# to update this file, run dpkg-reconfigure cloud-init
datasource_list: [ OpenStack ]
~
~
~
```

配置文件和以上回显信息一致，表示配置成功。

3.9 查看 Cloud-Init 服务状态

3.9.1 SUSE 11 SP4

- 1. 执行命令：**yast**，使用上下键选择“System”。
- 2. 按“Tab”键后，然后选择“Sytem Services (Runlevel)”。
- 3. 按“Enter”，显示如下图，可以看到Cloud-Init的四个阶段的服务已被自启动，其中“YES”为自启动，“No”为没有被自启动。如果没有自启动，按照图中的功能键将其添加到自启动中。



3.9.2 SUSE 12 SP1

1. 执行以下命令，检查Cloud-Init开机自启动服务是否设置成功。
查看Cloud-Init服务状态，执行**yast**命令，使用上下键，选择“System”，然后按“Tab”键，使用上下键选择“System Manager”。

| Service | Enabled | Active | Description |
|---------------------------|----------|----------|------------------------|
| auditd | Disabled | Inactive | Security Auditing Serv |
| bttrfsmaintenance-refresh | Enabled | Inactive | Update cron periods fr |
| cloud-config | Enabled | Active | Apply the settings spe |
| cloud-final | Enabled | Active | Execute cloud user/fin |
| cloud-init-local | Enabled | Inactive | Initial cloud-init job |
| cloud-init | Enabled | Active | Initial cloud-init job |
| cron | Enabled | Active | Command Scheduler |
| dm-event | Enabled | Active | Device-mapper event da |
| getty@tty1 | Enabled | Active | Getty on tty1 |
| haveged | Enabled | Active | Entropy Daemon based o |
| ipmi | Disabled | Inactive | LSB: OpenIPMI Driver i |
| irqbalance | Enabled | Active | irqbalance daemon |
| iscsi | Enabled | Active | Login and scanning of |
| iscsid | Disabled | Inactive | Open-iSCSI |
| iscsiuio | Disabled | Inactive | iSCSI UserSpace I/O dr |
| kdump | Enabled | Active | Load kdump kernel on s |
| lun2-lunetad | Disabled | Inactive | LVM2 metadata daemon |
| ncelog | Disabled | Inactive | Machine Check Exceptio |
| multipathd | Disabled | Inactive | Device-Mapper Multipat |
| network-config | Enabled | Inactive | Network Config |
| nfs-server | Disabled | Inactive | NFS server and service |
| nfs | Disabled | Inactive | Alias for NFS client |
| nfsserver | Disabled | Inactive | Alias for NFS server |
| nscd | Enabled | Active | Name Service Cache Dae |
| plymouth-quit-wait | Disabled | Inactive | Wait for Plymouth Boot |
| plymouth-quit | Disabled | Inactive | Terminate Plymouth Boo |
| plymouth-read-write | Disabled | Inactive | Tell Plymouth To Write |
| plymouth-start | Disabled | Active | Show Plymouth Boot Scr |
| postfix | Enabled | Active | Postfix Mail Transport |
| purge-kernels | Enabled | Inactive | Purge old kernels |
| rollback | Enabled | Inactive | Rollback Helper for Re |
| rpcbind | Disabled | Inactive | RPC Bind |
| rsyslog | Enabled | Active | System Logging Service |
| serial-getty@ttyS0 | Disabled | Inactive | Serial Getty on ttyS0 |
| smartd | Enabled | Inactive | Self Monitoring and Re |

[Start/Stop] [Enable/Disable]

[Show Details]

2. 可以看到上图中Cloud-Init的四个服务均为“enable”，表示已经自启动服务。
active表示处于开启状态。对于Cloud-Init四个阶段的服务，如果某个服务没有自启动，首先用上下键选中，然后按“Tab”，选中“Enable/Disable”，即可设置该服务为自启动服务。

3.9.3 SUSE 12 SP2/SUSE 12 SP3/SUSE 15/Oracle Linux 7 系列/Red Hat 7 系列/CentOS 7 系列/CentOS 8 系列

1. 执行以下命令，检查Cloud-Init开机自启动服务是否设置成功。
systemctl status cloud-init-local
systemctl status cloud-init
systemctl status cloud-config
systemctl status cloud-final
2. 下图所示回显信息“enabled”，说明开机自启动服务已经设置成功。

```
[root@localhost r74]# service cloud-init-local status
Redirecting to /bin/systemctl status cloud-init-local.service
● cloud-init-local.service - Initial cloud-init job (pre-networking)
   Loaded: loaded (/usr/lib/systemd/system/cloud-init-local.service; enabled; vendor preset: disabled)
   Active: inactive (dead)
[root@localhost r74]# service cloud-init status
Redirecting to /bin/systemctl status cloud-init.service
● cloud-init.service - Initial cloud-init job (metadata service crawler)
   Loaded: loaded (/usr/lib/systemd/system/cloud-init.service; enabled; vendor preset: disabled)
   Active: inactive (dead)
[root@localhost r74]# service cloud-config status
Redirecting to /bin/systemctl status cloud-config.service
● cloud-config.service - Apply the settings specified in cloud-config
   Loaded: loaded (/usr/lib/systemd/system/cloud-config.service; enabled; vendor preset: disabled)
   Active: inactive (dead)
[root@localhost r74]# service cloud-final status
Redirecting to /bin/systemctl status cloud-final.service
● cloud-final.service - Execute cloud user/final scripts
   Loaded: loaded (/usr/lib/systemd/system/cloud-final.service; enabled; vendor preset: disabled)
   Active: inactive (dead)
```

如果没有设置开机自启动，则执行以下命令开启对应阶段的开机自启动服务：

systemctl enable cloud-init-local

systemctl enable cloud-init

systemctl enable cloud-config

systemctl enable cloud-final

3. 执行以下命令，运行Cloud-Init服务命令。

systemctl start cloud-init-local

systemctl start cloud-init

systemctl start cloud-config

systemctl start cloud-final

4. 执行步骤1中命令，查看Cloud-Init状态是否为active，如图。

```
[root@localhost ~]# systemctl start cloud-init-local
[root@localhost ~]# systemctl start cloud-init
[root@localhost ~]# systemctl start cloud-config
[root@localhost ~]# systemctl start cloud-final
[root@localhost ~]# systemctl status cloud-init-local
● cloud-init-local.service - Initial cloud-init job (pre-networking)
   Loaded: loaded (/usr/lib/systemd/system/cloud-init-local.service; enabled; vendor preset: disabled)
   Active: active (exited) since Tue 2022-10-18 14:15:13 CST; 3min 12s ago
   Process: 15945 ExecStart=/usr/bin/cloud-init init --local (code=exited, status=0/SUCCESS)
   Main PID: 15945 (code=exited, status=0/SUCCESS)
   CGroup: /system.slice/system-hostos.slice/cloud-init-local.service
[root@localhost ~]# systemctl status cloud-init
● cloud-init.service - Initial cloud-init job (metadata service crawler)
   Loaded: loaded (/usr/lib/systemd/system/cloud-init.service; enabled; vendor preset: disabled)
   Active: active (exited) since Tue 2022-10-18 14:15:14 CST; 3min 28s ago
   Process: 15974 ExecStart=/usr/bin/cloud-init init (code=exited, status=0/SUCCESS)
   Main PID: 15974 (code=exited, status=0/SUCCESS)
   CGroup: /system.slice/system-hostos.slice/cloud-init.service
[root@localhost ~]# systemctl status cloud-config
● cloud-config.service - Apply the settings specified in cloud-config
   Loaded: loaded (/usr/lib/systemd/system/cloud-config.service; enabled; vendor preset: disabled)
   Active: active (exited) since Tue 2022-10-18 14:15:14 CST; 3min 36s ago
   Process: 16019 ExecStart=/usr/bin/cloud-init modules --mode=config (code=exited, status=0/SUCCESS)
   Main PID: 16019 (code=exited, status=0/SUCCESS)
   CGroup: /system.slice/system-hostos.slice/cloud-config.service
[root@localhost ~]# systemctl status cloud-final
● cloud-final.service - Execute cloud user/final scripts
   Loaded: loaded (/usr/lib/systemd/system/cloud-final.service; enabled; vendor preset: disabled)
   Active: active (exited) since Tue 2022-10-18 14:15:14 CST; 3min 53s ago
   Process: 16025 ExecStart=/usr/bin/cloud-init modules --mode=final (code=exited, status=0/SUCCESS)
   Main PID: 16025 (code=exited, status=0/SUCCESS)
```

3.9.4 EulerOS/OpenEuler

1. 执行以下命令，检查Cloud-Init开机自启动服务是否设置成功。

systemctl status cloud-init-local

systemctl status cloud-init

systemctl status cloud-config

systemctl status cloud-final

- 下图所示回显信息“enabled”，说明开机自启动服务已经设置成功。

```
[root@localhost ~]# systemctl status cloud-init
● cloud-init.service - Initial cloud-init job (metadata service crawler)
   Loaded: loaded (/usr/lib/systemd/system/cloud-init.service; enabled; vendor preset: disabled)
   Active: inactive (dead)
[root@localhost ~]# systemctl status cloud-init-local
● cloud-init-local.service - Initial cloud-init job (pre-networking)
   Loaded: loaded (/usr/lib/systemd/system/cloud-init-local.service; enabled; vendor preset: disabled)
   Active: inactive (dead)
[root@localhost ~]# systemctl status cloud-config
● cloud-config.service - Apply the settings specified in cloud-config
   Loaded: loaded (/usr/lib/systemd/system/cloud-config.service; enabled; vendor preset: disabled)
   Active: inactive (dead)
[root@localhost ~]# systemctl status cloud-final
● cloud-final.service - Execute cloud user/final scripts
   Loaded: loaded (/usr/lib/systemd/system/cloud-final.service; enabled; vendor preset: disabled)
   Active: inactive (dead)
[root@localhost ~]#
```

如果没有设置开机自启动，则执行以下命令开启对应阶段的开机自启动服务：

systemctl enable cloud-init-local

systemctl enable cloud-init

systemctl enable cloud-config

systemctl enable cloud-final

- 执行以下命令，运行Cloud-Init服务命令。

systemctl start cloud-init-local

systemctl start cloud-init

systemctl start cloud-config

systemctl start cloud-final

- 执行步骤1中命令，查看Cloud-Init状态是否为active，如图。

```
[root@localhost ~]# systemctl start cloud-init-local
[root@localhost ~]# systemctl start cloud-init
[root@localhost ~]# systemctl start cloud-config
[root@localhost ~]# systemctl start cloud-final
[root@localhost ~]# systemctl status cloud-init-local
● cloud-init-local.service - Initial cloud-init job (pre-networking)
   Loaded: loaded (/usr/lib/systemd/system/cloud-init-local.service; enabled; vendor preset: disabled)
   Active: active (exited) since Tue 2022-10-18 14:15:13 CST; 3min 12s ago
   Process: 15945 ExecStart=/usr/bin/cloud-init init --local (code=exited, status=0/SUCCESS)
   Main PID: 15945 (code=exited, status=0/SUCCESS)
   CGroup: /system.slice/system-hostos.slice/cloud-init-local.service
[root@localhost ~]# systemctl status cloud-init
● cloud-init.service - Initial cloud-init job (metadata service crawler)
   Loaded: loaded (/usr/lib/systemd/system/cloud-init.service; enabled; vendor preset: disabled)
   Active: active (exited) since Tue 2022-10-18 14:15:14 CST; 3min 28s ago
   Process: 15974 ExecStart=/usr/bin/cloud-init init (code=exited, status=0/SUCCESS)
   Main PID: 15974 (code=exited, status=0/SUCCESS)
   CGroup: /system.slice/system-hostos.slice/cloud-init.service
[root@localhost ~]# systemctl status cloud-config
● cloud-config.service - Apply the settings specified in cloud-config
   Loaded: loaded (/usr/lib/systemd/system/cloud-config.service; enabled; vendor preset: disabled)
   Active: active (exited) since Tue 2022-10-18 14:15:14 CST; 3min 36s ago
   Process: 16019 ExecStart=/usr/bin/cloud-init modules --mode=config (code=exited, status=0/SUCCESS)
   Main PID: 16019 (code=exited, status=0/SUCCESS)
   CGroup: /system.slice/system-hostos.slice/cloud-config.service
[root@localhost ~]# systemctl status cloud-final
● cloud-final.service - Execute cloud user/final scripts
   Loaded: loaded (/usr/lib/systemd/system/cloud-final.service; enabled; vendor preset: disabled)
   Active: active (exited) since Tue 2022-10-18 14:15:14 CST; 3min 53s ago
   Process: 16025 ExecStart=/usr/bin/cloud-init modules --mode=final (code=exited, status=0/SUCCESS)
   Main PID: 16025 (code=exited, status=0/SUCCESS)
```

3.9.5 Red Hat 6 系列/CentOS 6 系列/Oracle Linux 6 系列

- 执行下列命令：

chkconfig --list | grep cloud

如下图，若红框部分处于“on”状态，则表示服务已被自启动。

```
(root@localhost r69)# chkconfig --list | grep cloud
cloud-config      0:off  1:off  2:on   3:on   4:on   5:on   6:off
cloud-final       0:off  1:off  2:on   3:on   4:on   5:on   6:off
cloud-init        0:off  1:off  2:on   3:on   4:on   5:on   6:off
cloud-init-local  0:off  1:off  2:on   3:off  4:on   5:on   6:off
(root@localhost r69)#
```

2. 如果Cloud-Init四个阶段的服务未被自启动，在命令行执行：

```
chkconfig cloud-init on
chkconfig cloud-init-local on
chkconfig cloud-config on
chkconfig cloud-final on
```

3.9.6 Ubuntu 16.04/Ubuntu 18.04

1. 执行下列命令：

```
systemctl status cloud-init
systemctl status cloud-init-local
systemctl status cloud-config
systemctl status cloud-final
```

如下图，若红框部分是“enable”状态，则表示服务已被自启动。

```
root@ubuntu:/tmp/deb# systemctl status cloud-init
* cloud-init.service - Initial cloud-init job (metadata service crawler)
   Loaded: loaded (/lib/systemd/system/cloud-init.service; enabled; vendor preset: enabled)
   Active: inactive (dead)
root@ubuntu:/tmp/deb# systemctl status cloud-init-local
* cloud-init-local.service - Initial cloud-init job (pre-networking)
   Loaded: loaded (/lib/systemd/system/cloud-init-local.service; enabled; vendor preset: enabled)
   Active: inactive (dead)
root@ubuntu:/tmp/deb# systemctl status cloud-config
* cloud-config.service - Apply the settings specified in cloud-config
   Loaded: loaded (/lib/systemd/system/cloud-config.service; enabled; vendor preset: enabled)
   Active: inactive (dead)
root@ubuntu:/tmp/deb# systemctl status cloud-final
* cloud-final.service - Execute cloud user/final script,
   Loaded: loaded (/lib/systemd/system/cloud-final.service; enabled; vendor preset: enabled)
   Active: inactive (dead)
root@ubuntu:/tmp/deb#
```

2. 如果Cloud-Init四个阶段的服务未被自启动，在命令行执行：

```
systemctl enable cloud-init
systemctl enable cloud-init-local
systemctl enable cloud-config
systemctl enable cloud-final
```

3. 执行以下命令，运行Cloud-Init服务命令。

```
systemctl start cloud-init-local
systemctl start cloud-init
systemctl start cloud-config
systemctl start cloud-final
```

4. 执行步骤1中命令，查看Cloud-Init状态是否为active，如图。


```
[root@localhost ~]# systemctl start cloud-init-local
[root@localhost ~]# systemctl start cloud-init
[root@localhost ~]# systemctl start cloud-config
[root@localhost ~]# systemctl start cloud-final
[root@localhost ~]# systemctl status cloud-init-local
● cloud-init-local.service - Initial cloud-init job (pre-networking)
   Loaded: loaded (/usr/lib/systemd/system/cloud-init-local.service; enabled; vendor preset: disabled)
   Active: active (exited) since Tue 2022-10-18 14:15:13 CST; 3min 12s ago
   Process: 15945 ExecStart=/usr/bin/cloud-init init --local (code=exited, status=0/SUCCESS)
   Main PID: 15945 (code=exited, status=0/SUCCESS)
   CGroup: /system.slice/system-hostos.slice/cloud-init-local.service
[root@localhost ~]# systemctl status cloud-init
● cloud-init.service - Initial cloud-init job (metadata service crawler)
   Loaded: loaded (/usr/lib/systemd/system/cloud-init.service; enabled; vendor preset: disabled)
   Active: active (exited) since Tue 2022-10-18 14:15:14 CST; 3min 28s ago
   Process: 15974 ExecStart=/usr/bin/cloud-init init (code=exited, status=0/SUCCESS)
   Main PID: 15974 (code=exited, status=0/SUCCESS)
   CGroup: /system.slice/system-hostos.slice/cloud-init.service
[root@localhost ~]# systemctl status cloud-config
● cloud-config.service - Apply the settings specified in cloud-config
   Loaded: loaded (/usr/lib/systemd/system/cloud-config.service; enabled; vendor preset: disabled)
   Active: active (exited) since Tue 2022-10-18 14:15:14 CST; 3min 36s ago
   Process: 16019 ExecStart=/usr/bin/cloud-init modules --mode=config (code=exited, status=0/SUCCESS)
   Main PID: 16019 (code=exited, status=0/SUCCESS)
   CGroup: /system.slice/system-hostos.slice/cloud-config.service
[root@localhost ~]# systemctl status cloud-final
● cloud-final.service - Execute cloud user/final scripts
   Loaded: loaded (/usr/lib/systemd/system/cloud-final.service; enabled; vendor preset: disabled)
   Active: active (exited) since Tue 2022-10-18 14:15:14 CST; 3min 53s ago
   Process: 16025 ExecStart=/usr/bin/cloud-init modules --mode=final (code=exited, status=0/SUCCESS)
   Main PID: 16025 (code=exited, status=0/SUCCESS)
```

3.9.7 Ubuntu 14.04

执行下列命令：

```
initctl status cloud-init
```

```
initctl status cloud-init-local
```

```
initctl status cloud-config
```

```
initctl status cloud-final
```

如果安装成功，则回显Cloud-Init组件安装信息。

```
[root@ubuntu:~]# initctl status cloud-init
cloud-init stop/waiting
[root@ubuntu:~]# initctl status cloud-init-local
cloud-init-local stop/waiting
[root@ubuntu:~]# initctl status cloud-config
cloud-config stop/waiting
[root@ubuntu:~]# initctl status cloud-final
cloud-final stop/waiting
```

3.10 修改引导的硬件设备驱动

操作场景

此操作主要是在系统启动阶段，针对加载的实际物理服务器的硬件设备驱动所进行的。

前提条件

已登录虚拟机。

操作步骤

1. 增加或修改驱动。

- 对于Red Hat系列/Oracle Linux系列/EulerOS/OpenEuler/SUSE 12/SUSE 15/CentOS系列操作系统，使用vi编辑器打开“/etc/dracut.conf”文件，修改或添加“add_drivers”的取值，如下所示，add_drivers+="服务器的RAID卡驱动”。

```
logfile=/var/log/dracut.log
# fileloglvl=7
...
...
# additional kernel modules to the default
add_drivers+=" ahci megaraid_sas mpt3sas mpt2sas virtio_blk virtio_scsi virtio_net "
```

说明

如果有报错，请在引号内前后增加空格，再次尝试。

- 对于Ubuntu 14.04/Debian操作系统，使用vi编辑器打开“/etc/initramfs-tools/modules”文件，添加ahci、megaraid_sas、mpt3sas和mpt2sas驱动（具体格式需要根据操作系统本身的要求决定）。

```
# List of modules that you want to include in your initramfs.
# They will be loaded at boot time in the order below.
#
# Syntax: module_name [args ...]
#
# You must run update-initramfs(8) to effect this change.
#
# Examples:
#
# raid1
# sd_mod
ahci
megaraid_sas
mpt3sas
mpt2sas
virtio_blk
virtio_scsi
virtio_net
```

- 对于Ubuntu 16.04/Ubuntu 18.04操作系统，需要分别在“/etc/dracut.conf”和“/etc/initramfs-tools/modules”文件中添加驱动。但是，在编辑配置文件之前，需要安装相关软件：

- i. 由于该操作系统默认不带dracut软件，执行以下命令安装dracut软件：

apt-get install dracut

安装完成后，按照Red Hat系列/Oracle Linux 7.3等操作系统的方法，编辑“/etc/dracut.conf”文件，在末尾添加**add_drivers+="ahci megaraid_sas mpt3sas mpt2sas virtio_blk virtio_scsi virtio_net"**。

说明

如果有报错，请在引号内前后增加空格，再次尝试。

- ii. 由于该操作系统默认不带initramfs-tools软件，执行以下命令进行安装：

apt-get install initramfs-tools

安装完成后，按照Ubuntu 14.04操作系统的方法，编辑“/etc/initramfs-tools/modules”文件，在末尾添加ahci、megaraid_sas、mpt3sas、mpt2sas和virtio_blk virtio_scsi virtio_net驱动。

- 对于Ubuntu 16.04 ARM，还需要执行以下命令更新内核和驱动，之后再重启虚拟机。

sudo apt-get update

sudo apt-get dist-upgrade

- 对于SUSE 11 SP4操作系统，使用vi编辑器打开“/etc/sysconfig/kernel”文件，修改或添加“INITRD_MODULES”的取值，如下所示，INITRD_MODULES="服务器的RAID卡驱动"。

```
...
#
INITRD_MODULES=" ahci megaraid_sas mpt3sas mpt2sas virtio_blk virtio_scsi virtio_net "
## Type: string(yes)
...
```

📖 说明

服务器的RAID卡驱动可以填写多个，以空格分开。驱动信息可以根据采购的硬件设备获取RAID相关的驱动名称。镜像支持多种类型的驱动并存，如“mpt3sas”、“mpt2sas”和“megaraid_sas”就是3种不同的RAID卡型号。当操作系统不支持某些硬件的驱动时，需要额外安装硬件驱动。

2. 刷新内核，对于Ubuntu 16.04 ARM、Ubuntu 18.04 ARM，则跳过本步骤。

Rad Hat系列/Oracle Linux系列/EulerOS/SUSE 12/SUSE 15/Ubuntu 16.04/Ubuntu 18.04/CentOS系列刷新命令：**dracut -f**

- 对于Rad Hat系列/Oracle Linux系列/EulerOS/OpenEuler/CentOS系列，执行命令**dracut -f**，等待数秒后，如果命令执行完成不产生回显信息，说明加载成功。
- 对于SUSE 12 SP1，执行命令**dracut -f**，观察最后几行回显信息，显示信息提示“有些内核模块没有被加载，但不一定是一个错误”，可以看到回显信息中提到的没有加载的驱动，不包括上面手动加载的驱动，说明RAID卡驱动加载成功。

```
...
Some kernel modules could not be included
This is not necessarily an error:
pcmcia
sdhci_acpi
swap
```

- 对于SUSE 12/SUSE 15，执行命令**dracut -f**，结果如下图，表示刷新成功。

```
dracut: Skipping udev rule: 50-firmware.rules
dracut: Skipping udev rule: 50-udev.rules
dracut: Skipping udev rule: 91-permissions.rules
dracut: Skipping udev rule: 80-drivers-modprobe.rules
dracut: *** Including module: dracut-systemd ***
dracut: *** Including module: haveged ***
dracut: *** Including module: usrmount ***
dracut: *** Including module: base ***
dracut: *** Including module: fs-lib ***
dracut: *** Including module: shutdown ***
dracut: *** Including module: suse ***
dracut: *** Including modules done ***
dracut: *** Installing kernel module dependencies and firmware ***
dracut: *** Installing kernel module dependencies and firmware done ***
dracut: *** Resolving executable dependencies ***
dracut: *** Resolving executable dependencies done***
dracut: *** Hardlinking files ***
dracut: *** Hardlinking files done ***
dracut: *** Stripping files ***
dracut: *** Stripping files done ***
dracut: *** Generating early-microcode cpio image ***
dracut: *** Store current command line parameters ***
dracut: Stored kernel cmdline:
dracut: resume=UUID=0f55f21f-0b6d-45bf-b376-8d3136a7fd1e
dracut: root=UUID=8181e24a-3e1d-4195-b09f-710099935a20 rootfstype=ext4 rootflags=rw,relatime,data=ordered
dracut: *** Creating image file '/boot/initrd-4.4.21-69-default' ***
dracut: *** Creating initramfs image file '/boot/initrd-4.4.21-69-default' done ***
linux-96xx:/home/fsp #
```

- 对于Ubuntu 14.04/Ubuntu 16.04/Ubuntu 18.04/Debian，执行以下命令重新生成initrd。

update-initramfs -u

依次执行以下命令，检查是否已经成功装载了ahci、megaraid_sas、mpt3sas和mpt2sas驱动的相应模块。

```
lsinitramfs /boot/initrd.img-`uname -r` |grep ahci
lsinitramfs /boot/initrd.img-`uname -r` |grep megaraid_sas
lsinitramfs /boot/initrd.img-`uname -r` |grep mpt3sas
lsinitramfs /boot/initrd.img-`uname -r` |grep mpt2sas
```

- 对于SUSE 11 SP4，刷新命令：**mkinitrd**，查看Kernel Modules是否加载上了上面的驱动，如下图所示，表示已经成功加载。

```
linux-dann:~ # mkinitrd

Kernel image: /boot/vmlinuz-3.0.101-63-default
Initrd image: /boot/initrd-3.0.101-63-default
KMS drivers: intel-agg cirrus
Root device: /dev/sda2 (mounted on / as ext3)
Resume device: /dev/sda1
modprobe: Module crc_t10dif not found.
WARNING: no dependencies for kernel module 'crc_t10dif' found.
Kernel Modules: hwmon thermal_sys thermal processor fan scsi_mod libata libahci ahci megaraid_sas raid_class s
csi_transport_sas mpt3sas mpt2sas scsi_dh scsi_dh_emc scsi_dh_rdac scsi_dh_alua scsi_dh_hp_sw mbcache jbd ext3
intel-gtt intel-agg syscopyarea sysfillrect sysimgblt i2c-core drm drm_kms_helper ttm cirrus ata_piix usb-com
mon usbcore ohci-hcd uhci-hcd ehci-hcd xhci-hcd hid usbhid crc_t10dif sd_mod
Features: acpi kms block usb resume.userspace resume.kernel
Bootsplash: SLES (800x600)
40842 blocks
>>> Network: auto
>>> Calling mkinitrd -k /boot/vmlinuz-3.0.101-63-default -i /tmp/mkdumprd.Kp2uLV38DM -f 'kdump network' -B -s
..
Regenerating kdump initrd ...

Kernel image: /boot/vmlinuz-3.0.101-63-default
Initrd image: /tmp/mkdumprd.Kp2uLV38DM
KMS drivers: intel-agg cirrus
Root device: /dev/sda2 (mounted on / as ext3)
Resume device: /dev/sda1
modprobe: Module crc_t10dif not found.
WARNING: no dependencies for kernel module 'crc_t10dif' found.
Kernel Modules: hwmon thermal_sys thermal processor fan scsi_mod libata libahci ahci megaraid_sas raid_class s
csi_transport_sas mpt3sas mpt2sas scsi_dh scsi_dh_emc scsi_dh_rdac scsi_dh_alua scsi_dh_hp_sw mbcache jbd ext3
intel-gtt intel-agg syscopyarea sysfillrect sysimgblt i2c-core drm drm_kms_helper ttm cirrus ata_piix usb-com
mon usbcore ohci-hcd uhci-hcd ehci-hcd xhci-hcd hid usbhid af_packet virtio_ring virtio virtio_net nls_utf8 cr
c_t10dif sd_mod virtio_pci
Features: acpi kms block usb network resume.userspace resume.kernel kdump
54488 blocks
```

3. 对于SUSE 11 SP4，还需将虚拟机文件中的虚拟磁盘更改为实际物理磁盘，方法如下（其余操作系统Rad Hat/CentOS/Oracle Linux/SUSE 12/EulerOS等跳过此步骤）：

📖 说明

如果采用LVM分区机制，需要参考[3.a~3.e](#)修改盘符模式。

- a. 在虚拟机“/boot/grub”目录下执行**blkid**命令，检查磁盘分区是否正常。

```
linux-a5d6:/boot/grub # blkid
/dev/sda1: UUID="c23d47f8-ef1b-4c4e-9a3b-5ae138ef7184" TYPE="swap"
/dev/sda2: UUID="27644978-e244-4a8c-996a-03119fdaff71" TYPE="ext3"
```

如果出现磁盘分区没有UUID等异常情况，请检查安装虚拟机操作系统的步骤是否正确，并重新安装虚拟机。

- b. 使用vi编辑器打开“/boot/grub/menu.lst”文件，修改OS启动参数并保存配置。修改前后对比如下：

修改前的menu.lst：

```
# Modified by YaST2. Last modification on Thu Apr 27 10:18:42 EDT 2017
default 0
timeout 8
##YaST - generic_mbr
gfxmenu (hd0,1)/boot/message
##YaST - activate

###Don't change this comment - YaST2 identifier: Original name: linux###
title SUSE Linux Enterprise Server 11 SP4 - 3.0.101-63
    root (hd0,1)
    kernel /boot/vmlinuz-3.0.101-63-default root=/dev/disk/by-id/ata-QEMU_HARDDISK_QM00001-part2
    resume=/dev/disk/by-id/ata-QEMU_HARDDISK_QM00001-part1 splash=silent showopts vga=0x314
    initrd /boot/initrd-3.0.101-63-default

###Don't change this comment - YaST2 identifier: Original name: failsafe###
title Failsafe -- SUSE Linux Enterprise Server 11 SP4 - 3.0.101-63
    root (hd0,1)
    kernel /boot/vmlinuz-3.0.101-63-default root=/dev/disk/by-id/ata-QEMU_HARDDISK_QM00001-part2
    showopts ide=nodma apm=off noresume add=off powersaved=off nohz=off highres=off processor.max_cst
    ate=1 nomodeset x11failsafe vga=0x314
    initrd /boot/initrd-3.0.101-63-default

###Don't change this comment - YaST2 identifier: Original name: floppy###
title Floppy
    rootnoverify (fd0)
    chainloader +1
```

修改后的menu.lst:

```
# Modified by YaST2. Last modification on Thu Apr 20 15:02:03 EDT 2017
default 0
timeout 8
##YaST - generic_mbr
gfxmenu (hd0,1)/boot/message
##YaST - activate

###Don't change this comment - YaST2 identifier: Original name: linux###
title SUSE Linux Enterprise Server 11 SP4 - 3.0.101-63
    root (hd0,1)
    kernel /boot/vmlinuz-3.0.101-63-default root=/dev/disk/by-uuid/27644978-e244-4a8c-996a-03119fdaff71 resume=/dev/disk/by-uid/c23d47f8-ef1b-4c4e-9a3b-5ae138ef7184 splash=silent crashkernel=256M::128M showopts vga=0x314
    initrd /boot/initrd-3.0.101-63-default

###Don't change this comment - YaST2 identifier: Original name: failsafe###
title Failsafe -- SUSE Linux Enterprise Server 11 SP4 - 3.0.101-63
    root (hd0,1)
    kernel /boot/vmlinuz-3.0.101-63-default root=/dev/disk/by-uuid/27644978-e244-4a8c-996a-03119fdaff71 showopts ide=nodma ap
    m=off noresume add=off powersaved=off nohz=off highres=off processor.max_cstate=1 nomodeset x11failsafe vga=0x314
    initrd /boot/initrd-3.0.101-63-default

###Don't change this comment - YaST2 identifier: Original name: floppy###
title Floppy
    rootnoverify (fd0)
    chainloader +1
```

c. 使用vi编辑器打开 “/boot/grub/device.map” 文件，修改前后对比如下：

修改前的device.map:

```
(hd0) /dev/disk/by-id/ata-QEMU_HARDDISK_QM00001
~
```

修改后的device.map:

```
(hd0) /dev/sda
```

d. 使用vi编辑器打开 “/etc/fstab” 文件，修改前后对比如下：

修改前的fstab:

```
/dev/disk/by-id/ata-QEMU_HARDDISK_QM00001-part1 swap swap defaults 0 0
/dev/disk/by-id/ata-QEMU_HARDDISK_QM00001-part2 / ext3 acl,user_xattr 1 1
proc /proc proc defaults 0 0
sysfs /sys sysfs noauto 0 0
debugfs /sys/kernel/debug debugfs noauto 0 0
usbfs /proc/bus/usb usbfs noauto 0 0
devpts /dev/pts devpts mode=0620,gid=5 0 0
```

修改后的fstab:

```
/dev/disk/by-uuid/c23d47f8-ef1b-4c4e-9a3b-5ae138ef7184 swap swap defaults 0 0
/dev/disk/by-uuid/27644978-e244-4a8c-996a-03119fdaff71 / ext3 acl,user_xattr 1 1
proc /proc proc defaults 0 0
sysfs /sys sysfs noauto 0 0
debugfs /sys/kernel/debug debugfs noauto 0 0
usbfs /proc/bus/usb usbfs noauto 0 0
devpts /dev/pts devpts mode=0620,gid=5 0 0
```


- e. 使用vi编辑器打开“/etc/mtab”，删除/dev/sr0的CDROM所在行并保存配置。

删除后的mtab：

```
/dev/sda2 / ext3 rw,acl,user_xattr 0 0
proc /proc proc rw 0 0
sysfs /sys sysfs rw 0 0
debugfs /sys/kernel/debug debugfs rw 0 0
udev /dev tmpfs rw,mode=0755 0 0
tmpfs /dev/shm tmpfs rw,mode=1777 0 0
devpts /dev/pts devpts rw,mode=0620,gid=5 0 0
fusectl /sys/fs/fuse/connections fusectl rw 0 0
securityfs /sys/kernel/security securityfs rw 0 0
gvfs-fuse-daemon /root/.gvfs fuse.gvfs-fuse-daemon rw,nosuid,nodev 0 0
```

说明

如果没有/dev/sr0配置项，无需执行此步骤。

- f. 使用vi编辑器打开“/etc/sysconfig/bootloader”，修改前后对比如下：

修改前的bootloader：

```
## Path: System/Bootloader
## Description: Bootloader configuration
## Type: list(grub,lilo,none)
## Default: grub
LOADER_TYPE="grub"
DEFAULT_NAME="SUSE Linux Enterprise Server 11 SP4 - 3.0.101-63"
DEFAULT_APPEND="resume=/dev/disk/by-id/ata-QEMU_HARDDISK_QM00001-part1 splash=silent crashkernel=256M::128M showopts"
DEFAULT_VGA="0x314"
FAILSAFE_APPEND="showopts ide=nodma apm=off noresume edd=off powersaved=off nohz=off highres=off processor.max_cstate=1 nomod
eset x11failsafe"
FAILSAFE_VGA="0x314"
XEN_KERNEL_APPEND="resume=/dev/disk/by-id/ata-QEMU_HARDDISK_QM00001-part1 splash=silent showopts"
XEN_APPEND=""
XEN_VGA="0x314"
## Path: System/Bootloader
## Description: Bootloader configuration
## Type: yesno
## Default: no
#
# Should the boot cycle detection be used to
# avoid unconditional reboot cycles of not
# supervised system.
#
CYCLE_DETECTION="no"
```

修改后的bootloader：

```
## Path: System/Bootloader
## Description: Bootloader configuration
## Type: list(grub,lilo,none)
## Default: grub
LOADER_TYPE="grub"
DEFAULT_NAME="SUSE Linux Enterprise Server 11 SP4 - 3.0.101-63"
DEFAULT_APPEND="resume=/dev/disk/by-uuid/c23d47f8-ef1b-4c4e-9a3b-5ae138ef7184 splash=silent crashkernel=256M::128M showopts"
DEFAULT_VGA="0x314"
FAILSAFE_APPEND="showopts ide=nodma apm=off noresume edd=off powersaved=off nohz=off highres=off processor.max_cstate=1 nomod
eset x11failsafe"
FAILSAFE_VGA="0x314"
XEN_KERNEL_APPEND="resume=/dev/disk/by-uuid/c23d47f8-ef1b-4c4e-9a3b-5ae138ef7184 splash=silent showopts"
XEN_APPEND=""
XEN_VGA="0x314"
## Path: System/Bootloader
## Description: Bootloader configuration
## Type: yesno
## Default: no
#
# Should the boot cycle detection be used to
# avoid unconditional reboot cycles of not
# supervised system.
#
CYCLE_DETECTION="no"
```

如果SUSE 11 SP4采用LVM分区机制，则按照以下方法将虚拟机文件中的虚拟磁盘更改为实际物理磁盘：

- a. 在虚拟机“/boot/grub”目录下执行**blkid**命令，检查磁盘分区是否正常。

```
server0f0147a4-36b3-496e-a69d-7629d1785ec4:/home/rhel # blkid
/dev/sda1: UUID="118f08b7-e8b5-450c-961a-341af775b727" TYPE="ext3"
/dev/sda2: UUID="LvPjUd-vZxq-mBHN-CmGj-Hfhx-fI0m-0e1htr" TYPE="LVM2_member"
/dev/sda3: LABEL="config-2" TYPE="iso9660"
/dev/mapper/system-root: UUID="fad905b5-d12f-4996-8906-af70f9bdd884" TYPE="ext3"
/dev/mapper/system-swap: UUID="91cebcd2-4a04-4f5b-b1d4-e9bf43e67346" TYPE="swap"
server0f0147a4-36b3-496e-a69d-7629d1785ec4:/home/rhel #
```

如果出现磁盘分区没有UUID等异常情况，请检查安装虚拟机操作系统的步骤是否正确，并重新安装虚拟机。

- b. 使用vi编辑器打开“/boot/grub/menu.lst”文件，修改OS启动参数并保存配置。修改前后对比如下：

修改前的menu.lst：

```
# Modified by YaST2. Last modification on Wed Nov 1 19:32:04 CST 2017
default 0
timeout 8
##YaST - generic_mbr
gfxmenu (hd0,0)/message
##YaST - activate

serial --unit=0 --speed=115200
terminal --timeout=5 serial console

##Don't change this comment - YaST2 identifier: Original name: linux##
title SUSE Linux Enterprise Server 11 SP4 - 3.0.101-63
    root (hd0,0)
    kernel /vmlinuz-3.0.101-63-default root=/dev/system/root resume=/dev/system/swap splash=silent crashkernel=256M::128M showopts vga=0x314 cons
ole=tty0 console=ttyS0,115200n8
    initrd /initrd-3.0.101-63-default

##Don't change this comment - YaST2 identifier: Original name: failsafe##
title Failsafe -- SUSE Linux Enterprise Server 11 SP4 - 3.0.101-63
    root (hd0,0)
    kernel /vmlinuz-3.0.101-63-default root=/dev/system/root showopts ide=nodma apm=off noresume add-on-off powersaved=off nohz=off highres=off proc
essor.max_cstate=1 nomodeset x11failsafe vga=0x314 console=tty0 console=ttyS0,115200n8
    initrd /initrd-3.0.101-63-default
```

修改后的menu.lst：

```
# Modified by YaST2. Last modification on Wed Nov 1 19:32:04 CST 2017
default 0
timeout 8
##YaST - generic_mbr
gfxmenu (hd0,0)/message
##YaST - activate

serial --unit=0 --speed=115200
terminal --timeout=5 serial console

##Don't change this comment - YaST2 identifier: Original name: linux##
title SUSE Linux Enterprise Server 11 SP4 - 3.0.101-63
    root (hd0,0)
    kernel /vmlinuz-3.0.101-63-default root=/dev/disk/by-uuid/fad905b5-d12f-4996-8906-af70f9bdd884 resume=/dev/disk/by-uuid/91cebcd2-4a04-4f5b-b1
d4-e9bf43e67346 splash=silent crashkernel=256M::128M showopts vga=0x314 console=tty0 console=ttyS0,115200n8
    initrd /initrd-3.0.101-63-default

##Don't change this comment - YaST2 identifier: Original name: failsafe##
title Failsafe -- SUSE Linux Enterprise Server 11 SP4 - 3.0.101-63
    root (hd0,0)
    kernel /vmlinuz-3.0.101-63-default root=/dev/disk/by-uuid/fad905b5-d12f-4996-8906-af70f9bdd884 showopts ide=nodma apm=off noresume add-on-off po
wersaved=off nohz=off highres=off processor.max_cstate=1 nomodeset x11failsafe vga=0x314 console=tty0 console=ttyS0,115200n8
    initrd /initrd-3.0.101-63-default
```

- c. 使用vi编辑器打开“/boot/grub/device.map”文件，修改前后对比如下：

修改前的device.map：

```
(hd0) /dev/disk/by-id/ata-QEMU_HARDDISK_QM00001
```

修改后的device.map：

```
(hd0) /dev/sda
~
```

- d. 使用vi编辑器打开“/etc/fstab”文件，修改前后对比如下：

修改前的fstab：

```
/dev/system/swap swap swap defaults 0 0
/dev/system/root / ext3 acl,user_xattr 1 1
/dev/disk/by-id/ata-QEMU_HARDDISK_QM00001-part1 /boot ext3 acl,user_xattr 1 2
proc /proc proc defaults 0 0
sysfs /sys sysfs noauto 0 0
debugfs /sys/kernel/debug debugfs noauto 0 0
usbfs /proc/bus/usb usbfs noauto 0 0
devpts /dev/pts devpts mode=0620,gid=5 0 0
```

修改后的fstab：

```
/dev/disk/by-uuid/91cebcd2-4a04-4f5b-b1d4-e9bf43e67346 swap swap defaults 0 0
/dev/disk/by-uuid/fad905b5-d12f-4996-8906-af70f9bdd884 / ext3 acl,user_xattr 1 1
/dev/disk/by-uuid/118f08b7-e8b5-450c-961a-341af775b727 /boot ext3 acl,user_xattr 1 2
proc /proc proc defaults 0 0
sysfs /sys sysfs noauto 0 0
debugfs /sys/kernel/debug debugfs noauto 0 0
usbfs /proc/bus/usb usbfs noauto 0 0
devpts /dev/pts devpts mode=0620,gid=5 0 0
```

- e. 使用vi编辑器打开“/etc/sysconfig/bootloader”，修改前后对比如下：

修改前的bootloader：

```
## Path: System/Bootloader
## Description: Bootloader configuration
## Type: list(grub,lilo,none)
## Default: grub
LOADER_TYPE="grub"
DEFAULT_NAME="SUSE Linux Enterprise Server 11 SP4 - 3.0.101-63"
DEFAULT_APPEND="resume=/dev/system/swap splash=silent crashkernel=256M::128M showopts"
DEFAULT_VGA="0x314"
FAILSAFE_APPEND="showopts ide=nodma apm=off noresume edd=off powersaved=off nohz=off highres=off processor.max_cstate=1 nomodeset x11failsafe"
FAILSAFE_VGA="0x314"
XEN_KERNEL_APPEND="resume=/dev/system/swap splash=silent showopts"
XEN_APPEND=""
XEN_VGA="0x314"
```

修改后的bootloader:

```
## Path: System/Bootloader
## Description: Bootloader configuration
## Type: list(grub,lilo,none)
## Default: grub
LOADER_TYPE="grub"
DEFAULT_NAME="SUSE Linux Enterprise Server 11 SP4 - 3.0.101-63"
DEFAULT_APPEND="resume=/dev/disk/by-uuid/91cebcd2-4a04-4f5b-b1d4-e9bf43e67346 splash=silent crashkernel=256M::128M showopts"
DEFAULT_VGA="0x314"
FAILSAFE_APPEND="showopts ide=nodma apm=off noresume edd=off powersaved=off nohz=off highres=off processor.max_cstate=1 nomodeset x11failsafe"
FAILSAFE_VGA="0x314"
XEN_KERNEL_APPEND="resume=/dev/disk/by-uuid/91cebcd2-4a04-4f5b-b1d4-e9bf43e67346 splash=silent showopts"
XEN_APPEND=""
XEN_VGA="0x314"
```

配置完成后执行**mkinitrd**命令，如果回显信息中**resume**没有被更新为“**by-uuid**”模式，需要执行**reboot**后再次执行**mkinitrd**命令，确保**resume**被更新为“**by-uuid**”模式。

4. 对于Ubuntu 18.04、Ubuntu 16.04 for ARM，还需要修改grub引导文件、fstab文件和interfaces文件。

- a. 修改“/etc/default/grub”配置文件中的参数。

将GRUB_DISABLE_LINUX_UUID参数设置为true。

```
...
# Uncomment if you don't want GRUB to pass "root=UUID=xxx" parameter to linux
GRUB_DISABLE_LINUX_UUID=true

# Uncomment to disable generation of recovery mode menu entries
...
```

修改后执行命令：**sudo update-grub2**

- b. 修改“/etc/fstab”中的UUID为物理磁盘/dev/sdax。硬盘的UUID可以通过**sudo blkid**命令获取。

```
# <file system> <mount point> <type> <options> <dump> <pass>
# / was on /dev/sda2 during installation
/dev/sda2 / ext4 errors=remount-ro 0 1
# /boot/efi was on /dev/sda1 during installation
/dev/sda1 /boot/efi vfat umask=0077 0 1
# swap was on /dev/sda3 during installation
/dev/sda3 none swap sw 0 0
```

- c. 修改“/etc/network/interfaces”文件，删除其中除了“lo interface”之外的其他interface信息。

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces (5).
```

```
source /etc/network/interfaces.d/*
```

```
# The loopback network interface
auto lo
iface lo inet loopback
```

```
~
~
```


3.11 安装 bms-network-config 软件包

操作背景

安装bms-network-config软件包，与Cloud-Init配合完成集中式裸金属服务器的网络配置，需要安装bms-network-config软件包的裸金属服务器规格如表3-6所示。分布式裸金属服务器（带SDI 3.0卡或SDI 2.2网络卡的机型）不需要执行此步骤。

表 3-6 裸金属服务器规格

| 规格类型 | 规格名称 |
|--------|--|
| 通用型 | physical.s3.large、physical.s3.xlarge、physical.s3.2xlarge、physical.s4.medium、physical.s4.large、physical.s4.xlarge、physical.s4.2xlarge、physical.s4.3xlarge |
| 本地存储型 | physical.d1.large、physical.d2.tiny、physical.d2.large、physical.d2.xmedium |
| 内存优化型 | physical.m2.small、physical.m2.medium、physical.m2.large、physical.m2.xlarge |
| IO优化型 | physical.io1.large、physical.io2.xlarge |
| GPU加速型 | physical.p1.large、physical.p2.large、physical.g1.small、physical.p3.large、physical.pi6.3xlarge.6 |
| 高性能计算型 | physical.h2.large、physical.hc2.xlarge |
| 鲲鹏型 | physical.ks1.2xlarge |

前提条件

- 已登录虚拟机。
- 已安装好cloud-init软件。
- 已参考[软件准备](#)章节下载bms-network-config软件包及SHA256校验码并完成完整性校验，详情请参考[如何对软件进行完整性校验？](#)。

 说明

请下载对应操作系统的rpm包，需要注意：Ubuntu/Debian所用的是deb包，ARM系列操作系统（当前仅支持CentOS和EulerOS）使用aarch的rpm包。

操作步骤

1. 进入bms-network-config安装包所在目录，执行**rpm -ivh bms-network-config-1.0-7.centosRedhat7.x86_64.rpm**命令。
- ```
[root@localhost r74]# rpm -ivhbms-network-config-1.0-7.centosRedhat7.x86_64.rpm
Preparing... ##### [100%]
Updating / installing...
 1:bms-network-config-1.0.7.centosRe##### [100%]
```

## 说明

SUSE 12/SUSE 15系列安装bms-network-config时, 如果出现下面的依赖报错, 请执行:  
**rpm -ivh bms-network-config-1.0-9.suse12.x86\_64.rpm --nodeps --force**

```
linux-dyqy:/home/bms-network-config_1.0.9 # rpm -ivh bms-network-config-1.0-9.suse12.x86_64.rpm
error: Failed dependencies:
 cloud-init is needed by bms-network-config-1.0-9.suse12.x86_64
linux-dyqy:/home/bms-network-config_1.0.9 #
```

对于Ubuntu/Debian操作系统, 使用**dpkg -i xxx**进行安装, xxx为deb包名。

```
root@ubuntu:~/file# dpkg -i bms-network-config-1.0.7.ubuntu1604-918.deb
Selecting previously unselected package bms-network-config.
(Reading database ... 97630 files and directories currently installed.)
Preparing to unpack bms-network-config-1.0.7.ubuntu1604-918.deb ...
Unpacking bms-network-config (1.0) ...
Setting up bms-network-config (1.0) ...
root@ubuntu:~/file# dpkg -s bms-network-config
```

## 说明

rpm包和deb包的名称以实际为准。

2. 安装完成后, 执行**rpm -qa | grep bms-network-config**, 如下回显表示安装成功。

```
[root@localhost r74]# rpm -qa | grep bms
bms-network-config-1.0.7.centosRedhat7.x86_64
```

对于Ubuntu/Debian操作系统, 使用**dpkg -s bms-network-config**命令查看。

3. 查看服务状态。

- 对于Oracle Linux 7、Red Hat 7、CentOS 7、Ubuntu 16.04、Ubuntu 18.04、SUSE 12、SUSE 15、EulerOS操作系统, 执行**service bms-network-config status**命令查看服务状态。如果不是“enabled”, 请执行**systemctl enable bms-network-config**, 开启服务。

```
[root@localhost r74]# service bms-network-config status
Redirecting to /bin/systemctl status bms-network-config.service
bms-network-config.service - Network Config
Loaded: loaded (/usr/lib/systemd/system/bms-network-config.service; enabled vendor preset: disabled)
Active: inactive (dead)
```

- 对于Red Hat 6系列、CentOS 6系列、SUSE 11 SP4和Oracle Linux 6.8、Oracle Linux 6.9操作系统, 执行**chkconfig --list | grep bms-network-config**命令查看服务状态。如果不是“on”, 请执行**chkconfig bms-network-config on**命令, 开启服务。

```
[root@localhost r69]# chkconfig --list | grep bms
bms-network-config 0:off 1:off 2:on 3:on 4:off 5:on 6:off
```

- 对于Ubuntu 14.04/Debian, 可以使用**initctl status bms-network\_config**查看安装状态。

```
root@ubuntu:~# initctl status bms-network_config
bms-network_config stop/waiting
```

4. 检查服务启动依赖。

参考使用**systemctl cat bms-network-config**命令, 确保服务文件如下:

```
[Unit]
Description=NetworkConfig
DefaultDependencies=no
After=dbus.service
Wants=dbus.service
[Service]
Type=oneshot
ExecStart=/usr/bin/bms-network_config rhel
RemainAfterExit=yes
TimeoutSec=0
```

```
[Install]
WantedBy=multi-user.target
```

尤其注意依赖项顺序。如果不对，使用**`vim /usr/lib/systemd/system/bms-network-config.service`**命令进行修改。

## 3.12 安装 network 服务

### 操作背景

对于centos8系列/euler2.9系列/redhat8系列/ubuntu20及以上操作系统，默认不安装network服务，对于集中式裸金属服务器，需要使用该服务配合网络脚本完成服务器系统的网络配置。分布式裸金属服务器（带SDI 3.0卡或SDI 2.2网络卡的机型）不需要执行此步骤。

### 操作步骤

以下步骤以euler2.10系统为例。

1. 进入系统，查询network服务状态：  

```
[root@euler210-arm-new ~]# systemctl status network
```

  
Unit network.service could not be found.
2. 使用yum源安装 network-scripts：  

```
[root@euler210-arm-new network-scripts]# yum install network-scripts
```
3. 安装完毕后查询，network服务已存在：  

```
[root@euler210-arm-new network-scripts]# systemctl status network
```

  
• network.service - LSB: Bring up/down networking  
Loaded: loaded (/etc/rc.d/init.d/network; generated)  
Active: inactive (dead)  
Docs: man:systemd-sysv-generator(8)

## 3.13 安装 SDI 卡驱动

### 操作背景

当前SDI卡分为三种，SDI存储卡、SDI 2.2网络卡、SDI 3.0卡（存储和网络归一）。若服务器含有SDI存储卡（即不是SDI 2.2网络卡，也不是SDI 3.0卡），则镜像中需要安装SDI卡前端驱动插件，使裸金属服务器能够支持挂载云硬盘。如果不属于上述情况，则可以跳过此步骤。目前ARM 64系统不支持SDI存储卡，无需安装。

### 前提条件

- 已登录虚拟机。
- 已参考[软件准备](#)章节下载SDI卡的驱动程序（scsi\_ep\_front），并且已经上传到虚拟机。

#### 说明

scsi\_ep\_front驱动包的版本必须为1.0.13及以上。

## 操作步骤

以下步骤仅供参考，不同版本镜像安装SDI卡驱动的回显信息不同，视实际情况而定。

1. 进入SDI驱动安装包所在目录，执行以下命令：

```
rpm -ivh kmod-scsi_ep_front-
centos_7.6_1.0.18-3.10.0_957.el7.centos.x86_64.rpm
```

### 说明

Ubuntu/Debian系列操作系统的SDI驱动使用的是deb安装包，使用命令**dpkg -i xxx**（xxx为SDI安装包名）进行安装。

2. 安装完后执行：**rpm -qa | grep scsi**，如下回显表示安装成功。

```
[root@localhost ~] rpm -qa | grep scsi
lsscsi-0.27-6.el7.x86_64
kmod-scsi_ep_front-centos_7.6_1.0.18-3.10.0_957.el7.centos.x86_64
```

如果是Ubuntu 18.04、Ubuntu 16.04、Ubuntu 14.04和Debian，执行**dkms status**查看安装结果。若驱动显示“installed”，表示安装成功，然后执行**update-initramfs -u**命令将驱动打入内核。

```
...
root@ubuntu:~/file# dkms status
scsi_ep_front, 1.0.13, 4.4.0-21-generic, x86_64: installed
scsi_ep_front, 1.0.13, 4.4.0-59-generic, x86_64: built
scsi_ep_front, 1.0.13, 4.4.0-96-generic, x86_64: installed
```

3. 执行下列命令，查看SDI驱动是否为内核对应的版本：

- a. **uname -r**，获取当前系统内核版本，例如：

```
[root@localhost r74]# uname -r
3.10.0-957.el7.x86_64
```

- b. **find / -name "\*front\*.ko"**，查看内核版本对应的SDI驱动，如下回显可以看到对应内核的SDI卡驱动。

```
[root@localhost r74]# find / -name "*front*.ko"
/usr/lib/modules/3.10.0-957.el7.x86_64/extra/scsi_ep_front/scsi_ep_front.ko
```

## 3.14 安装 Hi1822 网卡驱动

在服务器需要配套Hi1822网卡时，需要安装Hi1822网卡驱动。区分三种场景，详见以下章节指导。

### 3.14.1 标卡驱动（集中式裸金属服务器）

#### 操作背景

当集中式裸金属服务器配套有Hi1822网卡时，需要参考本节指导安装Hi1822网卡驱动，以CentOS 7.6为例进行说明。

#### 前提条件

- 已登录虚拟机。
- 已获取Hi1822驱动安装包及管理软件并上传至虚拟机。

获取方式：参考[软件准备](#)章节下载驱动的rpm包。

注意：所下载的Hi1822驱动安装包及管理软件要和操作系统相对应。

## 操作步骤

1. 查看是否已有hinic网卡驱动。  
**rpm -qa | grep hinic**
  - 如果有，则执行2。
  - 如果没有，执行3。
2. 卸载hinic网卡驱动。  
**rpm -e kmod-hinic**  
**rmmod hinic**
3. 进入rpm安装包所在目录，执行以下命令安装驱动：  
**rpm -ivh NIC-Hi1822-CentOS7.6-hinic-3.9.0.8-1-x86\_64.rpm**
4. 执行**modprobe hinic**加载最新的Hi1822网卡驱动。
5. 安装Hi1822管理软件（hinicadm包）：  
**rpm -ivh NIC-Hi1822-CentOS7.6-hinicadm-3.9.0.8-1-x86\_64.rpm**

### 3.14.2 卸载卡驱动（SDI 2.2 分布式裸金属服务器）

#### 操作背景

当裸金属服务器配套有SDI 2.2硬件时，需要参考本节指导安装Hi1822网卡驱动，以EulerOS 2.3为例进行说明。

#### 前提条件

- 已登录虚拟机。
- 已获取Hi1822驱动安装包及管理软件并上传至虚拟机。  
获取方式如下：
  - a. 参考[软件准备](#)章节下载驱动的zip包。
  - b. 此处以“Hi1822\_BM\_X86\_1.19.3.B036.tar.gz”为例，介绍如何获取最终需要的安装包。  
下载“Hi1822\_BM\_X86\_1.19.3.B036.tar.gz”驱动包并解压，根据操作系统类型和内核版本获取对应的驱动rpm包：kmod-hinic-1.8.3.16\_3.10.0\_514.41.4.28.h70.x86\_64-1.x86\_64.rpm。  
**kmod-hinic-1.8.3.16\_3.10.0\_514.41.4.28.h70.x86\_64-1.x86\_64.rpm**

## 操作步骤

1. 查看是否已有hinic网卡驱动。  
**rpm -qa | grep hinic**
  - 如果有，则执行2。
  - 如果没有，则执行3。
2. 卸载hinic网卡驱动。  
**rpm -e kmod-hinic**  
**rmmod hinic**
3. 进入rpm安装包所在目录，执行以下命令安装驱动：

```
rpm -ivh kmod-hinic-1.8.3.16_3.10.0_514.41.4.28.h70.x86_64-1.x86_64.rpm
```

4. 执行**modprobe hinic**加载最新的Hi1822网卡驱动。

### 3.14.3 卸载卡驱动（SDI 3.0 分布式裸金属服务器）

#### 操作背景

当裸金属服务器配套有SDI 3.0硬件时，需要参考本节指导安装Hi1822网卡驱动，以CentOS 7.6为例进行说明。

#### 前提条件

- 已登录虚拟机。
- 已获取Hi1822驱动安装包及管理软件并上传至虚拟机。  
获取方式如下：
  - a. 参考[软件准备](#)章节下载驱动的zip包。
  - b. 此处以“uNIC\_GuestOS\_Driver\_BM\_2.21.8.B070.tar.gz”为例，介绍如何获取最终需要的安装包。  
下载“uNIC\_GuestOS\_Driver\_BM\_2.21.8.B070.tar.gz”驱动包并解压，根据操作系统类型和内核版本获取对应的驱动rpm包：kmod-hinic-5.0.0.7\_3.10.0\_957-1.el7.x86\_64.rpm。

#### 操作步骤

1. 查看是否已有hinic网卡驱动。  
**rpm -qa | grep hinic**
  - 如果有，则执行2。
  - 如果没有，则执行3。
2. 卸载hinic网卡驱动。  
**rpm -e kmod-hinic**  
**rmmod hinic**
3. 进入rpm安装包所在目录，执行以下命令安装驱动：  
**rpm -ivh kmod-hinic-5.0.0.7\_3.10.0\_957-1.el7.x86\_64.rpm**
4. 执行**modprobe hinic**加载最新的Hi1822网卡驱动。

## 3.15 安装 IB 驱动（可选）

#### 操作背景

此项为可选项，如需安装IB驱动，以CentOS 7.4为例进行说明。

#### 说明

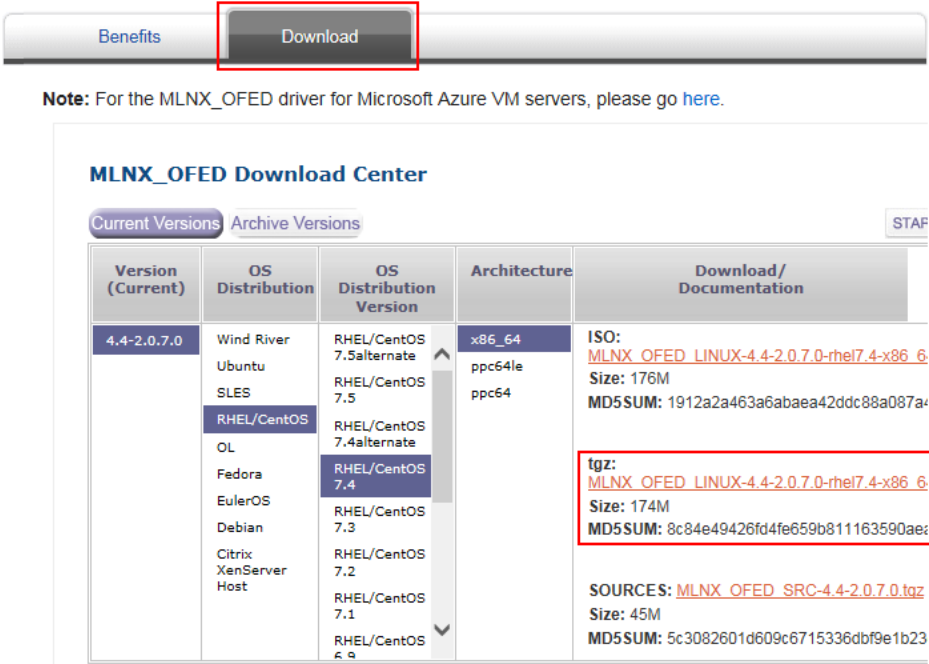
1. 支持的OS列表：CentOS 7.3，CentOS 7.4，CentOS 7.6 ARM，RedHat 7.3，RedHat 7.4，SUSE 12 SP3，Oracle Linux 7.4。
2. IB驱动安装版本：官网有4.2版本驱动时，建议安装4.2版本；如果没有4.2版本驱动，建议安装4.3或更新版本。

前提条件

- 已登录虚拟机。
- IB驱动安装包（下载对应操作系统的）已上传到虚拟机。

操作步骤

1. 参考[软件准备](#)章节下载OFED。  
根据虚拟机的操作系统选择相应的tgz安装包进行下载，并校验文件完整性。  
以CentOS 7.4为例，选择驱动版本信息如下图：



2. 将下载完成的OFED安装包，上传到虚拟机中。具体操作请参见“[虚拟机环境配置](#)”章节中的步骤。
3. 执行命令`tar -zxvf xxx`（xxx为OFED安装包名）解压OFED安装包。
4. 解压完成后，进入“MLNX\_OFED\_LINUX-4.2-1.2.0.0-RHEL7.4-X86\_64”文件夹执行安装。

./mlnxofedinstall

```
[root@localhost MLNX_OFED_LINUX-4.2-1.2.0.0-rhel7.4-x86_64]# ./mlnxofedinstall
Logs dir: /tmp/MLNX_OFED_LINUX.1479.logs
General log file: /tmp/MLNX_OFED_LINUX.1479.logs/general.log
Verifying KMP rpms compatibility with target kernel...
Error: One or more required packages for installing MLNX_OFED_LINUX are missing.
Please install the missing packages using your Linux distribution Package Management tool.
Run:
yum install tcl tk
```

5. 在安装过程中，如果缺少相关依赖包，执行`yum install xxx`进行安装。  
按照上一步提示执行：`yum install tcl tk`
6. 按照步骤5安装依赖完成后，再次执行`./mlnxofedinstall`安装脚本。  
等待一段时间后如下所示，即为安装成功。

Installation finished successfully.

```
Preparing... ##### [100%]
Updating / installing...
```



```
1:mlnx-fw-updater-4.2-1.2.0.0 ##### [100%]
Added 'RUN_FW_UPDATER_ONBOOT=no' to /etc/infiniband/openib.conf
...
```

7. 安装成功后，执行**`/etc/init.d/openibd restart`**加载驱动。

```
[root@localhost MLNX_OFED_LINUX-4.2-1.2.0.0-rhel7.4-x86_64]# /etc/init.d/openibd restart
Uploading HCA driver: [OK]
Loading HCA driver and Access Layer: [OK]
```

## 3.16 安装 FusionServer/TaiShanServer 服务器 iDriver 驱动

### 3.16.1 准备工作

1. 参考[软件准备](#)章节下载驱动的zip包。

例如：

- CentOS 7.6: FusionServer iDriver-CentOS7.6-Driver-V116.zip
- RHEL 7.3: FusionServer iDriver-RHEL7.3-Driver-V116.zip
- Ubuntu 16.04: FusionServer iDriver-Ubuntu16.04-Driver-V116.zip
- EulerOS 2.8 ARM: TaiShanServer iDriver-EulerOS2.8-Driver-V103.zip
- CentOS 7.6 ARM: TaiShanServer iDriver-CentOS7.6-Driver-V112.zip

2. 此处以CentOS 7.6的v5服务器驱动以及EulerOS 2.8 ARM和CentOS 7.6 ARM的TaiShan服务器驱动为例，介绍如何获取最终需要的安装包。

- CentOS 7.6

下载“FusionServer iDriver-CentOS7.6-Driver-V116.zip”驱动包并解压，获取“onboard\_driver\_CentOS7.6.iso”文件。继续解压

“onboard\_driver\_CentOS7.6.iso”，获取“NIC-

X710\_X722\_XL710\_XXV710-CentOS7.6-i40e-2.15.9-1-x86\_64.rpm”、

“RAID-3008IR\_3008IT\_3408IT\_3416IT-CentOS7.6-mpt3sas-27.00.00.00-1-x86\_64.rpm”和“RAID-3004iMR\_3108\_3408iMR\_3416iMR\_3508\_3516-CentOS7.6-megaraid\_sas-07.716.01.00-1-x86\_64.rpm”文件。

“NIC-X710\_X722\_XL710\_XXV710-CentOS7.6-i40e-2.15.9-1-

x86\_64.rpm”、“RAID-3008IR\_3008IT\_3408IT\_3416IT-CentOS7.6-mpt3sas-27.00.00.00-1-x86\_64.rpm”和

“RAID-3004iMR\_3108\_3408iMR\_3416iMR\_3508\_3516-CentOS7.6-megaraid\_sas-07.716.01.00-1-x86\_64.rpm”文件将分别用于v5服务器板载网卡驱动、mpt3驱动和megaraid\_sas驱动的安装。

- EulerOS 2.8 ARM

下载“TaiShanServer iDriver-EulerOS2.8-Driver-V103.zip”驱动包并解压，获取“onboard\_driver\_EulerOS2.8.iso”。继续解压

“onboard\_driver\_EulerOS2.8.iso”，获取“NIC-IN200-EulerOS2.8-

hinic-2.4.1.0-aarch64.rpm”，可以用于Hi1822标卡驱动的安装，但由于[安装Hi1822网卡驱动](#)章节已经安装Hi1822驱动，故可以跳过此步骤。

- CentOS 7.6 ARM

下载“TaiShanServer iDriver-CentOS7.6-Driver-V112.zip”驱动包并解压，获取“onboard\_driver\_CentOS7.6.iso”。继续解压

“onboard\_driver\_CentOS7.6.iso”，获取

“RAID-3108\_3408iMR\_3416iMR\_3508\_3516-CentOS7.6-

megaraid\_sas-07.716.01.00-aarch64.rpm”文件和“NIC-Hi1822-CentOS7.6-hinic-3.9.0.8-aarch64.rpm”文件，用于megaraid\_sas驱动和Hi1822标卡驱动的安装。其中[安装Hi1822网卡驱动](#)章节已经安装Hi1822驱动，故不需要再次安装Hi1822（hinic）驱动。

## 3.16.2 安装服务器板载网卡驱动

### 操作背景

当服务器配套有X722板载网卡时，需要参考此章节安装服务器板载网卡驱动。

#### 说明

本文以CentOS 7.6为例介绍如何安装服务器板载网卡驱动，其他操作系统的安装方法类似。

### 操作步骤

1. 将[准备工作](#)中获取的“NIC-X710\_X722\_XL710\_XXV710-CentOS7.6-i40e-2.15.9-1-x86\_64.rpm”上传至虚拟机。
2. 进入rpm安装包所在目录，执行以下命令安装驱动：

```
rpm -ivh NIC-X710_X722_XL710_XXV710-CentOS7.6-i40e-2.15.9-1-x86_64.rpm
[root@localhost i40e]# rpm -ivh NIC-X710_X722_XL710_XXV710-CentOS7.6-i40e-2.15.9-1-x86_64.rpm
Preparing... ##### [100%]
Updating / installing...
 1:i40e-2.15.9-1 ##### [100%]
original pci.ids saved in /usr/local/share/i40e
Updating initrd...
Using dracut to update initrd...
Successfully updated initrd.
[root@localhost i40e]#
```

3. 安装完成后，执行rpm -qa | grep i40e，如下回显表示安装成功。

```
i40e-2.15.9-1.x86_64
```

## 3.16.3 安装 mpt3 驱动

### 操作背景

如果服务器配套的是3108或3008 RAID卡，需要在镜像中安装mpt3sas驱动。

### 操作步骤

1. 将[准备工作](#)中获取的“RAID-3008IR\_3008IT\_3408IT\_3416IT-CentOS7.6-mpt3sas-27.00.00.00-1-x86\_64.rpm”上传至虚拟机。
2. 安装mpt3驱动。

- a. 进入rpm安装包所在目录，执行以下命令安装驱动：

```
rpm -ivh RAID-3008IR_3008IT_3408IT_3416IT-CentOS7.6-mpt3sas-27.00.00.00-1-x86_64.rpm
[root@localhost mpt3-82599]# rpm -ivh RAID-3008IR_3008IT_3408IT_3416IT-CentOS7.6-mpt3sas-27.00.00.00-1-x86_64.rpm
Preparing... ##### [100%]
Updating / installing...
 1:kmod-mpt3sas-27.00.00.00_el7.6-1 #####
```

```
[100%]
[root@localhost mpt3-82599]#
```

- b. 执行**dracut -f**命令刷新内核。
3. 安装完成后, 执行**rpm -qa | grep mpt3**, 如下回显表示安装成功。  
kmod-mpt3sas-27.00.00.00\_el7.6-1.x86\_64

## 3.16.4 安装 megaraid\_sas 驱动

### 操作背景

如果服务器配套的是3408或3508 RAID卡, 需要在镜像中安装megaraid\_sas驱动。

### 操作步骤

1. 将[准备工作](#)中获取的“RAID-3004iMR\_3108\_3408iMR\_3416iMR\_3508\_3516-CentOS7.6-megaraid\_sas-07.716.01.00-1-x86\_64.rpm”上传至虚拟机。
2. 安装megaraid\_sas驱动。

- a. 进入rpm安装包所在目录, 执行以下安装命令:

```
rpm -ivh RAID-3004iMR_3108_3408iMR_3416iMR_3508_3516-
CentOS7.6-megaraid_sas-07.716.01.00-1-x86_64.rpm
[root@localhost 3408]# rpm -ivh RAID-3004iMR_3108_3408iMR_3416iMR_3508_3516-
CentOS7.6-megaraid_sas-07.716.01.00-1-x86_64.rpm
Preparing... ##### [100%]
Updating / installing..
1:kmod-megaraid_sas-07.716.01.00_el#####
[100%]
[root@localhost 3408]#
```

- b. 执行**dracut -f**命令刷新内核。
3. 安装完成后, 执行**rpm -qa | grep raid**命令, 查看是否安装成功。  
kmod-megaraid\_sas-07.716.01.00-1-x86\_64

## 3.17 安装多路径软件（可选）

### 操作背景

在服务器需要配套HBA卡挂载FC存储卷的场景, 需要参考本节安装多路径软件。

### 操作步骤

1. 请参考《OceanStor UltraPath for Linux xxx 用户指南》进行多路径软件安装。

#### 📖 说明

- 参考[软件准备](#)章节, 获取相应版本软件及文档。
2. 安装完成后, 执行**reboot**命令重启虚拟机使UltraPath生效。
3. 查看多路径软件版本是否正确。

```
upadmin show version
```

## 3.18 安装一键式重置密码插件

为了保证使用镜像创建的裸金属服务器可以实现一键式密码重置功能，建议您在制作镜像时安装重置密码插件“CloudResetPwdAgent”。

### 📖 说明

如果您需要卸载该插件，可参考[卸载方法](#)进行卸载。

一键式重置密码插件暂不支持鲲鹏服务器。

### 前提条件

需保证虚拟机根目录可写入，且剩余空间大于600MB。

### 操作步骤

1. 请参考[获取并校验一键式重置密码插件完整性](#)，下载对应的一键式重置密码插件CloudResetPwdAgent.zip并完成完整性校验。

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。

2. 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**

3. 安装一键式重置密码插件。

- a. 执行以下命令，进入文件CloudResetPwdAgent.Linux。

**cd CloudResetPwdAgent/CloudResetPwdAgent.Linux**

- b. 执行以下命令，添加文件setup.sh的运行权限。

**chmod +x setup.sh**

- c. 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

- 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。
- 一键重置密码插件界面不可用问题相关命令如下：
  1. 在OS内执行curl http://169.254.169.254/openstack/latest/resetpwd\_flag，如果返回值为True，代表是支持一键重置密码的；如果为False或None，代表不支持一键重置密码脚本；
  2. vim /var/log/message如果是centos系列，则为message日志；如果是Ubuntu系列，则为syslog；在启动时间附近搜索bound to看到获取ip的时间
  3. vim /var/log/message在启动时间附近搜索第一个ResetPwd看到cloudResetPwdAgent启动时间
  4. 执行systemd-analyze plot > start.svg,将start.svg导出，用ie打开进行查看
  5. 参照如下配置进行操作  
先使用systemctl cat cloudResetPwdAgent获取配置及配置文件位置  
再参照cloud-init的配置如下修改，尤其是After那一句  
[Unit]  
...  
After=network-online.target
  6. 重启裸机

## 卸载方法

1. 登录裸金属服务器。
2. 执行以下命令，进入bin目录，删除服务cloudResetPwdAgent。  
**cd /CloudrResetPwdAgent/bin**  
**sudo ./cloudResetPwdAgent.script remove**
3. 执行以下命令，删除插件。  
**sudo rm -rf /CloudrResetPwdAgent**  
请检查CloudResetPwdUpdateAgent是否存在，如果存在，执行以下命令删除：  
**sudo rm -rf /CloudResetPwdUpdateAgent**

## 3.19 安全性配置

### 3.19.1 修改 SSH 服务配置项

#### 操作场景

用户可以根据需要选择登录裸金属服务器的登录方式或账户类型，如果需要特殊配置，可执行本节操作修改SSH服务配置项。

#### 操作步骤

1. 如果要禁用密码远程登录，仅支持证书登录的方式，以提高裸金属服务器的安全性，可设置如下参数：
  - 查看文件“/etc/ssh/sshd\_config”中是否存在参数“ChallengeResponseAuthentication”且值是否为“no”，若不是则修改或

添加该参数且值为“no”。使其在使用Xshell登录时，拒绝通过keyboard inactive方式输入密码登录。

2. 如果开放root密码远程登录并开启root用户的ssh权限，需要执行以下操作：

#### 注意

允许root用户登录有一定的安全隐患，请谨慎操作。

- a. 执行以下命令，在vi编辑器中打开“/etc/ssh/sshd\_config”。

**vi /etc/ssh/sshd\_config**

将“sshd\_config”中的“PasswordAuthentication”的值修改为“yes”，“UseDNS”的值修改为“no”。

#### 说明

- 如果是SUSE和OpenSUSE系列操作系统，需要将“sshd\_config”中的“PasswordAuthentication”和“ChallengeResponseAuthentication”参数同时配置为“yes”。
  - 如果是Ubuntu系列操作系统，需要将“PermitRootLogin”参数配置为“yes”。
- b. 修改shadow文件配置，将镜像模板中的初始root账户密码锁定，避免安全风险。
    - i. 使用vi编辑器打开“/etc/shadow”配置文件。

**vi /etc/shadow**

在root账户的密码hash值中添加“!!”。修改后的配置文件如下：

```
cat /etc/shadow | grep root
root:!!6SphQRPXu$Nvg6izXbhDPrcY3j1vRiHaQFVRpNiV3HD/
bjDgnZrACOWPXWJahx78iaut1liglUrwavVGSYQ1JOlw.rDIvH7.:17376:0:99999:7:::
```

- ii. 修改完成后，按“Esc”，输入:wq保存并退出文件编辑。

#### 说明

如果是Ubuntu系列操作系统，需要将安装操作系统过程中新创建的用户删除。例如创建的用户为“ubuntu”，删除命令：**userdel -rf ubuntu**。

## 3.19.2 修改网络脚本权限（可选）

### 操作场景

集中式裸金属镜像，安装完bms-network-config后，需要修改网络脚本权限，满足安全要求。

### 操作步骤

1. 执行**chmod 700 -R /opt/huawei/**命令修改权限。
2. 使用**ls -l /opt/huawei/**和**ls -l /opt/huawei/\***检查权限。

```
[root@bms-zc4 ~]# ls -l /opt/huawei/
total 4
drwx----- 2 root root 4096 Aug 22 19:43 network_config
[root@bms-zc4 ~]#
```

```
[root@bms-zc4 ~]# ls -l /opt/huawei/*
total 164
-rwx----- 1 root root 280 May 9 2018 bms-network-config.conf
-rwx----- 1 root root 55915 May 9 2018 bms-network_config.py
-rwx----- 1 root root 51226 May 9 2018 bms-network_config.pyc
-rwx----- 1 root root 51099 May 9 2018 bms-network_config.pyo
[root@bms-zc4 ~]#
```

### 3.19.3 修改/etc/motd 配置项（可选）

#### 操作场景

修改motd配置项，可以有效地提醒用户在初次登录时，及时修改服务器的密码，以提高裸金属服务器的安全性。如果需要此特殊配置，可执行本节操作。

#### 操作步骤

1. 执行以下命令，在vi编辑器中打开或者新建“/etc/motd”。

```
vi /etc/motd
```

#### 说明

“/etc/motd”是存放Linux操作系统登录后的提示语的目录。

2. 按“i”进入编辑模式，在文件末尾增加登录时希望显示的内容。
3. 修改完成后，按“Esc”，输入:~保存并退出文件编辑。

### 3.19.4 修改历史记录配置项

#### 操作场景

通过修改“/etc/profile.d/history.sh”配置文件，对历史记录增加日期和时间排列，保存历史记录数为1000。

#### 说明

除EulerOS系统外，其他OS均适用。

#### 操作步骤

1. 执行以下命令，在vi编辑器中打开“/etc/profile.d/history.sh”。

```
vi /etc/profile.d/history.sh
```

2. 按“i”进入编辑模式，在文件中增加如下内容：

```
export HISTTIMEFORMAT="%F %T `whoami` "
export HISTSIZE=1000
```

3. 修改完成后，按“Esc”，输入:~保存并退出文件编辑。

### 3.19.5 优化 udev 配置和 Selinux

#### 优化 udev 配置

删除“/etc/udev/rules.d/70-persistent-net.rules”配置文件。



## 优化 Selinux

### 📖 说明

SUSE操作系统无selinux配置文件，跳过该项配置。

1. 执行以下命令，在vi编辑器中打开“/etc/selinux/config”。

**vi /etc/selinux/config**

2. 按“i”进入编辑模式，修改SELINUX=disabled。

```
This file controls the state of SELinux on the system.
SELINUX- can take one of these three values:
enforcing - SELinux security policy is enforced.
permissive - SELinux prints warnings instead of enforcing.
disabled - No SELinux policy is loaded.
SELINUX=disabled
SELINUXTYPE- can take one of three two values:
targeted - Targeted processes are protected.
minimum - Modification of targeted policy. Only selected processes
mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

3. 修改完成后，按“Esc”，输入:wq保存并退出文件编辑。

## 3.19.6 卸载 denyhosts

首先查看denyhosts是否已安装，如果安装了需要卸载。因为安装denyhosts会有功能性问题，如ssh登录失败。

参考以下命令查询并卸载denyhosts。

**rpm -qa | grep denyhosts**

**rpm -e denyhosts**

## 3.19.7 主机名自动更新问题设置（可选）

### 📖 说明

**问题现象：**重启后主机名发生变化，恢复为console名称或曾用名。/etc/hosts 中localhost恢复为console名称或曾用名。**建议：**为保证不再意外出现这个问题，建议修改console名称与hostname一致。

**检查方法：**

- 检查/etc/cloud/cloud.cfg文件中update\_hostname和manage\_etc\_hosts参数是否被注释（或者preserve\_hostname参数是否为true）。
- 检查NetworkManager中是否有/etc/NetworkManager/NetworkManager.conf中hostname-mode是否配置为none
- 检查/opt/huawei/network\_config/bms-network-config.conf中enable\_preserve\_hostname为True

**处理命令：**

- 集中式裸机网关：

# 避免network修改主机名

**sed -i 's/enable\_preserve\_hostname = False/enable\_preserve\_hostname = True/g' /opt/huawei/network\_config/bms-network-config.conf**

```
#避免/etc/hosts重启后被修改
sed -i '/manage_etc_hosts/s/^/#/g' /etc/cloud/cloud.cfg
#避免hostname更新
sed -i '/- update_hostname/s/^/#/g' /etc/cloud/cloud.cfg
• 分布式裸机网关：
#避免/etc/hosts重启后被修改
sed -i '/manage_etc_hosts/s/^/#/g' /etc/cloud/cloud.cfg
#避免hostname更新
sed -i '/- update_hostname/s/^/#/g' /etc/cloud/cloud.cfg
#避免NetworkManager修改名称
sed -i '/\[main\]/a\hostname-mode=none' /etc/NetworkManager/
NetworkManager.conf
```

### 3.19.8 安装常用运维工具（可选）

#### 说明

常见的运维必备内置软件持续更新中，常用的软件有：

gcc、perl、python2-pip、strace、sysstat、tcpdump、vim-common、vim-enhanced、vim-filesystem、wget、telnet

目前CentOS、RedHat系列支持连接互联网，使用yum源在线安装。根据常用软件列表，请依次执行下面命令进行安装：

```
yum install gcc
yum install perl
yum install python2-pip
yum install strace
yum install sysstat
yum install tcpdump
yum install vim-common
yum install vim-enhanced
yum install vim-filesystem
yum install wget
yum install telnet
```

### 3.19.9 设置登录口令有效期（可选）

1. 执行以下命令查看密码有效期。  
**vi /etc/login.defs**  
找到配置项“PASS\_MAX\_DAYS”，该参数表示密码的有效时间。

2. 执行以下命令，修改步骤1中“PASS\_MAX\_DAYS”参数的取值。  
**chage -M 99999 user\_name**  
其中，99999为密码有效期限，user\_name为系统用户。  
建议用户根据实际情况及业务需求进行配置，定期使用该命令更新密码有效期。
3. 再次执行**vi /etc/login.defs**，验证配置是否生效。

```
Password aging controls:
#
PASS_MAX_DAYS Maximum number of days a password may be used.
PASS_MIN_DAYS Minimum number of days allowed between password changes.
PASS_MIN_LEN Minimum acceptable password length.
PASS_WARN_AGE Number of days warning given before a password expires.
#
PASS_MAX_DAYS 99999
PASS_MIN_DAYS 0
PASS_MIN_LEN 5
PASS_WARN_AGE 7
```

## 3.20 配置裸金属服务器远程登录

裸金属服务器远程登录功能需要在操作系统中进行相关配置，使操作系统能够在串口上显示。本章节指导管理员在制作镜像时如何配置串口，使租户可以对裸金属服务器进行远程登录，不同操作系统配置方式有差异，请参考对应的章节进行配置。

### 3.20.1 x86: Oracle Linux 7.3/Oracle Linux 7.4/Red Hat 7 系列/CentOS 7.2/CentOS 7.4/CentOS 7.5/CentOS 7.6

#### 📖 说明

以下配置文件以CentOS 7.2操作系统为例，其他类型的配置文件视具体情况而定。

1. 使用vi编辑器打开“/etc/default/grub”，在GRUB\_CMDLINE\_LINUX字段内容后面添加“consoleblank=600 console=tty0 console=ttyS0,115200n8”，如下所示：

```
GRUB_TIMEOUT=5
GRUB_DISTRIBUTOR="$(sed 's, release .*$,g' /etc/system-release)"
GRUB_DEFAULT=saved
GRUB_DISABLE_SUBMENU=true
GRUB_TERMINAL_OUTPUT="console"
GRUB_CMDLINE_LINUX="crashkernel=512M rhgb quiet consoleblank=600 console=tty0
console=ttyS0,115200n8"
GRUB_DISABLE_RECOVERY="true"
~
~
```
2. 执行下面命令刷新。  
**stty -F /dev/ttyS0 speed 115200**  
**grub2-mkconfig -o /boot/grub2/grub.cfg**  
**systemctl enable serial-getty@ttyS0**
3. 如果要使用root用户通过串口进行登录，需要修改安全配置文件，在“/etc/securetty”最后添加下面的信息：

```
vc/1
...
vc/9
vc/10
vc/11
tty1
...
```

```
tty9
tty10
tty11
ttyS0
"securetty" 39L, 221C
```

### 说明

对于CentOS7系列，若按照以上正文配置出现串口乱码影响正常登录，如图3-2所示，处理方式如下。

图 3-2 登录乱码

```
[SQL Session operational. Use ~? for help]
```

```
CentOS Linux 7 (Core)
Kernel 3.10.0-957.el7.x86_64 on an x86_64
```

```
bms-jx login: root
Password: Hu
Login incorrect
```

```
bms-jx login: root
Password:
```

a.使用vi编辑器打开“/etc/default/grub”，在GRUB\_CMDLINE\_LINUX字段内容后面添加“115200”，如下所示：

```
[root@x6000-update ~]# cat /etc/default/grub | grep GRUB_CMDLINE_LINUX
GRUB_CMDLINE_LINUX="crashkernel=auto rhgb quiet console=tty0 console=ttyS0,115200"
[root@x6000-update ~]#
```

b.依次执行systemctl disable getty@ttyS0 systemctl stop getty@ttyS0命令，使getty@ttyS0服务状态如下：

```
[root@x6000-update ~]# systemctl disable getty@ttyS0
[root@x6000-update ~]# systemctl stop getty@ttyS0
[root@x6000-update ~]# systemctl status getty@ttyS0
● getty@ttyS0.service - Getty on ttyS0
 Loaded: loaded (/usr/lib/systemd/system/getty@.service; enabled; vendor preset: enabled)
 Active: inactive (dead)
 Docs: man:agetty(8)
 man:systemd-getty-generator(8)
```

c.执行stty -F /dev/ttyS0 speed 115200命令刷新波特率为115200，如图

```
[root@x6000-update ~]# stty -F /dev/ttyS0 speed 115200
115200
[root@x6000-update ~]#
```

d.再次执行grub2-mkconfig -o /boot/grub2/grub.cfg进行刷新即可。(grub.cfg文件所在目录以当前os中grub.cfg文件目录为准)

e.执行命令查询波特率为115200，如图所示：

```
stty -F /dev/ttyS0 -a
```

```
[root@x6000-update ~]# stty -F /dev/ttyS0 -a
speed 115200 baud; rows 24; columns 80; line = 0;
intr = ^C; quit = ^\; erase = ^?; kill = ^U; eof = ^D; eol = <undef>; eol2 = <undef>; swtch = <undef>; start = ^Q; stop = ^S;
susp = ^Z; rprnt = ^R; werase = ^W; lnext = ^O; flush = ^U; min = 1; time = 0;
- parenb -parodd -cmspar cs8 hupcl -cstoptb cread cllocal -crtscts
-ignbrk -brkint -ignpar -parmrk -inpck -istrip -inlcr -igncr -icrnl -ixon -ixoff -iucLc -ixany -imaxbel iutf8
opost -olcuc -ocrnl onlcr -onocr -onlret -ofill -ofdel nl0 cr0 tab0 bs0 vt0 ff0
-isig -icanon -lxten -echo -echoe -echok -echonl -noflsh -xcase -tostop -echoprtr -echoctl -echoke
[root@x6000-update ~]#
```

## 3.20.2 x86: Oracle Linux 6 系列/Red Hat 6 系列/CentOS 6 系列

### 说明

以下配置文件以Red Hat 6.7操作系统为例，其他类型的配置文件视具体情况而定。

1. 使用vi编辑器打开“/boot/grub/grub.conf”，找到关键字“hiddenmenu”，在“hiddenmenu”后面添加以下两行信息：

```
serial --unit=0 --speed=115200
terminal --timeout=5 serial console
```

然后在kernel行的尾部添加下面信息：

```
consoleblank=600 console=tty0 console=ttyS0,115200n8
```

```
#boot=/dev/sda
default=0
timeout=5
splashimage=(hd0,0)/grub/splash.xpm.gz
hiddenmenu
serial --unit=0 --speed=115200
terminal --timeout=5 serial console
title Red Hat Enterprise Linux 6 (2.6.32-573.el6.x86_64)
 root (hd0,0)
 kernel /vmlinuz-2.6.32-573.el6.x86_64 ro root=/dev/mapper/VolGroup
ot rd_NO_LUKS LANG=en_US.UTF-8 rd_NO_MD rd_LVM_LV=VolGroup/lo_swap SYSFONT
cyrheb-sun16 crashkernel=auto rd_LVM_LV=VolGroup/lo root KEYBOARDTYPE=pc
LE=us rd_NO_DM rhgb quiet console=tty0 console=ttyS0,115200n8
 initrd /initramfs-2.6.32-573.el6.x86_64.img

"grub.conf" 19L, 942C
```

2. 如果要使用root用户通过串口进行登录，需要修改安全配置文件，在“/etc/securetty”最后添加ttyS0。

## 3.20.3 x86: SUSE 11 SP4

1. 使用vi编辑器打开“/etc/inittab”，在文件最后添加如下信息：  
S0:12345:respawn:/sbin/agetty -L 115200 ttyS0
2. 修改文件“/boot/grub/menu.lst”，在“gfxmenu (hd0,1)/boot/message”后面添加以下信息：

```
serial --unit=0 --speed=115200
terminal --timeout=5 serial console
```

在kernel所在行的末尾添加下面信息，有两处都需要添加：

```
consoleblank=600 console=tty0 console=ttyS0,115200n8
```

```
Modified by YaST2. Last modification on Sat Jun 3 16:45:01 CST 2017
default 0
timeout 0
##YaST - generic_mbr
gfxmenu (hd0,1)/boot/message
##YaST - activate
serial --unit=0 --speed=115200
terminal --timeout=5 serial console
###Don't change this comment - YaST2 identifier: Original name: linux###
title SUSE Linux Enterprise Server 11 SP4 - 3.0.101-63
 root (hd0,1)
 kernel /boot/vmlinuz-3.0.101-63-default root=/dev/sda2 resume=/dev/sda1 splash=silent crashk
ernel=256M-:128M showopts vga=0x314 console=tty0 console=ttyS0,115200n8
 initrd /boot/initrd-3.0.101-63-default

###Don't change this comment - YaST2 identifier: Original name: failsafe###
title Failsafe -- SUSE Linux Enterprise Server 11 SP4 - 3.0.101-63
 root (hd0,1)
 kernel /boot/vmlinuz-3.0.101-63-default root=/dev/sda2 showopts ide=nodma apm=off noresume e
dd=off powersaved=off nohz=off highres=off processor.max_cstate=1 nonodeset x11failsafe vga=0x31
4 console=tty0 console=ttyS0,115200n8
 initrd /boot/initrd-3.0.101-63-default

###Don't change this comment - YaST2 identifier: Original name: floppy###
title Floppy
 rootnoverify (fd0)
 chainloader +1
```

3. 如果要使用root用户通过串口进行登录，需要修改安全配置文件，在“/etc/securetty”最后添加信息ttyS0。

### 3.20.4 ARM: CentOS 7

1. 使用vi编辑器打开“/etc/default/grub”，在GRUB\_CMDLINE\_LINUX字段后面添加如下信息：  
`consoleblank=600 console=tty0 console=ttyAMA0,115200`
2. 执行以下命令刷新配置。  
**stty -F /dev/ttyAMA0 speed 115200**  
**grub2-mkconfig -o /boot/efi/EFI/centos/grub.cfg**  
**systemctl enable serial-getty@ttyAMA0**
3. 如果要使用root用户通过串口进行登录，需要修改安全配置文件。确认“/etc/securetty”文件中是否存在ttyAMA0，如果不存在则添加。

### 3.20.5 x86: SUSE 12 系列/SUSE 15 系列/CentOS 7.3/EulerOS/OpenEuler/Oracle Linux 7.2

#### 说明

以下配置文件以SUSE 12 SP1操作系统为例，其他类型的配置文件视具体情况而定。

1. 使用vi编辑器打开“/etc/default/grub”，在GRUB\_CMDLINE\_LINUX字段后面添加如下信息：  
`consoleblank=600 console=tty0 console=ttyS0,115200`
2. 执行以下命令刷新配置。  
**stty -F /dev/ttyS0 speed 115200**  
**grub2-mkconfig -o /boot/grub2/grub.cfg**  
**systemctl enable serial-getty@ttyS0**
3. 如果要使用root用户通过串口进行登录，需要修改安全配置文件。在“/etc/securetty”文件最后添加ttyS0。

### 3.20.6 ARM: EulerOS/OpenEuler

1. 使用vi编辑器打开“/etc/default/grub”，在GRUB\_CMDLINE\_LINUX字段后面添加如下信息：  
`consoleblank=600 console=tty0 console=ttyAMA0,115200`
2. 执行以下命令刷新配置。  
**stty -F /dev/ttyAMA0 speed 115200**  
**grub2-mkconfig -o /boot/efi/EFI/euleros/grub.cfg**  
**systemctl enable serial-getty@ttyAMA0**
3. 如果要使用root用户通过串口进行登录，需要修改安全配置文件，确认“/etc/securetty”文件中是否存在ttyAMA0，如果不存在则添加。

### 3.20.7 x86: Ubuntu 16.04/Ubuntu 18.04

1. 使用vi编辑器打开“/etc/default/grub”，在GRUB\_CMDLINE\_LINUX字段内容的后面添加如下信息：  
`consoleblank=600 console=tty0 console=ttyS0,115200`



## 2. 执行下面命令刷新配置。

- 执行 `stty -F /dev/ttyS0 speed 115200` 命令刷新波特率为115200

```
root@bms-ubuntu:~# stty -F /dev/ttyS0 speed 115200
9600
root@bms-ubuntu:~# stty -F /dev/ttyS0 speed 115200
115200
```

- 执行 `stty -F /dev/ttyS0 -a` 命令查看当前波特率是否为115200

```
root@bms-ubuntu:~# stty -F /dev/ttyS0 -a
speed 115200 baud; rows 24; columns 80; line = 0;
intr = ^C; quit = ^\; erase = ^?; kill = ^U; eof = ^D; eol = <undef>; eol2 = <undef>; swtch = <undef>; start = ^Q; stop = ^S; susp = ^Z; rprnt = ^R;
werase = ^W; lnext = ^V; discard = ^O; min = 1; time = 0;
-parenb -parodd -cmspar cs8 hupcl -cstopb cread cllocal -crtscts
-ignbrk -brkint -ignpar -parmrk -mpck -istrip -inlcr -igncr -icrnl -ixon -ixoff -iucL -ixany -imaxbel iutf8
-ost -olcuc -ocrnl onlcr -onocr -onlret -ofill -ofdel n0 cr0 tab0 bs0 vt0 ff0
-isig -icanon -ixten -echo -echoe -echok -echonl -noflsh -xcase -tostop -echprt -echoctl -echoke -flusho -extproc
root@bms-ubuntu:~#
```

- 执行 `grub-mkconfig -o /boot/grub/grub.cfg` 命令

```
root@bms-ubuntu:~# grub-mkconfig -o /boot/grub/grub.cfg
Sourcing file `/etc/default/grub'
Sourcing file `/etc/default/grub.d/kdump-tools.cfg'
Generating grub configuration file ...
Found linux image: /boot/vmlinuz-4.15.0-55-generic
Found initrd image: /boot/initrd.img-4.15.0-55-generic
Adding boot menu entry for EFI firmware configuration
done
root@bms-ubuntu:~#
```

## 3. 使用vi编辑器打开“/etc/rc.local”，在文件中增加如下内容：

```
systemctl stop getty@ttyS0
```

## 4. 如果要使用root用户通过串口进行登录，需要修改安全配置文件。在“/etc/securetty”文件最后添加ttyS0。

5. 执行命令 `systemctl is-enabled serial-getty@ttyS0` 查看serial-getty@ttyS0状态设置为开机自启动

```
root@bms-ubuntu:~# systemctl is-enabled getty@ttyS0
disabled
root@bms-ubuntu:~# systemctl is-enabled serial-getty@ttyS0
enabled-runtime
root@bms-ubuntu:~#
```

### 3.20.8 ARM: Ubuntu 16.04、Ubuntu 18.04

## 1. 使用vi编辑器打开“/etc/default/grub”，按如下要求修改参数：

- GRUB\_CMDLINE\_LINUX参数为 `consoleblank=600 console=tty0 console=ttyAMA0,115200`
- GRUB\_TERMINAL参数为 `serial`
- GRUB\_SERIAL\_COMMAND参数为 `serial --speed=115200 --unit=0 --word=8 --parity=no --stop=1`

修改完成后执行以下命令：

**sudo update-grub2**

## 2. 如果要使用root用户通过串口进行登录，需要修改安全配置文件，在“/etc/securetty”最后添加下面的信息。

对于x86服务器，需要确认该文件中是否存在ttyS0；对于ARM64，需要确认是否存在ttyAMA0，如果不存在则添加。



### 3.20.9 x86: Ubuntu 14.04/Debian

1. 使用vi编辑器打开“/etc/default/grub”，在GRUB\_CMDLINE\_LINUX字段内容的后面添加如下信息：  
`consoleblank=600 console=tty0 console=ttyS0,115200`
2. 执行下面命令刷新配置。  
**stty -F /dev/ttyS0 speed 115200**  
**grub-mkconfig -o /boot/grub/grub.cfg**
3. 新建“/etc/init/ttyS0.conf”，使用vi编辑器修改文件内容如下：  
`start on stopped rc RUNLEVEL=[12345]  
stop on runlevel [!12345]  
respawn  
exec /sbin/getty -L 115200 ttyS0 vt102`
4. 执行以下命令启动ttyS0：  
**sudo start ttyS0**
5. 如果要使用root用户通过串口进行登录，需要修改安全配置文件。在“/etc/securetty”文件最后添加ttyS0。

## 3.21 配置根分区自动扩盘

### 3.21.1 配置说明

通过为操作系统安装growpart包，可以实现镜像下发后的根分区自动扩盘。SUSE系列系统默认自带了growpart，但是SUSE 11 SP4比较特殊，默认自带的growpart版本过高，需要手动安装低版本growpart包。

其他SUSE系列系统无需安装。Ubuntu系列系统默认自带了growpart，并且版本符合要求，因此无需安装growpart包。

#### 说明

如果镜像的启动方式为UEFI（如ARM架构镜像），还需要安装gdisk包，执行**yum install gdisk**命令进行安装。

### 3.21.2 CentOS 6/RedHat 6 系列

#### 说明

CentOS 6/RedHat 6系列要实现根分区自动扩盘，要安装三个包：cloud-init、cloud-utils-growpart、dracut-modules-growroot。其中，cloud-init已在[安装Cloud-Init](#)中安装。

本文以CentOS 6.9操作系统为例，其他操作系统的安装过程类似。

1. 检查是否已安装cloud-init、cloud-utils-growpart、dracut-modules-growroot。若没有，需要执行本章节操作进行安装。  
`[root@localhost ~]# rpm -qa | grep cloud-init  
cloud-init-0.7.5-10.el6.centos.2.x86_64  
[root@localhost ~]# rpm -qa | grep growpart  
[root@localhost ~]# rpm -qa | grep growroot  
[root@localhost ~]#`
2. 下载cloud-utils-growpart-0.27-10.el6.x86\_64.rpm包，并参考[上传需要的软件包到虚拟机](#)将安装包上传至虚拟机。  
下载网址：

<https://dl.fedoraproject.org/pub/epel/>

3. 执行以下命令，安装cloud-utils-growpart包。

```
rpm -ivh cloud-utils-growpart-0.27-10.el6.x86_64.rpm
```

```
[root@localhost redhat]# rpm -ivh cloud-utils-growpart-0.27-10.el6.x86_64.rpm
Preparing... ##### [100%]
 1:cloud-utils-growpart ##### [100%]
[root@localhost redhat]#
```

4. 安装dracut-modules-growroot包。

- a. 先执行以下命令：

```
yum install -y https://archives.fedoraproject.org/pub/archive/epel/6/x86_64/epel-release-6-8.noarch.rpm
```

- b. 执行**yum install dracut-modules-growroot**命令，在线安装dracut-modules-growroot包。

安装过程中，出现“Is this ok [y/N]”提示信息时，输入“y”。

- c. 安装完成后，执行**dracut -f**命令刷新内核。

5. 再次执行1中的命令，检查cloud-init、cloud-utils-growpart、dracut-modules-growroot包是否已安装成功。

### 3.21.3 CentOS 7/RedHat 7/Oracle Linux 7 系列

#### 📖 说明

CentOS 7/RedHat 7/Oracle Linux 7系列操作系统使用的是3.10 (> 3.8) 的内核，所以不需要安装dracut-modules-growroot包，只安装cloud-init和cloud-utils-growpart即可。其中，cloud-init已在[安装Cloud-Init](#)中安装。

本文以CentOS 7.3操作系统为例，其他操作系统的安装过程类似。

1. 检查是否已安装cloud-init、cloud-utils-growpart。若没有，需要执行本章节操作进行安装。

```
[root@localhost ~]# rpm -qa | grep cloud-init
cloud-init-0.7.5-10.el7.centos.1.x86_64
[root@localhost ~]# rpm -qa | grep growpart
[root@localhost ~]#
```

2. 在线yum安装cloud-utils-growpart。

```
yum install cloud-utils-growpart
```

3. 安装完成后，检查是否安装成功。

```
[root@localhost ~]# rpm -qa | grep growpart
cloud-utils-growpart-0.29-2.el7.noarch
```

### 3.21.4 CentOS/EulerOS ARM/OpenEuler ARM

#### 📖 说明

CentOS/EulerOS ARM系列要实现根分区自动扩盘，要安装三个包：cloud-init、cloud-utils-growpart、gdisk。其中，cloud-init已在[安装Cloud-Init](#)中安装。

本文以CentOS 7.6 ARM操作系统为例，其他操作系统的安装过程类似。

1. 检查是否已安装cloud-init、cloud-utils-growpart。若没有，需要执行本章节操作进行安装。

```
[root@localhost ~]# rpm -qa | grep cloud-init
cloud-init-0.7.5-10.el7.centos.1.x86_64
[root@localhost ~]# rpm -qa | grep growpart
[root@localhost ~]#
```

2. 在线yum安装cloud-utils-growpart。

**yum install cloud-utils-growpart**

3. 检查是否已安装gdisk。

**rpm -qa | grep gdisk**

```
[root@localhost ~]# rpm -qa | grep gdisk
[root@localhost ~]#
```

若没有，执行**yum install gdisk**进行安装。

4. 安装完成后，检查是否安装成功。

```
[root@localhost ~]# rpm -qa | grep growpart
cloud-utils-growpart-0.29-2.el7.noarch
[root@localhost ~]# rpm -qa | grep gdisk
gdisk-0.8.10-3.el7.x86_64
```

## 3.21.5 EulerOS/OpenEuler

### 说明

EulerOS不需要安装dracut-modules-growroot包，只安装cloud-init和cloud-utils-growpart即可。其中，cloud-init已在[安装Cloud-Init](#)中安装。

1. 检查是否已安装cloud-init、cloud-utils-growpart。若没有，需要执行本章节操作进行安装。

```
[root@localhost ~]# rpm -qa | grep cloud-init
cloud-init-0.7.6-3.x86_64
[root@localhost ~]# rpm -qa | grep growpart
[root@localhost ~]#
```

2. 下载cloud-utils-growpart-0.27-10.x86\_64.rpm包（下载网址：[http://repo.huaweicloud.com/euler/2.2/os/x86\\_64/Packages/](http://repo.huaweicloud.com/euler/2.2/os/x86_64/Packages/)），并将安装包上传至虚拟机，然后执行以下命令，安装cloud-utils-growpart包。

**rpm -ivh cloud-utils-growpart-0.27-10.x86\_64.rpm**

```
[root@bms-eulor22 home]# rpm -ivh cloud-utils-growpart-0.27-10.x86_64.rpm
Preparing... ##### [100%]
Updating / installing...
 1:cloud-utils-growpart-0.27-10 ##### [100%]
```

也可以使用ISO做本地repo源，执行**yum install cloud-utils-growpart**进行安装。

3. 再次执行1中的命令，检查cloud-utils-growpart包是否已安装成功。

```
[root@localhost ~]# rpm -qa | grep growpart
cloud-utils-growpart-0.27-10.x86_64
```

## 3.21.6 Debian

### 说明

Debian操作系统需要安装growroot包来实现自动扩盘。

1. 执行命令：**apt-get install cloud-initramfs-growroot**
2. 执行以下命令，查看cloud-initramfs-growroot包是否安装好。

**dpkg -l | grep cloud-initramfs-growroot**

回显如下类似信息表示安装成功。

```
root@bms:/home/bzqd# dpkg -l | grep cloud-initramfs-growroot
ii cloud-initramfs-growroot 0.18.debian5 all automatically resize the root partition on first boot
```

## 3.22 昇腾服务器驱动和固件安装升级

### 3.22.1 安装与维护

#### 3.22.1.1 安装前必读

NPU芯片是昇腾服务器的处理芯片，首次安装NPU驱动需使用20.2.0.spc300及以上版本。

NPU固件使用配套驱动同步发布的版本，且均不支持回退至20.2.0.spc200及更低版本。

登录HOST查询NPU芯片名称命令：*npu-smi info*

#### 3.22.1.2 安装前准备

##### 3.22.1.2.1 获取软件包

安装前，请根据运行环境的操作系统获取对应的固件包及驱动包。具体如表[获取软件包](#)所示

表 3-7 软件包名称

组件	操作系统	软件包名	执行用户
固件包	EulerOS 2.8/Ubuntu 18.04/ CentOS 7.6/CentOS 8.2/ BC_Linux 7.6/kylinosv10SP1/ BC_Linux 7.7  说明： kylinosv10SP1和BC_Linux 7.7在 21.0.rc1及以上版本才支持。	A800-9000-npu-firmware_x.x.x.run	root
驱动包	EulerOS 2.8+aarch64	A800-9000-npu-driver_x.x.x_euleros2.8-aarch64.run	root
驱动包	Ubuntu 18.04+aarch64	A800-9000-npu-driver_x.x.x_ubuntu18.04-aarch64.run	root
驱动包	CentOS 7.6+aarch64	A800-9000-npu-driver_x.x.x_centos7.6-aarch64-.run	root
驱动包	CentOS 8.2+aarch64	A800-9000-npu-driver_x.x.x_linux-aarch64.run	root

组件	操作系统	软件包名	执行用户
驱动包	BC_Linux 7.6+aarch64	A800-9000-npu-driver_x.x.x_centos7.6-aarch64-.run A800-9000-npu-driver_x.x.x_linux-aarch64.run	root
驱动包	kylinosv10SP1+aarch64 说明： kylinosv10SP1和BC_Linux 7.7在21.0.rc1及以上版本才支持。	A800-9000-npu-driver_x.x.x_linux-aarch64.run	root
驱动包	BC_Linux 7.7+aarch64 说明： kylinosv10SP1和BC_Linux 7.7在21.0.rc1及以上版本才支持。	A800-9000-npu-driver_x.x.x_linux-aarch64.run	root

📖 说明

- 表中的“x.x.x”表示具体版本号。
- A800-9000-npu-driver\_x.x.x\_linux-aarch64.run适配所有操作系统。

操作步骤

1. 登录**A800-9000**
2. 打开目标版本“A800-9000 X.X.X”。

固件、驱动与CANN的配套关系，请查看CANN软件版本的《**CANN 版本配套表**》。

3. 单击软件包（如A800-9000-npu-driver\_x.x.x\_euleros2.8-aarch64.run）后面的下载符号及数字签名符号，可获取软件包和数字签名文件

3.22.1.2.2 检验软件包完整性

为了防止软件包在传递过程或存储期间被恶意篡改，下载软件包时需下载对应的数字签名文件用于完整性验证。

在软件包下载之后，请参考《OpenPGP签名验证指南》，对从Support网站下载的软件包进行PGP数字签名校验。如果校验失败，请不要使用该软件包，先联系技术支持工程师解决。

使用软件包安装/升级之前，也需要按上述过程先验证软件包的数字签名，确保软件包未被篡改。

运营商客户请访问：<https://support.huawei.com/carrier/digitalSignatureAction>

企业客户请访问：<https://support.huawei.com/enterprise/zh/tool/pgp-verify-TL1000000054>

3.22.1.2.3 系统要求和环境检查

软件包要求的操作系统和内核，如表3-8和表3-9所示。

表 3-8 二进制驱动包默认操作系统版本信息

硬件形态	host操作系统版本	软件包默认的host操作系统内核版本	GCC编译器版本
aarch64+Atlas 800 9000	Ubuntu 18.04	4.15.0-45-generic 说明： 若内核版本和操作系统版本匹配失败，需先安装 dkms，安装方法请参见 <a href="#">驱动源码编译</a> 。	7.4.0
aarch64+Atlas 800 9000	EulerOS 2.8	4.19.36-vhulk1907.1.0.h475	7.3.0
aarch64+Atlas 800 9000	CentOS 7.6	4.14.0-115.el7a.0.1.aarch64	-
aarch64+Atlas 800 9000	BC_Linux 7.6	4.19	-

表 3-9 通用驱动包兼容操作系统版本信息

硬件形态	host操作系统版本	软件包默认的host操作系统内核版本	GCC编译器版本
aarch64+Atlas 800 9000	CentOS 8.2	4.18.X 说明： 支持升级到 5.6.14。	8.3.1
aarch64+Atlas 800 9000	BC_Linux 7.6	4.19	-
aarch64+Atlas 800 9000	kylinosv10SP1 说明： kylinosv10SP1在 21.0.rc1及以上版本才支持。	4.19.90-17.ky10.aarch64	8.3.1-4.5
aarch64+Atlas 800 9000	BC_Linux 7.7 说明： BC_Linux 7.7在 21.0.rc1及以上版本才支持。	4.19.25-203.e17.bclinux.aarch64	4.8.5

### 1. 验证Linux操作系统版本信息

执行**uname -m && cat /etc/\*release**命令，查询正在运行的操作系统版本和操作系统架构。

系统正在运行的操作系统版本和操作系统架构必须与[表3-8](#)或[表3-9](#)中要求一致。

### 2. 通用驱动包特有检查项

- 确认make工具是否已安装；执行**make -v**命令，若能查询到make工具的版本，则make已安装。
- 以下2个检查项满足其中任意一个，驱动就可以安装成功。
  - 确认系统已安装dkms等安装依赖；具体请参见[驱动源码编译](#)。
  - 确认默认内核源码路径“`/lib/modules/`uname -r`/build`”是否存在；执行**ls /lib/modules/`uname -r`/build**命令，查看路径是否存在。
    - 若存在，则会自动使用内核进行驱动编译。
    - 若不存在，您可以在安装时提供对应的源码路径，具体请参见[5](#)。

### 3. 验证Linux操作系统内核版本

执行**uname -r**命令，查询当前host操作系统的内核版本。

- 若使用二进制驱动包安装，当前host操作系统的内核版本必须与[表3-8](#)中要求一致。若不一致，有两种解决方法：
  - 用户需要重新进行源码编译，详细信息请参见[驱动源码编译](#)。
  - 参见下方[检查运行环境是否安装过软件包](#)检查系统是否安装过软件包，若未安装则直接升级系统内核版本；若安装过，则先卸载软件包，再升级内核。
- 若使用通用驱动包安装，当前host操作系统的内核版本与[表3-9](#)中要求一致，若不一致，可能导致驱动包安装不成功或者影响功能使用。

### 4. 检查运行环境是否安装过软件包

如需要升级系统内核版本，在升级之前，请确保当前系统上未安装软件包。否则会出现内核升级后软件包启动异常的问题。出现该问题时请参见[升级系统内核版本时未卸载软件包导致软件包不可用](#)解决。请使用**lsmod|grep drv**命令查询是否安装过软件包。

- 如无内容表示未安装过软件包，可以直接升级系统内核版本。
- 如有内容，表示安装过软件包，需要先卸载软件包后，升级系统内核版本。卸载请参见[卸载驱动和固件](#)。

#### 3.22.1.2.4 创建运行用户

- 若使用root用户安装

可以使用root用户进行安装，但安装完之后要求使用非root用户运行，所以安装前需要先创建运行用户。

  - 如果创建的用户是HwHiAiUser，安装软件包时可以直接使用该运行用户，默认即为HwHiAiUser。
  - 如果创建的用户是非HwHiAiUser，安装软件包时需要指定运行用户（通过**--install-username=username --install-usergroup=usergroup**参数指定）。
- 若使用非root用户安装



该场景下安装及运行用户必须相同。

- 若已有非root用户，则无需再次创建。

若想使用新的非root用户，则需要先创建该用户，请参见如下方法创建：  
创建非root用户操作方法如下，以下命令请以root用户执行。

- i. 创建非root用户。

```
groupadd usergroup
useradd -g usergroup -d /home/username -m username
```
- ii. 设置非root用户密码。

```
passwd username
```

 说明

- 指定运行用户是指定Driver的运行用户，Firmware不支持指定，共用Driver的运行用户。
- 上述命令中的username为用户名，请根据实际情况替换。
- 由于权限控制可能存在安全风险，创建的运行用户不建议为root用户属组。
- 创建完HwHiAiUser用户后，请勿关闭该用户的登录认证功能。
- 密码有效期为90天，您可以在/etc/login.defs文件中修改有效期的天数，或者通过chage命令来设置用户的有效期限，详情请参见[设置用户有效期](#)。

设置用户的有效期限

为保证用户的安全性，应设置用户的有效期限，使用系统命令**chage**来设置用户的有效期限。

命令为：

**chage [-m mindays] [-M maxdays] [-d lastday] [-I inactive] [-E expiredate] [-W warndays] user**

相关参数请参见。[表3-10](#)

表 3-10 设置用户有效期

参数	参数说明
-m	密码可更改的最小天数。设置为“0”表示任何时候都可以更改密码。
-M	密码保持有效的最大天数。设置为“-1”表示可删除这项密码的检测，则用户密码不会超期，存在安全风险，请谨慎使用。
-d	上一次更改密码的日期。
-I	停滞时期。过期指定天数后，设定密码为失效状态。
-E	用户到期的日期。超过该日期，此用户将不可用。
-W	密码到期前，提前收到警告信息的天数。
-l	列出用户以及密码的有效期限。

### 说明

- [表3-10](#)只列举出常用的参数，用户可通过**chage --help**命令查询详细的参数说明。
- 日期格式为YYYY-MM-DD，如**chage -E 2019-12-01 test**表示用户**test**的用户在2019年12月1日过期。
- User如果不填写，默认为root用户。

例如：修改用户**test**的有效期为2019年12月31日。可执行如下命令：

```
chage -E 2019-12-31 test
```

## 安装网络脚本

请参考[安装bms-network-config软件包](#)。

### 3.22.1.3 安装 nvme 驱动

nvme盘驱动获取地址：<https://support.huawei.com/enterprise/zh/category/accelerator-components-pid-1548148324389?submodel=software>

安装对应的hioadm和kmod-hiodriver：

1. rpm -ivh hioadm-xxx.rpm

```
[root@euler2u8-d918-b638 aarch64]# ls
hioadm-5.0.4.1-1.aarch64.rpm
[root@euler2u8-d918-b638 aarch64]# rpm -ivh hioadm-5.0.4.1-1.aarch64.rpm
Verifying... [100%]
Preparing... [100%]
Updating / installing...
 1:hioadm-5.0.4.1-1 [100%]
Install ES3000 hioadm-5.0.4.1 successfully.
```

2. rpm -ivh kmod-hiodriver-xxx.rpm

```
[root@euler2u8-d918-b638 aarch64]# uname -r
4.19.36-vmulk1987.1.0.b619.eulerosv2r8.aarch64
[root@euler2u8-d918-b638 aarch64]# rpm -ivh araf.19.36-vmulk1987/kmod-hiodriver-5.0.4.1-4.19.36-vmulk1987.1.0.b418.eulerosv2r8.e
uleros.aarch64.rpm
Verifying... [100%]
Preparing... [100%]
Updating / installing...
 1:kmod-hiodriver-5.0.4.1-4.19.36-vmulk1987 [100%]
Installing... Please wait for a moment.
Install the new mmc kernel module successfully.
Modify kdump image.Please wait for a moment...58%
Install kmod-hiodriver package successfully.
[root@euler2u8-d918-b638 aarch64]#
```

### 3.22.1.4 安装驱动和固件

#### 前提条件

请参见[安装前准备](#)。

#### 操作步骤

请按照“驱动 > 固件”的顺序，分别安装组件软件包。每个组件软件包的安装步骤相同，详细操作如下。软件包中的\*请根据实际情况进行替换。

1. 使用**root**用户登录到运行环境，将\*.run软件包上传至到运行环境任意路径下，如/opt下。
2. 增加安装用户对软件包的可执行权限。

在软件包所在路径执行**ls -l**命令检查安装用户是否有该文件的执行权限，若没有，请执行如下命令。

**chmod +x \*.run**

3. 执行如下命令，校验软件包安装文件的一致性和完整性。

**./\*.run --check**

4. 执行以下命令进行安装。

如果软件包安装路径是指定的，以“/test/HiAI/”为例。执行**./\*.run --full --install-path=/test/HiAI/**

如果软件包安装路径是默认的，执行**./\*.run -full**

以Euler2.8操作系统，安装驱动包为例（固件包安装方法同等）：进入软件包所在目录，执行**./A800-9000-npu-driver\_xxx\_euleros2.8-aarch64.run --full --install-for-all**（如果没有--install-for-all可能会导致非root用户使用的时候报证书问题）

#### 说明

- 驱动包支持在虚拟机环境安装，固件包仅支持在Atlas物理服务器上安装
- 如果指定root用户为运行用户，则需要与--install-for-all参数配合使用
- 软件包默认安装路径：/usr/local/Ascend
- 安装详细日志路径：/var/log/ascend\_seclog/ascend\_install.log
- 安装后Driver/Firmware的安装路径、安装命令以及运行用户信息记录路径：/etc/ascend\_install.info

5. 对于通用驱动包，若出现以下回显，说明未安装dkms，默认内核源码路径/lib/modules/`uname -r`/build不存在。请根据系统提示输入相应内容

[WARNING]rebuild ko has something wrong, detail in /var/log/ascend\_seclog/ascend\_rebuild.log Do you want to try build driver after input kernel absolute path? [y/n]:

若需要继续安装，请输入**y**。

出现如下回显后，请根据提示输入您提供的内核源码的实际路径，如：**/lib/modules/`uname -r`/build-bak**。

Please input your kernel absolute path:

按“Enter”键继续安装。

#### 说明

- 若已安装dkms及相应kernel-header、kernel-devel等组件，则会自动执行dkms驱动编译安装。
- 若未安装dkms，但默认内核源码路径/lib/modules/`uname -r`/build已存在，则会自动使用该内核进行驱动编译。

6. 若出现如下信息，则说明安装成功：

- 驱动：Driver package install success! Reboot needed for installation/upgrade to take effect!
- 固件：Firmware package install success! Reboot needed for installation/upgrade to take effect!

7. 重启运行环境。

8. 查看安装的驱动版本号。

在软件包的安装路径下，例如root用户默认路径“/usr/local/Ascend/{package\_name}”，执行如下命令查看所升级软件包版本是否正确。

cat version.info

```
Version=1.73.T105.0.B050
```

9. 查看安装的NPU固件版本号

/usr/local/Ascend/driver/tools/upgrade-tool --device\_index -1 --component -1 --version

```
Get component version(1.73.5.0.b050) succeed for deviceId(0), componentType(0). {"device_id":0,
"component":nve, "version":1.73.5.0.b050} Get component version(1.73.5.0.b050) succeed for deviceId(0),
componentType(3). {"device_id":0, "component":uefi, "version":1.73.5.0.b050} Get component
version(1.73.5.0.b050) succeed for deviceId(0), componentType(8). {"device_id":0, "component":imu,
"version":1.73.5.0.b050} Get component version(1.73.105.0.b050) succeed for deviceId(0),
componentType(9). {"device_id":0, "component":imp, "version":1.73.105.0.b050}
```

10. 执行npu-smi info查看NPU工具安装是否成功。

回显如下类似信息，说明安装成功。否则，说明安装失败，请联系技术支持处理。

```
[root@localhost ~]# npu-smi info

| npu-smi 1.5.8 | Version: 1.73.T205.0.B050 |

| NPU Name | Health | Power(W) | Temp(C) | Memory-Usage(MB) | HBM-Usage(MB) |
| Chip | | Bus-Id | AICore(%) | | | |

| 0 910 | OK | 66.1 | 36 | 156 / 15688 | 0 / 32255 |
| 0 0000:C1:00.0 | 0 | 0 | | | |

| 1 910 | OK | 62.6 | 33 | 313 / 15689 | 0 / 32255 |
| 0 0000:81:00.0 | 0 | 0 | | | |

| 2 910 | OK | 66.6 | 33 | 784 / 15689 | 0 / 32255 |
| 0 0000:41:00.0 | 0 | 0 | | | |

| 3 910 | OK | 259.7 | 35 | 2340 / 15600 | 0 / 32255 |
| 0 0000:01:00.0 | 0 | 0 | | | |

| 4 910 | OK | 65.8 | 35 | 313 / 15688 | 0 / 32255 |
| 0 0000:C2:00.0 | 0 | 0 | | | |

| 5 910 | OK | 64.2 | 37 | 470 / 15689 | 0 / 32255 |
| 0 0000:82:00.0 | 0 | 0 | | | |

| 6 910 | OK | 65.8 | 33 | 1725 / 15689 | 0 / 32255 |
| 0 0000:42:00.0 | 0 | 0 | | | |

| 7 910 | OK | 260.9 | 35 | 1248 / 15600 | 0 / 32255 |
| 0 0000:02:00.0 | 0 | 0 | | | |

```

📖 说明

- 步骤回显中“npu-smi”后面的字段为NPU工具版本号，“Version:”后面的字段为NPU驱动版本号。
- 其他命令使用请参考《Atlas 800 训练服务器 npu-smi 命令参考 (型号9000)》。

版本安装注意事项

- 查看日志时需注意：日志时间采用的是系统时间，device侧时间与host侧时间保持同步，修改host侧时间可以使用date命令。  
例如：执行date -s 17:55:55命令，将系统时间设定为17点55分55秒。

3.22.1.5 卸载驱动和固件

操作步骤

驱动和固件的卸载没有先后顺序。软件包中的\*请根据实际情况进行替换。

1. 使用root账户登录运行环境。
2. 支持两种卸载方式，用户根据实际情况选择一种方式卸载即可：

- 使用软件包进行卸载，请在run包存放目录下执行以下命令，如/opt:  
./\*.run --uninstall

- 在任意路径执行如下命令卸载。  
bash {install\_path}/{package\_name}/script/uninstall.sh

说明

- 步骤回显中“npu-smi”后面的字段为NPU工具版本号，“Version:”后面的字段为NPU驱动版本号。

• 其他命令使用请参考《Atlas 800 训练服务器 npu-smi 命令参考 (型号9000)》。

3. 如果卸载过程中无错误信息提示，则表示卸载成功，根据系统提示信息决定是否重启服务器，完成对软件包的卸载。

3.22.2 升级

3.22.2.1 升级前必读

升级影响

升级过程禁止进行其他维护操作动作。

Atlas 800 训练服务器（型号9000）软件版本升级过程中需要复位系统，会导致业务中断。

注意事项

Atlas 800 训练服务器（型号9000）版本升级时的注意事项如表3-11所示。

表 3-11 升级时注意事项

序号	描述
1	在进行升级操作之前，请仔细阅读本文档，确定已经理解全部内容。如果您对文档有任何意见或建议，请联系技术支持解决。
2	为了减少对业务的影响，请提前切走业务或在业务量低时进行升级操作。
3	升级后，请确保所有组件的版本保持一致。
4	需要确保系统已经正常运行且固件升级工具及依赖的驱动被正确加载。如果不能确保，请复位后再升级。相关检查方法请参见 <a href="#">如何检查device是否正常运行</a> 。

序号	描述
5	不建议用户修改“/etc/ascend_install.info”，如果修改，会导致系统功能不可用。
6	NPU C72B050（不含）之前版本升级至C72B050（含）之后版本，升级目标版本固件时请跳过<a href="#">升级昇腾服务器芯片固件</a>中的<b>重启系统</b>步骤，待升级完驱动包后（升级中也跳过<a href="#">升级昇腾服务器芯片驱动</a>中的<b>重启系统</b>步骤），进行冷复位操作（先下电再上电）。   NPU版本号查询命令：  <pre>/usr/local/Ascend/driver/tools/upgrade-tool --device_index -1 --component -1 --version</pre>

版本要求

建议使用同一个Atlas 800 训练服务器（型号9000）软件版本列表下驱动版本、固件版本，使二者保持配套关系。

升级流程

请按照“固件->驱动”的升级顺序，分别升级软件包。

3.22.2.2 升级前准备

升级前检查

升级操作前，请按照[表3-12](#)中序号顺序进行检查，并记录检查结果。

表 3-12 升级前检查表

序号	项目	检查标准
1	检查软件版本	1. 查询并记录当前系统中Atlas 800 训练服务器（型号9000）版本。   2. 确认需要升级的版本。
2	检查系统状态	检查Atlas 800 训练服务器（型号9000）告警：     - 未出现告警，可直接升级。   - 出现告警时，请与技术支持确认告警情况，然后再决定下一步动作。

不同芯片的软件版本支持情况如[表3-13](#)所示。

表 3-13 软件版本配套表

芯片名称	适配版本
昇腾服务器 A	20.0.0、20.1.0、20.2.0

芯片名称	适配版本
昇腾服务器 B	20.0.0、20.1.0、20.2.0
昇腾服务器 Pro A	20.0.0、20.1.0、20.2.0
昇腾服务器 Pro B	20.2.0.SPC300
昇腾服务器 Premium A	20.2.0.SPC300

获取升级包

请通过下述方法获取升级包。

1. 登录[A800-9000](#)。
2. 打开目标版本“A800-9000 X.X.X”。

固件、驱动与CANN的配套关系，请查看CANN软件版本的《[CANN 版本配套表](#)》。  
在指定版本页面下载对应升级包到客户端（本地PC）。获取的升级包如[表3-14](#)所示。

说明

- 下文中所有x.x.x代表软件版本号。
- A800-9000-npu-driver\_x.x.x\_linux-aarch64.run适配所有操作系统。

表 3-14 升级包信息

硬件形态	host操作系统版本	host操作系统内核版本	驱动包名称	固件包名称
aarch_64+Atlas 800训练服务器（型号9000）	EulerOS 2.8	4.19.36-vhulk1907.1.0.h475	A800-9000-npu-driver_x.x.x_euleros2.8-aarch64.run	A800-9000-npu-firmware_x.x.x.run
	CentOS7.6	4.14.0-115.el7a.0.1.aarch64	A800-9000-npu-driver_x.x.x_centos7.6-aarch64-.run	
	Ubuntu18.04	4.15.0-45-generic 说明： 若内核版本和操作系统版本匹配失败，需先安装dkms，安装方法请参见 <a href="#">驱动源码编译</a> 。	A800-9000-npu-driver_x.x.x_ubuntu18.04-aarch64.run	



硬件形态	host操作系统版本	host操作系统内核版本	驱动包名称	固件包名称
	CentOS8.2	4.18.X 说明： 支持升级到5.6.14。	A800-9000-npu-driver_x.x.x_linux-aarch64.run	
	BC_Linux 7.6	4.19	A800-9000-npu-driver_x.x.x_centos7.6-aarch64-.run A800-9000-npu-driver_x.x.x_linux-aarch64.run	
	kylinosv10 SP1 说明： kylinosv10 SP1在21.0.rc1及以上版本才支持。	4.19.90-17.ky10.aarch64	A800-9000-npu-driver_x.x.x_linux-aarch64.run	
	BC_Linux 7.7 说明： BC_Linux 7.7在21.0.rc1及以上版本才支持。	4.19.25-203.e17.bclinux.aarch64	A800-9000-npu-driver_x.x.x_linux-aarch64.run	

检查软件包完整性

为了防止软件包在传递过程或存储期间被恶意篡改，下载软件包时需下载对应的数字签名文件用于完整性验证。

在软件包下载之后，请参考《OpenPGP签名验证指南》，对从Support网站下载的软件包进行PGP数字签名校验。如果校验失败，请不要使用该软件包，先联系技术支持工程师解决。

使用软件包安装/升级之前，也需要按上述过程先验证软件包的数字签名，确保软件包未被篡改。

运营商客户请访问：<https://support.huawei.com/carrier/digitalSignatureAction>

企业客户请访问：<https://support.huawei.com/enterprise/zh/tool/pgp-verify-TL1000000054>

3.22.2.3 升级组件

### 3.22.2.3.1 升级昇腾服务器芯片固件

Atlas 800 训练服务器（型号9000）支持昇腾服务器芯片固件升级。本章节以Atlas 800 训练服务器（型号9000）的A800-9000-npu-firmware\_x.x.x.run包为例介绍升级固件的相关操作。

#### 操作步骤

1. 参照[升级前准备](#)准备软件包“A800-9000-npu-firmware\_x.x.x.run”。
2. 使用root账户登录Atlas 800 训练服务器（型号9000）所在服务器。
3. 将“A800-9000-npu-firmware\_x.x.x.run”上传至Linux系统任意目录下，如/opt下。
4. 进入软件包“A800-9000-npu-firmware\_x.x.x.run”所在路径，如/opt下。

**cd /opt**

5. 执行如下命令，更改“A800-9000-npu-firmware\_x.x.x.run”软件包权限。

**chmod u+x A800-9000-npu-firmware\_x.x.x.run**

6. 执行./A800-9000-npu-firmware\_x.x.x.run --check命令，校验run安装包的一致性和完整性。
7. 执行升级命令完成升级操作。

升级过程支持一键式升级操作，可执行./A800-9000-npu-firmware\_x.x.x.run --upgrade命令，完成升级。

出现以下类似回显，表示升级成功。

```
Firmware package install success! Reboot needed for installation/upgrade to take effect!
```

#### 📖 说明

- 软件包名称中的“x.x.x”表示固件版本号。
- 安装过程中的日志信息输出在“/var/log/ascend\_seclog/ascend\_install.log”文件中，用户可以执行**vim /var/log/ascend\_seclog/ascend\_install.log**命令打开日志。

8. 重启系统。

**reboot**

9. 检查升级后的版本号。

在软件包安装路径下，执行以下命令查看所升级软件包版本是否正确。

**cat version.info**

#### 📖 说明

- 软件包安装默认路径：“/usr/local/Ascend/firmware”。

### 3.22.2.3.2 升级昇腾服务器芯片驱动

#### 操作场景

该任务指导用户升级Atlas 800 训练服务器（型号9000）昇腾服务器芯片的驱动。

run升级包支持一键式升级，本章节以Atlas 800 训练服务器（型号9000）的A800-9000-npu-driver\_x.x.x\_euleros2.8-aarch64.run包为例说明，具体操作以客户实际host系统对应驱动包为准。

### 说明

- 驱动升级不会改变系统的用户名和密码。

### 对系统的影响

Atlas 800 训练服务器（型号9000）驱动版本升级过程中需要复位系统，会导致业务中断。为了减少对业务的影响，执行升级操作前请提前切走业务。

### 操作步骤

1. 参照[升级前准备](#)准备驱动包“A800-9000-npu-driver\_x.x.x\_euleros2.8-aarch64.run”。
2. 使用root账户登录Atlas 800 训练服务器（型号9000）所在服务器。
3. 将“A800-9000-npu-driver\_x.x.x\_euleros2.8-aarch64.run”上传至Linux系统任意目录下，如/opt下。
4. 进入软件包“A800-9000-npu-driver\_x.x.x\_euleros2.8-aarch64.run”所在路径，如/opt下。

### `cd /opt`

5. 执行如下命令，更改“A800-9000-npu-driver\_x.x.x\_euleros2.8-aarch64.run”软件包权限。

### `chmod u+x A800-9000-npu-driver_x.x.x_euleros2.8-aarch64.run`

6. 执行`./A800-9000-npu-driver_x.x.x_euleros2.8-aarch64.run --check`命令，校验run安装包的一致性和完整性。
7. 执行升级命令完成升级操作。

升级过程支持一键式升级操作，可执行`./A800-9000-npu-driver_x.x.x_euleros2.8-aarch64.run --upgrade`命令，完成升级。

出现以下类似回显，表示升级成功。

```
Driver package install success! Reboot needed for installation/upgrade to take effect!
```

### 说明

- 升级驱动时，会将动态库libdcmi.so和头文件dcmi\_interface\_api.h拷贝到“/usr/local/dcmi/”目录下。
  - 驱动升级过程中昇腾服务器驱动的日志信息输出在“/var/log/ascend\_seclog/ascend\_install.log”文件中。
8. 重启系统。

### reboot

9. 检查升级后的版本号。

在软件包安装路径下，执行以下命令查看所升级软件包版本是否正确。

### `cat version.info`

### 说明

- 软件包安装默认路径：“/usr/local/Ascend/driver”。
- 如果升级后无法登录host系统，请联系技术支持解决。

- 如果升级后不是目标版本或者升级失败，请重新进行升级。如果依然升级失败，请记录故障现象和操作步骤，并联系技术支持解决。

## 3.23 清理文件

### 清理上传文件

将上传至虚拟机中的文件删除，比如bms-network-config和SDI驱动的rpm软件包等。

### 清理临时文件

- 执行下面命令，清理用户登录记录。  
**echo > /var/log/wtmp**  
**echo > /var/log/btmp**
- 执行下面命令，清理相应目录下的临时文件。  
**rm -rf /var/log/cloud-init\***  
**rm -rf /var/lib/cloud/\***  
**rm -rf /var/log/network-config.log**  
**rm -rf /opt/huawei/network\_config/network\_config.json**  
**echo > /etc/hccn.conf**
- 执行下面命令，清理残留配置信息。
  - SUSE操作系统：查看“/etc/sysconfig/network/”文件夹下有哪些以“ifcfg”开头的文件，删除除了“ifcfg-lo及ifcfg.template”以外的以“ifcfg”开头的文件。  
查看文件命令：**ll /etc/sysconfig/network/**  
删除文件命令：**rm -rf /etc/sysconfig/network/ifcfgxxx**
  - RedHat/CentOS/Oracle/Euler操作系统：查看“/etc/sysconfig/network-scripts/”文件夹下有哪些以“ifcfg”开头的文件，删除除了“ifcfg-lo”以外的以“ifcfg”开头的文件。  
查看文件命令：**ll /etc/sysconfig/network-scripts/**  
删除文件命令：**rm -rf /etc/sysconfig/network-scripts/ifcfgxxx**
  - Ubuntu操作系统：**rm -rf /etc/network/interfaces.d/50-cloud-init.cfg**
- 执行下面命令清除历史操作记录。  
**history -w;echo > /root/.bash\_history;history -c;history -c;history -c;**

# 4 Windows 镜像制作流程

## 4.1 硬件和软件准备

首先需要准备一台Linux系统的物理服务器作为镜像制作服务器（即宿主机），以及一台用于登录宿主机的本地Windows跳板机（物理服务器或虚拟机）。

宿主机必须安装：

- vsftpd服务
- sshd服务
- vnc服务
- virt-manager工具
- kvm虚拟化软件

 说明

虚拟化管理工具virt-manager可以从开源官网下载安装，也可以在操作系统安装包里选择安装。

需要准备[表4-1](#)所示的软件，并且参考[裸金属服务器规格与驱动对应关系](#)确认对应机型/操作系统需要安装的驱动。

表 4-1 软件列表

名称	说明	获取路径
宿主机ISO镜像	CentOS 7.x	建议从官方网站获取。
操作系统ISO文件	要制作镜像的ISO文件，要制作镜像的ISO文件。裸金属服务器镜像支持的操作 系统见 <a href="#">表1-2</a> 和 <a href="#">表1-3</a> 。	建议从操作系统官方网站获取。

名称	说明	获取路径
bms-network-config网络配置程序软件包	用于网络自动化配置。	软件： <a href="https://bms-image-package.obs.cn-north-1.myhuaweicloud.com/bms-network-config-23.8.0.zip">https://bms-image-package.obs.cn-north-1.myhuaweicloud.com/bms-network-config-23.8.0.zip</a> SHA256校验码： <a href="https://bms-image-package.obs.cn-north-1.myhuaweicloud.com/bms-network-config-23.8.0.zip.sha256">https://bms-image-package.obs.cn-north-1.myhuaweicloud.com/bms-network-config-23.8.0.zip.sha256</a>
SDI卡驱动程序软件包	镜像中安装SDI卡前端驱动插件，使裸金属服务器能够支持挂载云硬盘，并且支持从云硬盘启动，实现快速发放能力。	<a href="https://support.huawei.com/enterprise/zh/software/250607156-ESW2000158115">https://support.huawei.com/enterprise/zh/software/250607156-ESW2000158115</a>
Cloudbase-Init初始化工具	Windows密码注入插件。	软件： <a href="https://bms-image-package.obs.cn-north-1.myhuaweicloud.com/windows/cloudbase_init/CloudbaseInitSetup_x64.msi">https://bms-image-package.obs.cn-north-1.myhuaweicloud.com/windows/cloudbase_init/CloudbaseInitSetup_x64.msi</a> SHA256校验码： <a href="https://bms-image-package.obs.cn-north-1.myhuaweicloud.com/CloudbaseInitSetup_x64.msi.sha256">https://bms-image-package.obs.cn-north-1.myhuaweicloud.com/CloudbaseInitSetup_x64.msi.sha256</a>
v5服务器驱动	v5服务器的板载网卡、RAID卡、主板驱动等。	访问 <a href="https://support.huawei.com/enterprise/zh/servers/fusionserver-idriver-pid-21588909/software">https://support.huawei.com/enterprise/zh/servers/fusionserver-idriver-pid-21588909/software</a> ，单击进入最新的版本，根据镜像OS类型，下载对应的驱动包。

 说明

裸金属服务器的内存大小最低要求为：比所制作的镜像文件大150MB。请确保所制作的镜像满足此要求。

需要准备的工具如[表4-2](#)所示。

表 4-2 工具列表

名称	说明	获取路径
跨平台远程访问工具	制作镜像过程中，用于访问宿主机的工具，如“Xshell”。	建议从官方网站获取。
文件传输工具	制作镜像过程中，用于为虚拟机传输文件的工具，如“Xftp”。	建议从官方网站获取。

名称	说明	获取路径
VNC工具	制作镜像过程中，用于登录虚拟机的工具，如“VNC Viewer”。   <b>说明</b> 如果VNC Viewer为5.3.2版本，请在连接页面选择“Options &gt; Expert &gt; ColorLevel”，将ColorLevel值设置为“rgb222”，否则可能连接不上远端操作系统。	建议从官方网站获取。

## 4.2 安装 v5 服务器驱动（可选）

### 4.2.1 准备工作

Windows驱动安装依赖于硬件设备，通过使用Dism++软件在ISO镜像文件中注入驱动的方式进行安装。之后使用注入完成的ISO进行cloudbase-init、bms-network-config等软件的安装。

在安装驱动前需要准备如下工作：

1. 在本地Windows跳板机上准备对应版本Windows系统的ISO镜像文件。
2. 在本地Windows跳板机上安装Dism++软件。
3. 参考[硬件和软件准备](#)章节下载FusionServer服务器驱动至本地Windows跳板机。  
例如：
  - Windows Server 2012 R2: FusionServer iDriver-Win2K12R2-Driver-V113.zip
  - Windows Server 2016: FusionServer iDriver-Win2K16-Driver-V115.zip
4. 解压该zip驱动包，获取ISO驱动文件。
  - Windows Server 2012 R2: 解压“FusionServer iDriver-Win2K12R2-Driver-V113.zip”，获取“onboard\_driver\_win2k12r2.iso”文件。
  - Windows Server 2016: 解压“FusionServer iDriver-Win2K16-Driver-V115.zip”，获取“onboard\_driver\_win2k16.iso”文件。

### 4.2.2 安装 chipset 驱动

#### 操作背景

chipset为Windows主板驱动，如果不安装会影响其他硬件的性能发挥，还可能导致其他硬件的驱动工作不正常。

#### 说明

本文以Windows Server 2016为例，Windows Server 2012 R2与之类似。

#### 操作步骤

1. 解压缩[准备工作](#)中的“onboard\_driver\_win2k16.iso”文件，获取含有chipset字样的文件夹（如“Chipset-Win2K16-XXX”）中的“SetupChipset.exe”。



## 2. 将chipset安装包转换成文件夹。

- a. 单击“开始”菜单，在“搜索程序和文件”中输入“cmd”，并按“Enter”，打开命令提示符窗口。
- b. 进入chipset安装包目录，此处以“D:\windows2016\tmp”为例。

```
D:\windows2016\tmp>dir
2018/04/08 17:25 <DIR> .
2018/04/08 17:25 <DIR> ..
2018/03/08 15:14 <DIR> drivers1
2018/03/08 11:28 3,525,008 SetupChipset.exe
```

```
D:\windows2016\tmp>
```

- c. 执行以下命令，生成一个名为driver的文件夹。

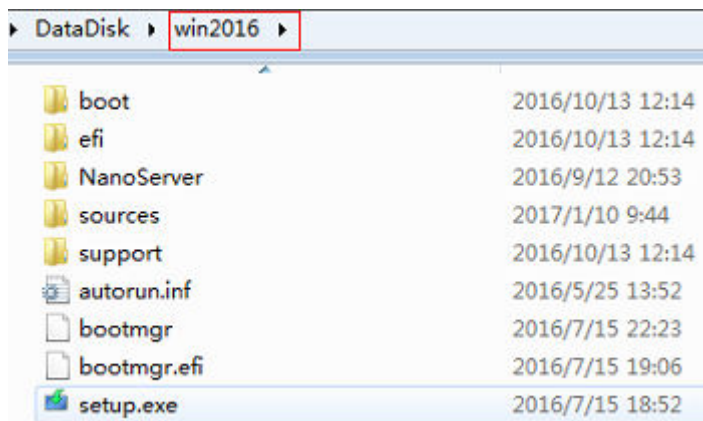
```
.\SetupChipset.exe -extract driver
```

```
D:\windows2016\tmp>.\SetupChipset.exe -extract driver
```

```
D:\windows2016\tmp>dir
2018/04/08 17:32 <DIR> .
2018/04/08 17:32 <DIR> ..
2018/04/08 17:32 <DIR> driver
2018/03/08 15:14 <DIR> drivers1
2018/03/08 11:28 3,525,008 SetupChipset.exe
```

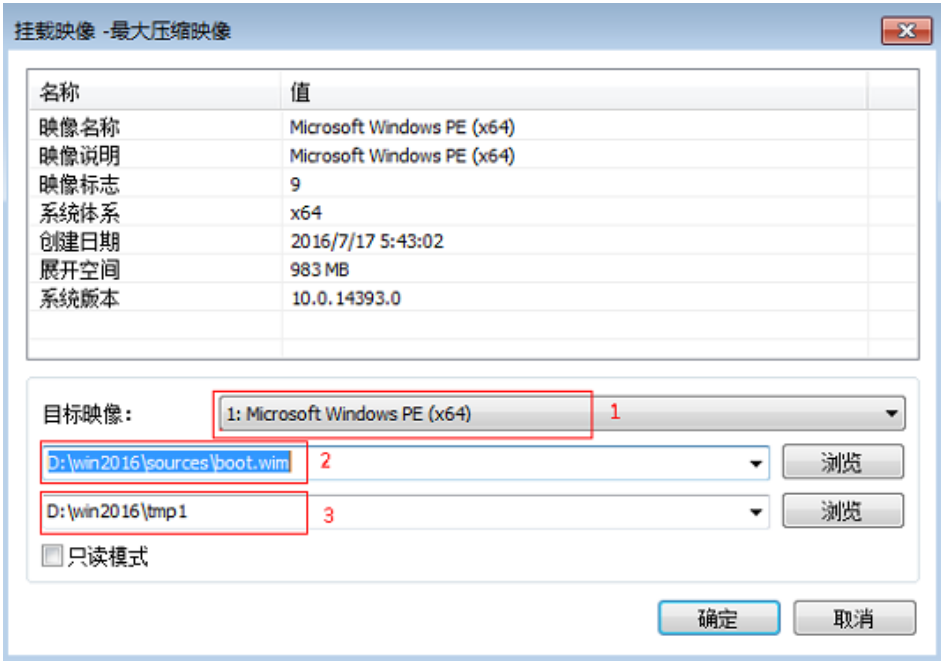
```
D:\windows2016\tmp>
```

## 3. 解压Windows 2016的ISO镜像文件至一个文件夹内。

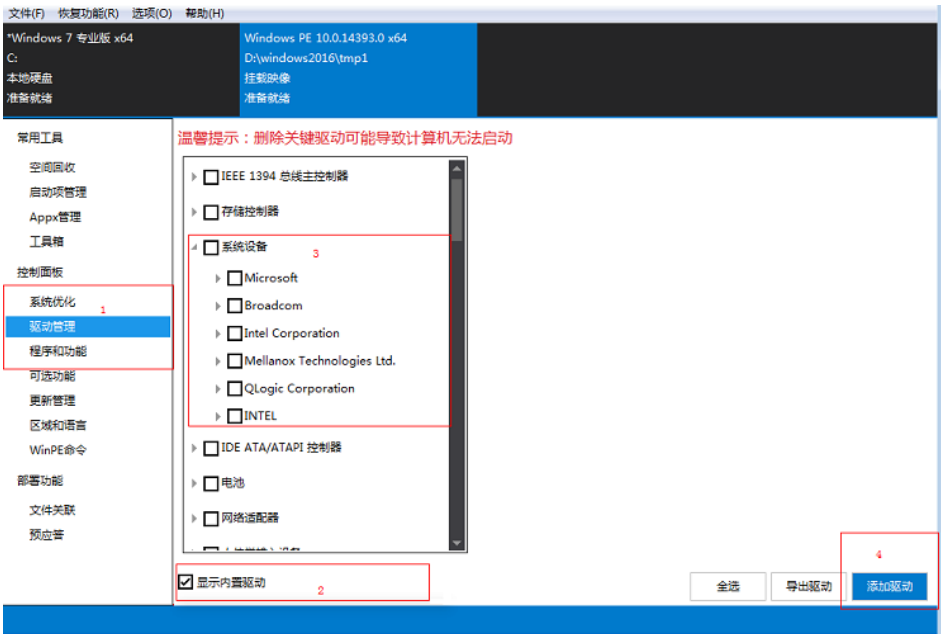


## 4. 挂载boot.wim文件镜像进行驱动注入。

- a. 在Dism++页面，单击“文件 > 挂载映像”。
- b. 在“挂载映像”界面，按下图1、2、3顺序进行操作。



- 1：选择目标映像。
  - 2：选择3中解压的sources文件夹中的boot.wim。
  - 3：选择一个新建的空文件夹，如本例中的tmp1。
  - 4：单击“确定”，提示“正在挂载”。
- c. 待状态变为“准备就绪”后，单击页面中间的“打开会话”。
- d. 按下图1、2、3、4顺序添加驱动。



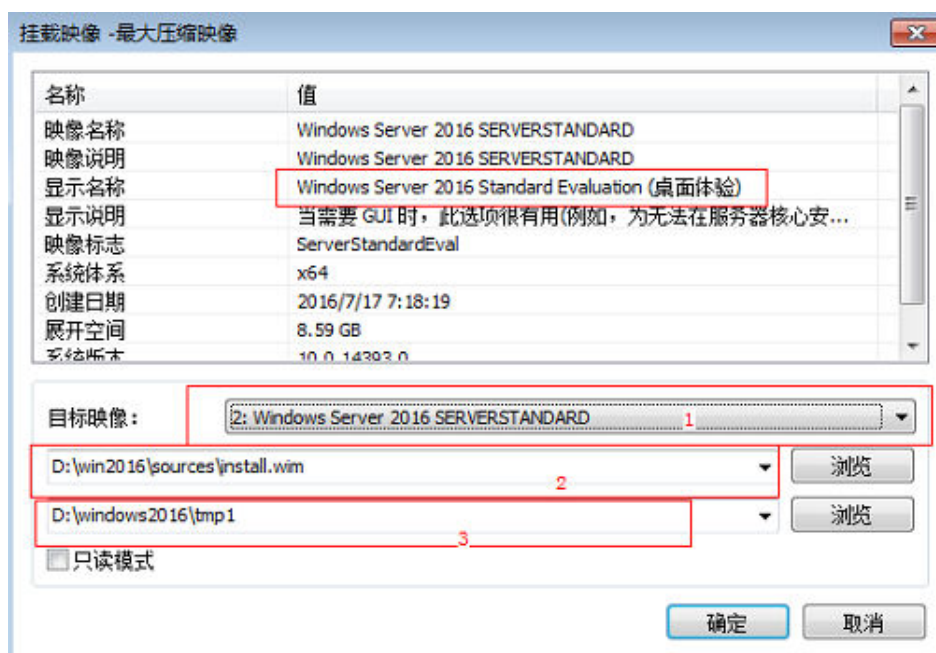
- 1：单击左侧导航中的“驱动管理”。
- 2：勾选“显示内置驱动”。
- 3：勾选“系统设备”。
- 4：单击“添加驱动”。

- e. 选择2中chipset安装包转换后的文件夹。

弹出如下提示框：



5. 保存添加好的驱动，并卸载映像。
- 单击“文件 > 保存映像”，在弹出的对话框中选择“直接保存”。
  - 等待保存成功。  
页面上方的状态由“正在保存”变为“就绪”，表示保存成功。
  - 单击“文件 > 卸载映像”。
6. 挂载install.wim文件镜像进行驱动注入。
- 在Dism页面，选择“文件 > 挂载映像”。
  - 在“挂载映像”界面，按下图1、2、3顺序进行操作。



- 1: 默认选择Windows Server 2016的标准安装。您可以根据所需安装的版本类型自行选择，如需安装其他类型，单击右侧的下拉箭头进行选择。
  - 2: 选择3中解压的sources文件夹中的install.wim。
  - 3: 选择一个新建的空文件夹，如本例中的tmp1。
- c. 待状态变为“准备就绪”后，单击页面中间的“打开会话”。
  - d. 按照4.d~4.e中的操作添加驱动。

7. 按照5中的操作保存驱动并卸载映像。

### 4.2.3 安装 v5 服务器板载网卡驱动（可选）

#### 操作背景

安装X722板载网卡驱动软件包，使裸金属服务器支持在v5服务器上下发。其他类型服务器可跳过此步骤。

#### 说明

本文以Windows Server 2016为例，Windows Server 2012 R2与之类似。

#### 操作步骤

1. 解压缩[准备工作](#)中的“onboard\_driver\_win2k16.iso”文件，找到其中的Intel网卡所在的压缩包或文件夹，如果是压缩包，如“NIC-82599\_I350\_X540\_X550\_X710\_X722\_XL710\_XXV710-Win2K16-XXX”，解压之后获取其中的“PRO40GB\Winx64\NDIS64”文件夹。
2. 参考[安装chipset驱动](#)章节的步骤4，挂载boot.wim文件镜像，添加NDIS64文件夹中的驱动。
3. 参考[安装chipset驱动](#)章节的步骤5，保存完成后，再进行卸载。
4. 参考[安装chipset驱动](#)章节的步骤6，挂载install.wim文件镜像。
5. 参考[安装chipset驱动](#)章节的步骤3.2.2-7，保存完成后进行卸载。

### 4.2.4 3408/3508 RAID 卡驱动安装（可选）

#### 操作背景

如果裸金属服务器要在3408/3508 RAID卡机型上下发，需要安装3408/3508 RAID卡驱动。其他类型服务器可跳过此步骤。

#### 说明

本文以Windows Server 2016为例，Windows Server 2012 R2与之类似。

#### 操作步骤

1. 解压缩[准备工作](#)中的“onboard\_driver\_win2k16.iso”文件，获取包含RAID及megasas字样的文件夹（如“RAID-3408iMR\_3416iMR\_3508\_3516-Win2K16-megasas35-XXX”）。
2. 参考[安装chipset驱动](#)章节的步骤4，挂载boot.wim文件镜像，添加“RAID-3408iMR\_3416iMR\_3508\_3516-Win2K16-megasas35-7.716.3.0-x86\_64”文件夹中的驱动
3. 参考[安装chipset驱动](#)章节的步骤5，保存完成后，再进行卸载。
4. 参考[安装chipset驱动](#)章节的步骤6，挂载install.wim文件镜像。
5. 参考[安装chipset驱动](#)章节的步骤3.2.2-7，保存完成后进行卸载。

## 4.3 通过 Dism++为 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。

本节操作以使用Dism++为例，介绍如何为ISO文件集成virtio驱动。

### 前提条件

已准备ISO文件。

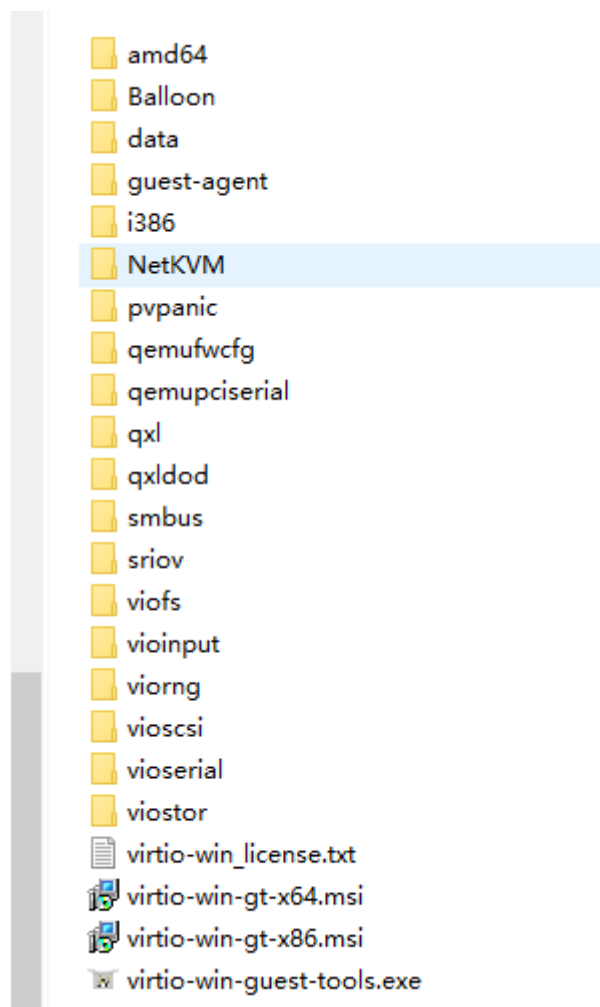
#### 说明

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。

### 操作步骤

1. 下载virtio驱动。  
其他版本获取地址如下：  
<https://fedorapeople.org/groups/virt/virtio-win/direct-downloads/archive-virtio/>
2. 解压1中下载的“virtio-win.iso”文件，解压后得到“virtio-win-0.1.189”文件夹，如图4-1所示。

图 4-1 “virtio-win-0.1.189” 文件夹

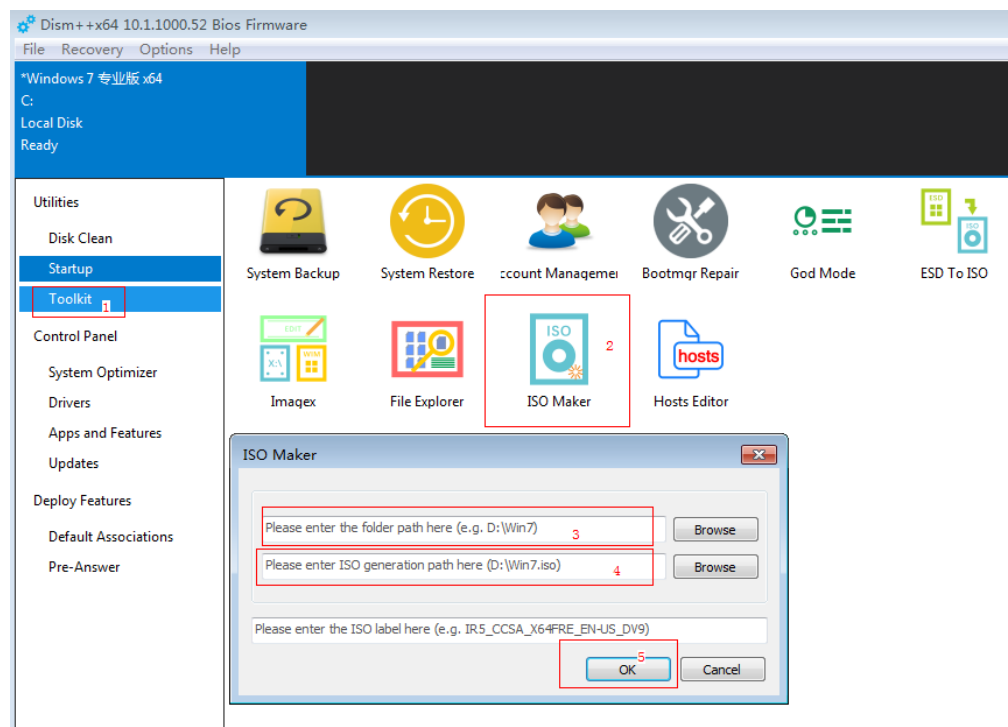


3. 解压缩ISO镜像文件，并将2中解压的“virtio-win-0.1.189”文件夹的所有文件复制到ISO文件父节点下，和boot等文件夹同级目录。

## 4.4 ISO 镜像合成

在Dism++中，将已安装VMTools驱动的ISO文件导出至本地，即另存为iso格式。

按下图1、2、3、4、5顺序进行操作。



- 1：单击“工具箱”。
- 2：选择“ISO生成器”，弹出对话框。
- 3：选择原始ISO解压后的路径。
- 4：选择新生成ISO的路径。
- 5：单击“确定”，等待合成结束。

## 4.5 创建 windows 虚拟机

### 4.5.1 使用 virt-manager 工具创建虚拟机

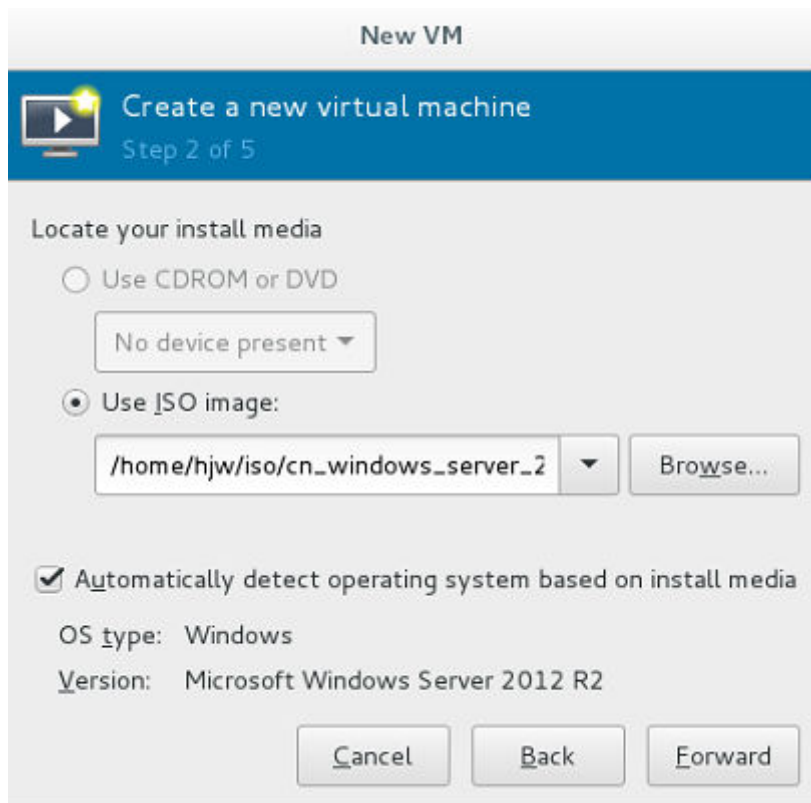
#### 4.5.1.1 创建并安装虚拟机（Windows）

使用virt-manager创建虚拟机，以Windows Server 2012 R2虚拟机操作系统为例，其他操作系统类似。以下步骤图示仅供参考，视实际情况而定。

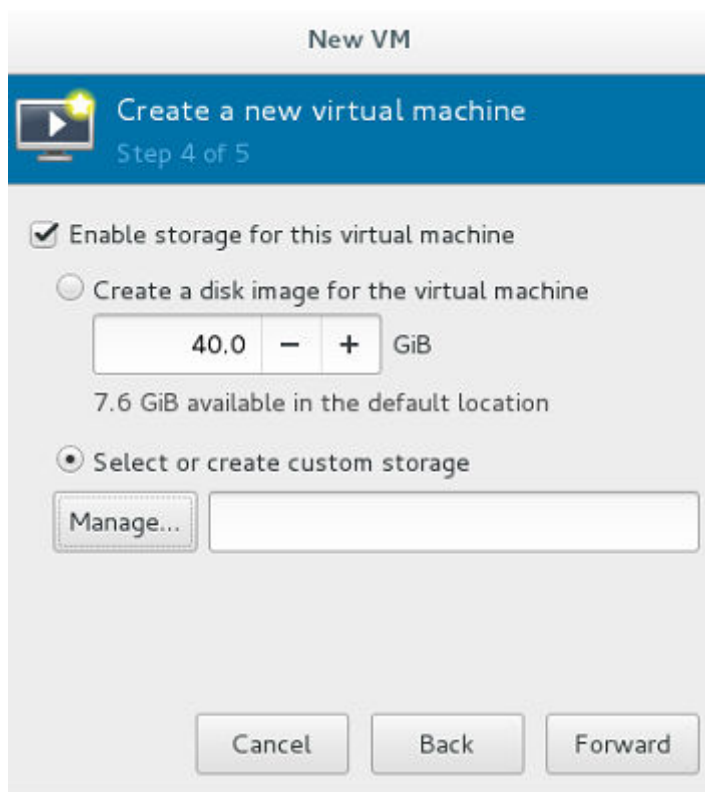
1. 登录镜像制作服务器，将ISO镜像合成中保存的ISO文件上传到服务器系统中,并启动virt-manager。

```
[root@localhost Desktop]# virt-manager
[root@localhost Desktop]#
```
2. 新建虚拟机，选择“Local install media (ISO image or CDROM)”，单击“Forward”。
3. 选择放置在宿主机上的ISO镜像文件，并选择相应的操作系统类型和版本。

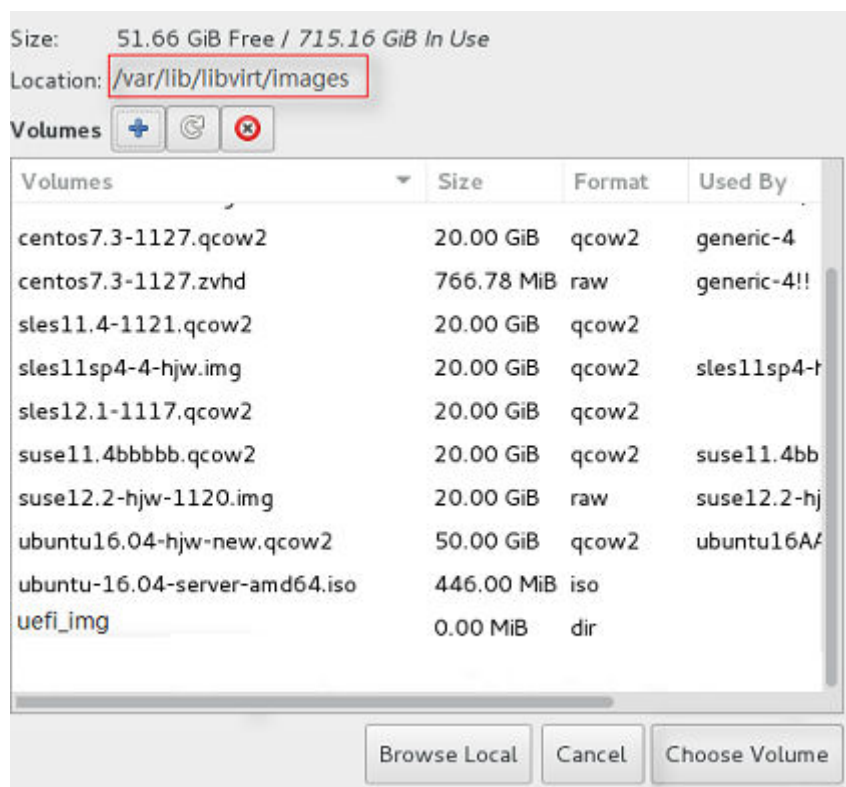




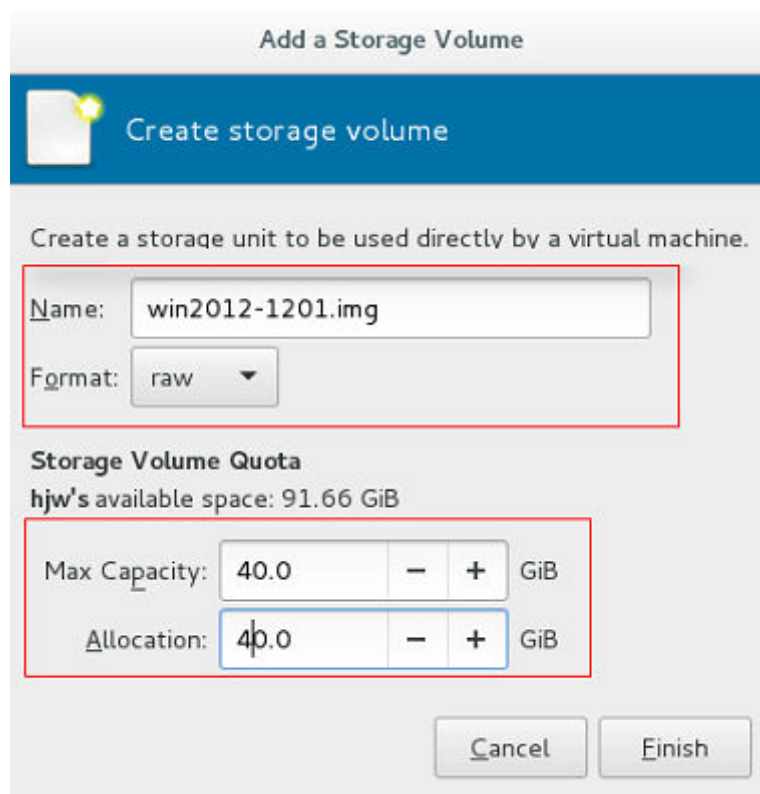
4. 配置虚拟机Window操作系统，内存和CPU大小。
  - Memory (RAM): 4096 MiB
  - CPUs: 4
5. 配置创建磁盘映像大小（不能选太大，避免生成的镜像文件太大）。



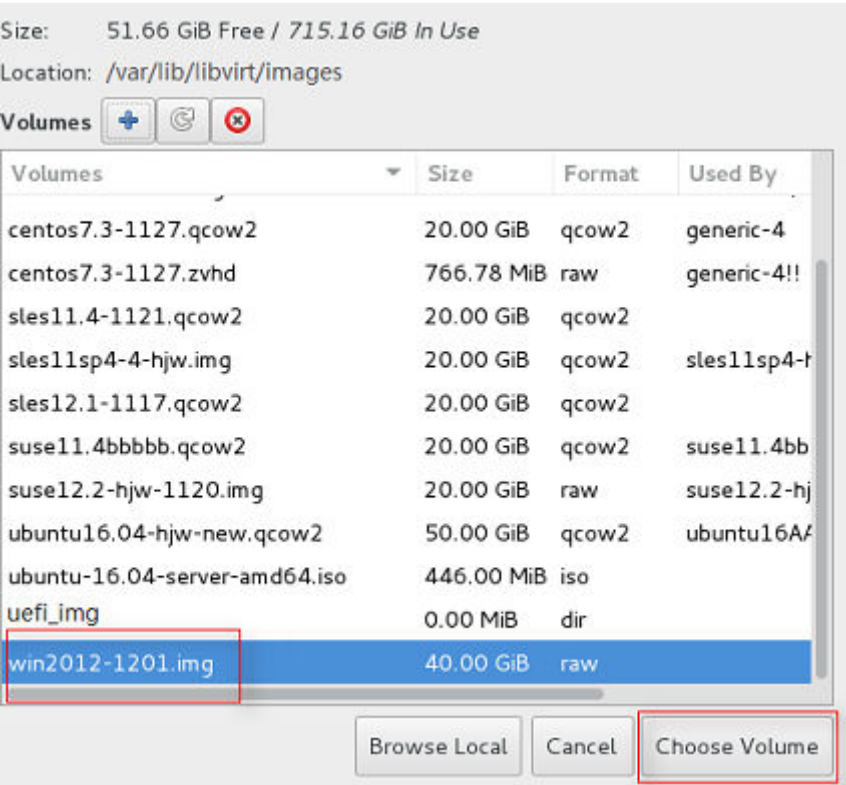
6. 单击上一步的“Manage”，在左侧目录列表选择镜像最终的保存位置，如本例选择“/var/lib/libvirt/images”。



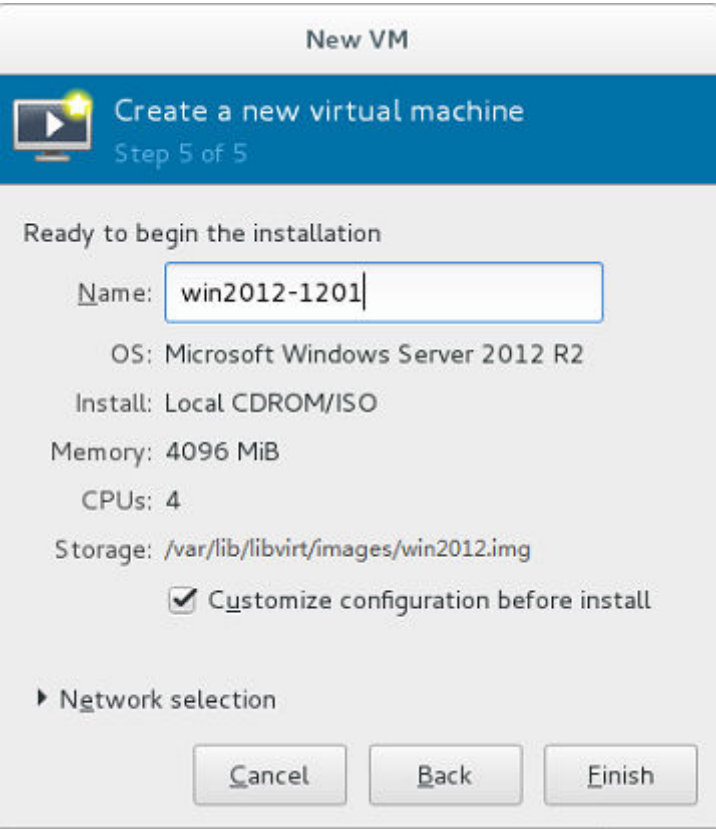
单击  图标创建存储卷，建议Format选择raw，名称可自定义，建议为img格式，方便镜像太大时进行压缩。



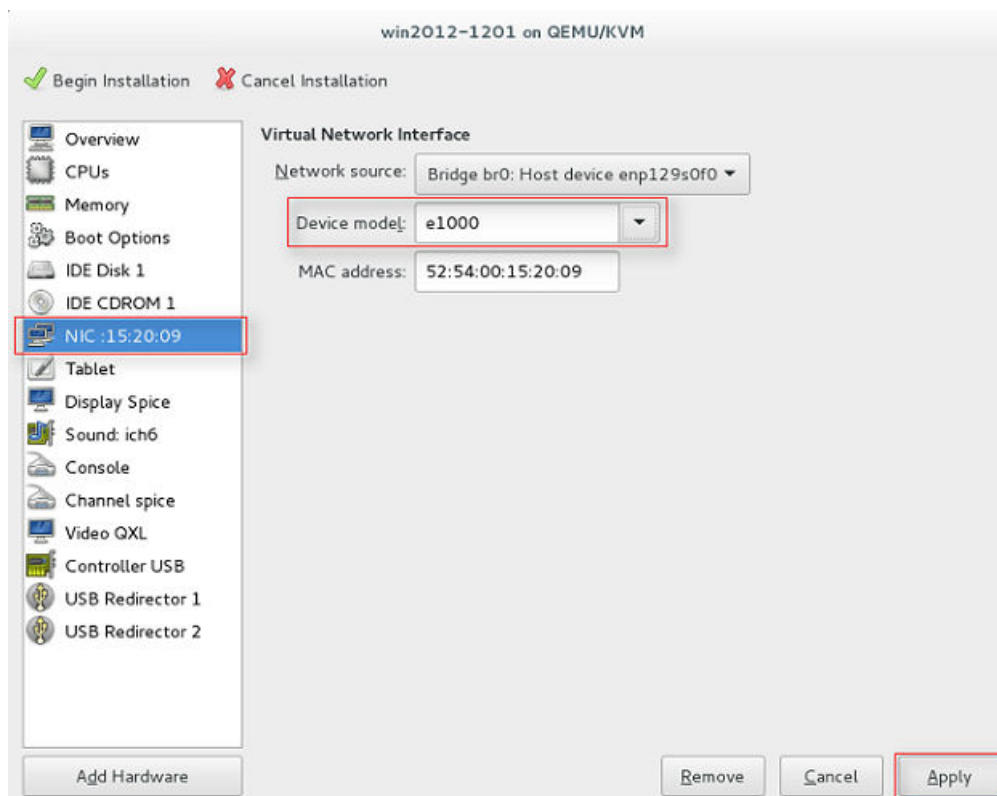
单击“Finish”，返回存储卷列表，选择刚刚创建的卷，单击“Choose Volume”。



7. 选择自定义配置。



8. 选择网卡配置，选择设备型号为e1000，单击“Apply”。



9. 单击左上角的“Begin Installation”开始安装。  
10. 安装Windows操作系统。

参考[安装Windows操作系统和virtio驱动](#)，选择要安装的语言、时间和货币格式等参数，根据所购买的Windows Server 2012 R2操作系统产品密钥对系统进行激活，其他配置项根据实际需求来设置。

## 4.5.2 使用镜像服务的 ISO 镜像功能创建虚拟机

### 4.5.2.1 注册 ISO 文件（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，需先将合成镜像的ISO文件上传到OBS桶中。

#### 前提条件

- 待注册文件必须为iso格式。
- 已上传ISO镜像文件到个人OBS桶中。

#### 📖 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。

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，创建方式选择“导入私有镜像”，镜像类型选择“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。

图 4-2 使用 ISO 文件创建私有镜像

镜像类型和来源

区域

不同区域的云服务产品之间内网互不相通；请就近选择靠近您业务的区域，可减少网络时延，提高访问速度。

创建方式

创建私有镜像 导入私有镜像

镜像类型

系统盘镜像 数据盘镜像 ISO镜像

选择镜像文件

创建私有镜像使用的文件需要上传到对象存储为标准类型的情中，从情中选取的镜像文件的实际大小不能超过128GiB。

ISO文件创建的ISO镜像在云服务器界面不可用。

所创建的私有镜像的格式和大小可能跟您的原始镜像文件不同。

创建镜像前，请确保Windows操作系统的ISO文件已集成UVP VMTools。

桶名称

标准存储

2023/11/21 15:28:56 GMT+08:00

4. 在“配置信息”区域，填写镜像的基本信息。

图 4-3 配置镜像信息

配置信息

架构类型

x86 ARM

启动方式

BIOS 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。

操作系统

请选择操作系统 请选择操作系统版本

系统盘 (GiB)

10

请确保输入的大小不小于镜像文件的系统盘大小。

名称

企业项目

请选择

标签

如果您需要使用同一标签标识多种云资源，即所有服务均可在标签输入框下拉选择同一标签，建议在TMS中创建预定义标签。

标签键 标签值

您还可以添加10个标签。

描述

0/1,024

协议

我已经阅读并同意《华为镜像免责声明》

- 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。
- 许可证类型：支持使用自带许可证（BYOL）。如果您已有相应操作系统的授权许可证，创建弹性云服务器后可直接使用已有许可证，无需重新申请。该参数仅在操作系统为Windows时可见。
- 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。
- 企业项目：选择资源所属的企业项目。
- 标签：可选配置，根据需要为待创建私有镜像添加资源标签。

#### 说明

如您的组织已经设定了镜像的相关标签策略，则需按照标签策略规则为私有镜像添加标签。标签如果不符合标签策略的规则，则可能会导致私有镜像标签创建失败，请联系组织管理员了解标签策略详情。

- 每个标签由键值对组成，标签的key的长度不超过36个字符，value的长度不超过43个字符。key不能为空或空白字符串，value不能为空，但可以是空白字符串。
  - 单个镜像最多添加10个标签。
  - 描述：可选配置，根据需要输入描述信息。
5. 单击“立即创建”。
  6. 根据界面提示，确认镜像参数。阅读并勾选《镜像免责声明》，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

## 4.5.2.2 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。更多信息详见[专属云](#)。

## 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”，创建弹性云服务器。
- 由于此云服务器仅作为临时云服务器使用，最终需要删除。因此，系统会默认创建一个固定规格且“按需付费”的云服务器，使用该云服务器创建的私有镜像再次创建云服务器时不会限制规格和付费方式。

#### 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. 根据界面提示完成弹性云服务器的配置，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。

### 4.5.2.3 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。

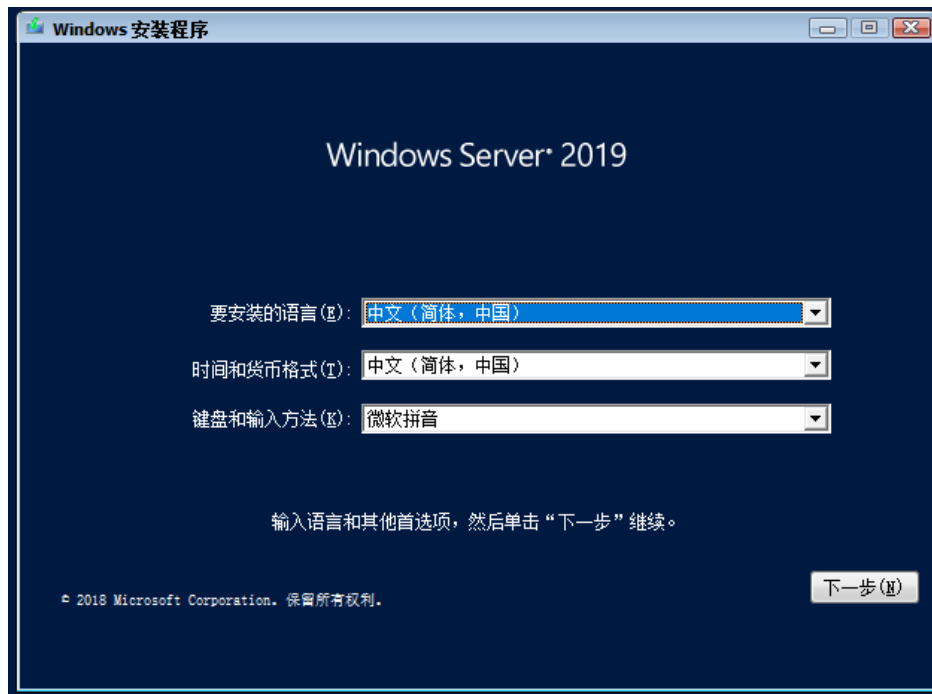
---



**步骤1** 安装Windows操作系统。

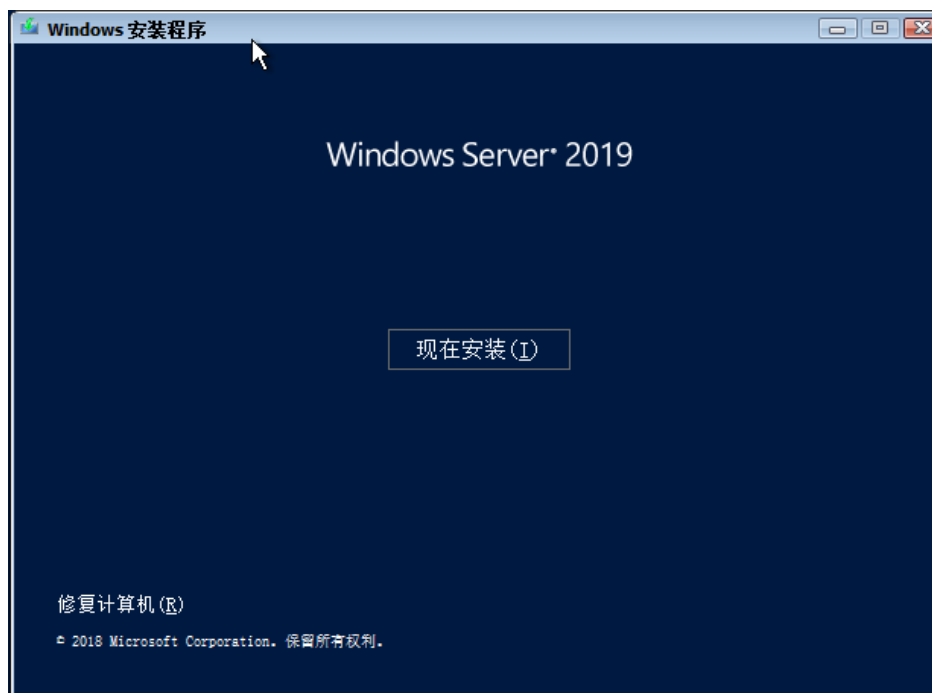
1. 在安装界面设置安装首选项，如图4-4所示。

**图 4-4** 安装首选项



2. 单击“下一步”。  
进入安装确认界面，如图4-5所示。

**图 4-5** 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。
5. 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4-6所示。

图 4-6 安装类型



6. 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4-7所示，则执行步骤1.7。

图 4-7 安装位置



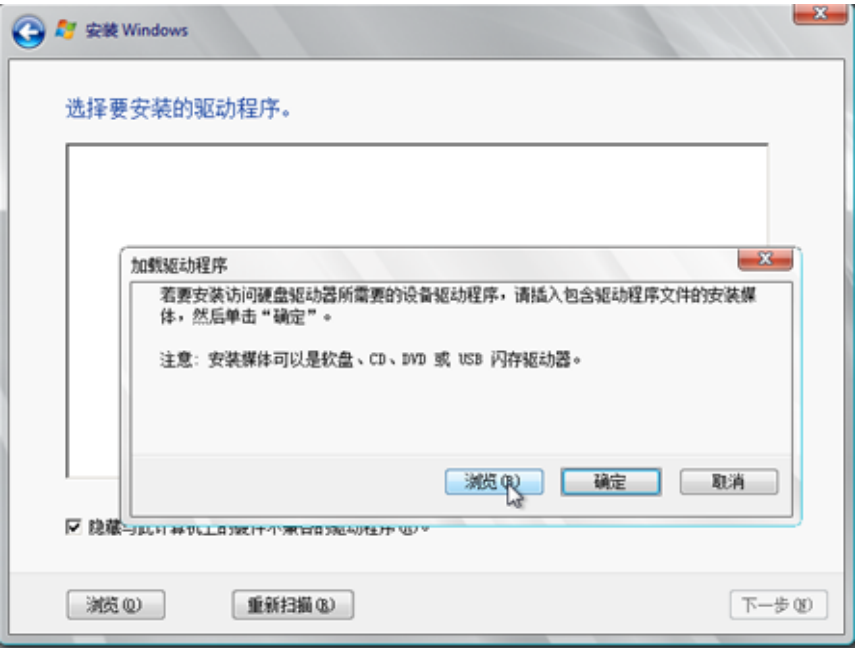
- 如果系统显示磁盘，如[图4-8](#)所示，则执行[步骤1.9](#)。

图 4-8 安装位置



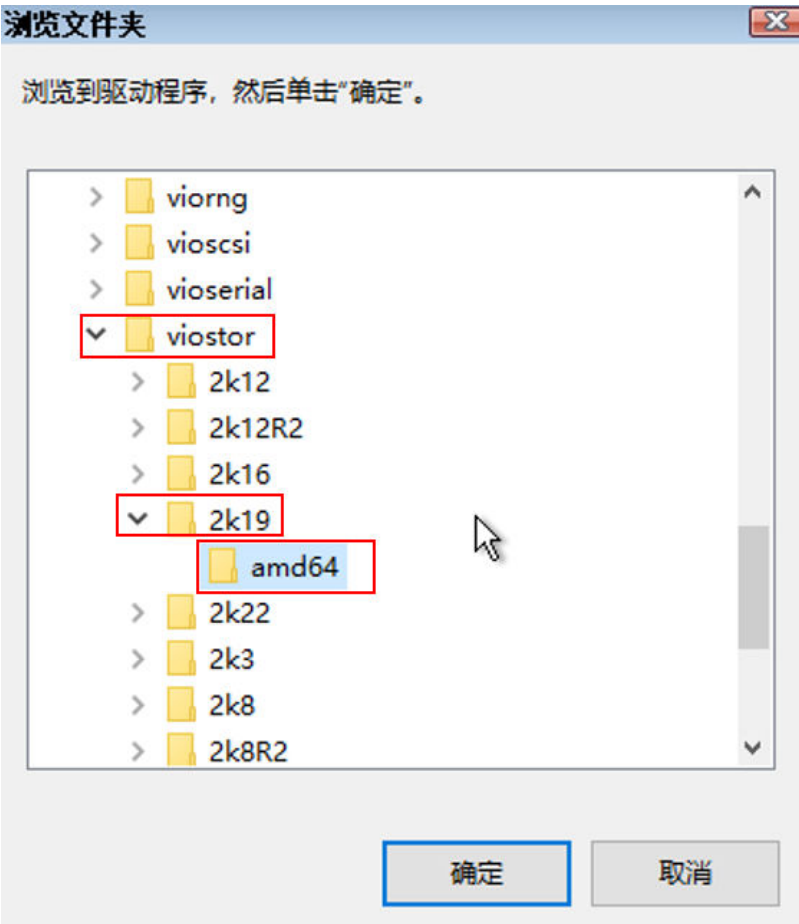
7. 依次单击“加载驱动程序 > 浏览”。

图 4-9 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 4-10 选择 viostor 驱动文件夹



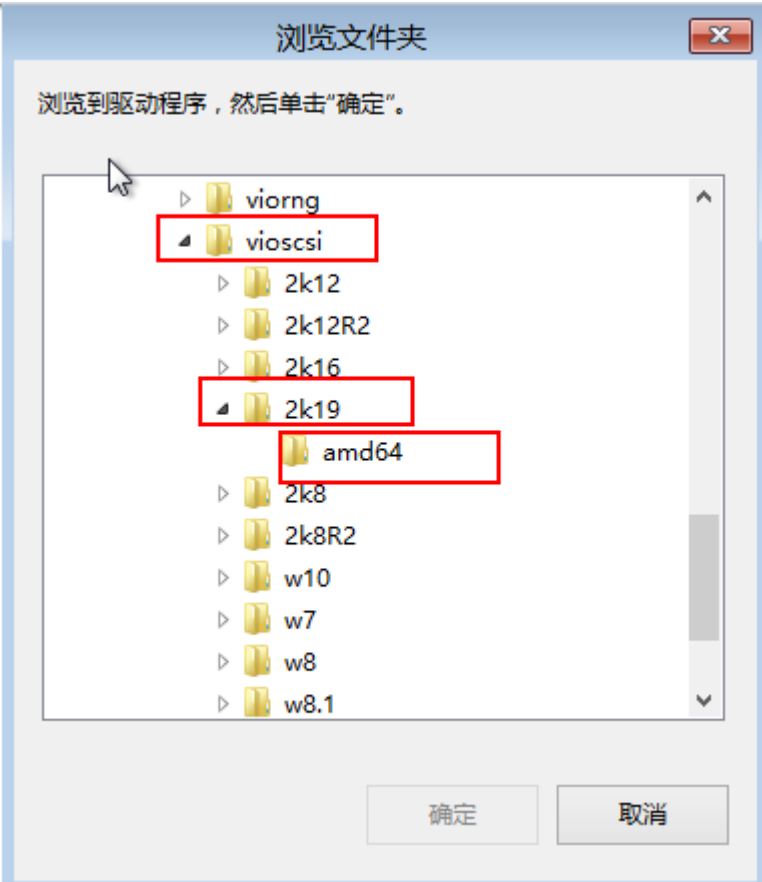
选择viostor.inf驱动并单击“下一步”。

图 4-11 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 4-12 选择 vioscsi 驱动文件夹



选择vioscsi.inf驱动并单击“下一步”。

图 4-13 选择要安装的驱动程序 vioscsi.inf



9. 选择磁盘，并单击“下一步”。

图 4-14 安装位置



说明

如果磁盘类型为“脱机”状态，如图4-15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 4-15 磁盘脱机



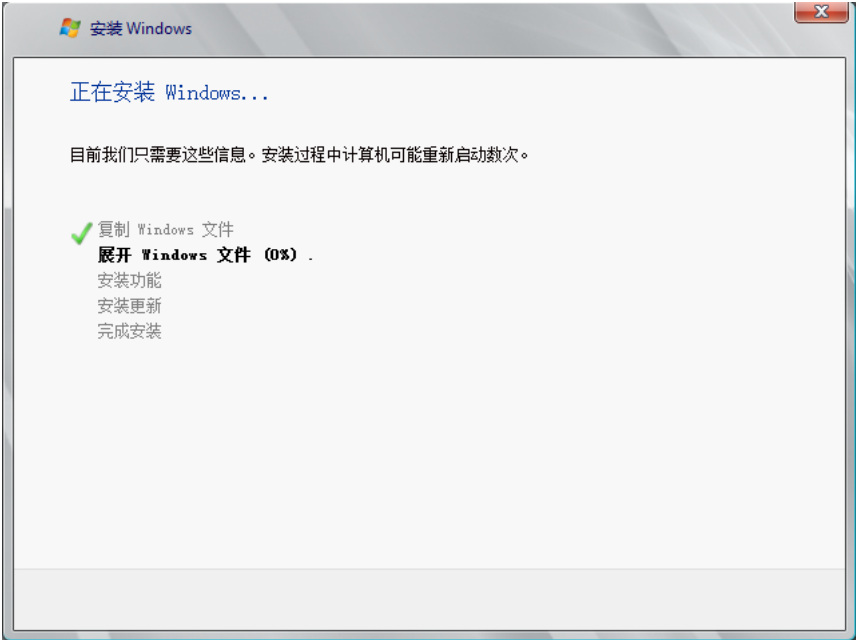
10. 进入“正在安装Windows”界面，开始安装操作系统，如图4-16所示。  
安装过程大约耗时50分钟，过程中云服务器会自动重启，请重新登录弹性云服务器并按照提示完成操作系统的后续配置。



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：!@\$%^\_-=+[{ }],./?

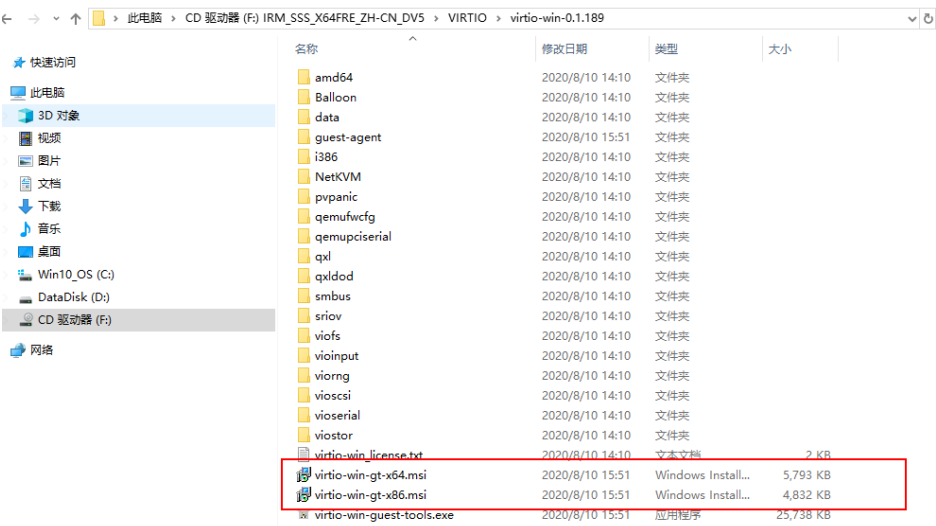
图 4-16 安装进度



步骤2 安装相关驱动。

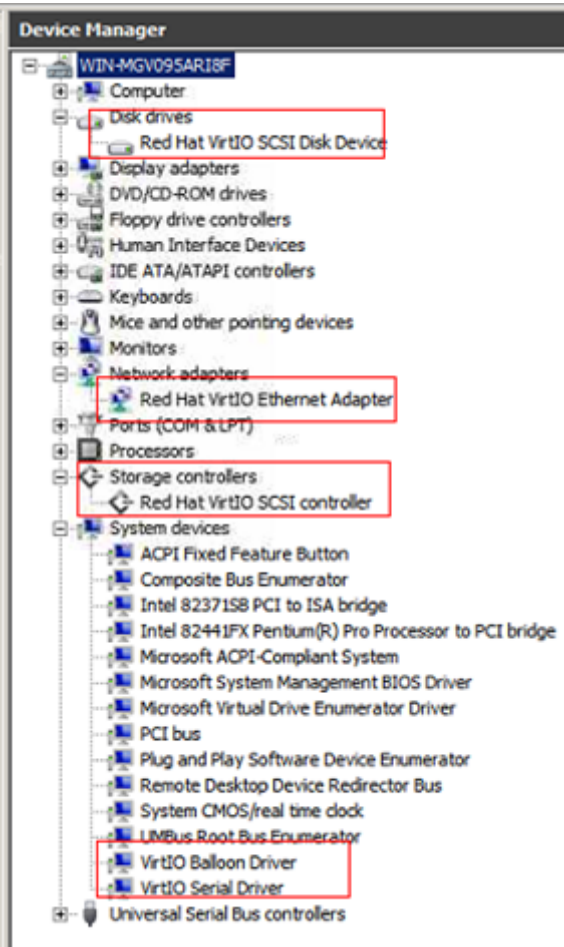
1. 打开“计算机”，并双击驱动器，如图4-17所示。

图 4-17 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 4-18 查看设备管理器



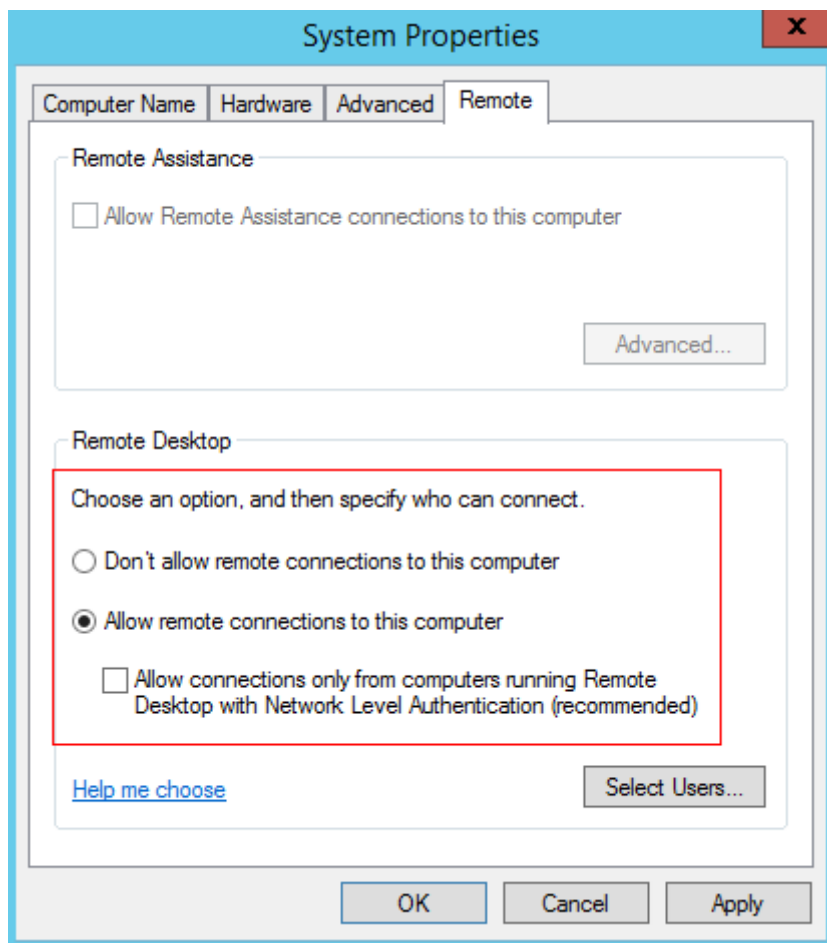
----结束

## 4.6 虚拟机环境配置

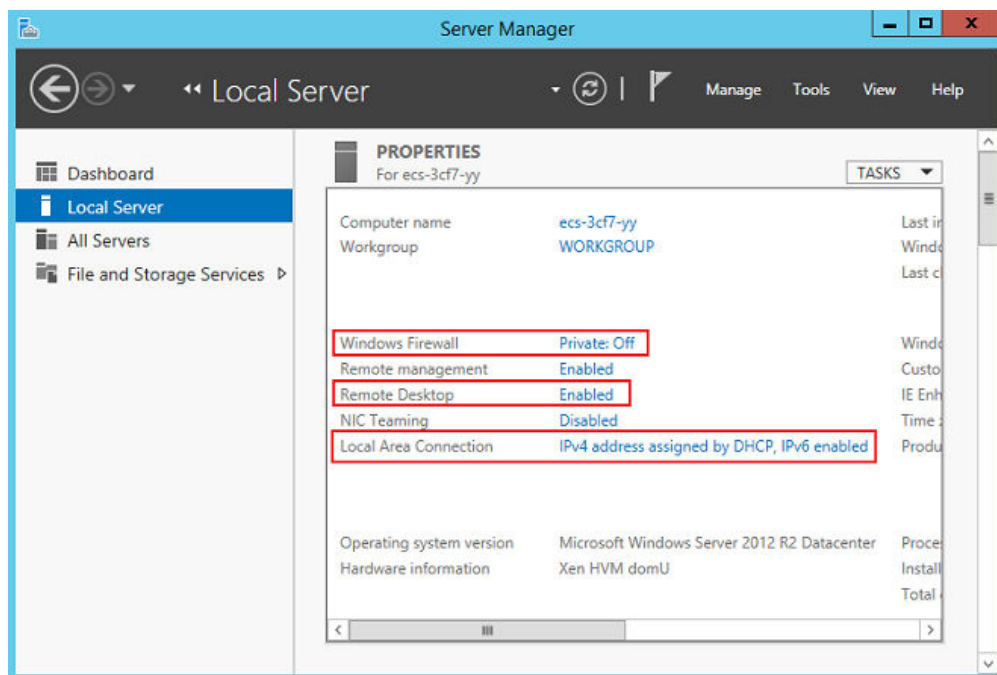
虚拟机Windows操作系统安装完成后，需要关闭防火墙，并配置远程桌面启动。

### 操作步骤

1. 单击左下角的图标，选择“服务器管理”，进入“服务器管理”页面。
2. 在左侧导航树中选择“本地服务器”，在右侧“Windows防火墙”区域，单击其后的状态链接。
3. 在“Windows防火墙”页面，选择左侧导航树中的“开启或关闭Windows防火墙”。
4. 在“自定义设置”页面，将专用网络和公用网络的Windows防火墙设置为关闭。
5. 防火墙设置完成后，返回“本地服务器”页面，单击“远程桌面”后的状态链接。
6. 在“系统属性”页面，选择“允许远程连接到此计算机”。



7. 返回“本地服务器”页面，确保各属性状态如下图所示。



## 4.7 安装 Cloudbase-Init

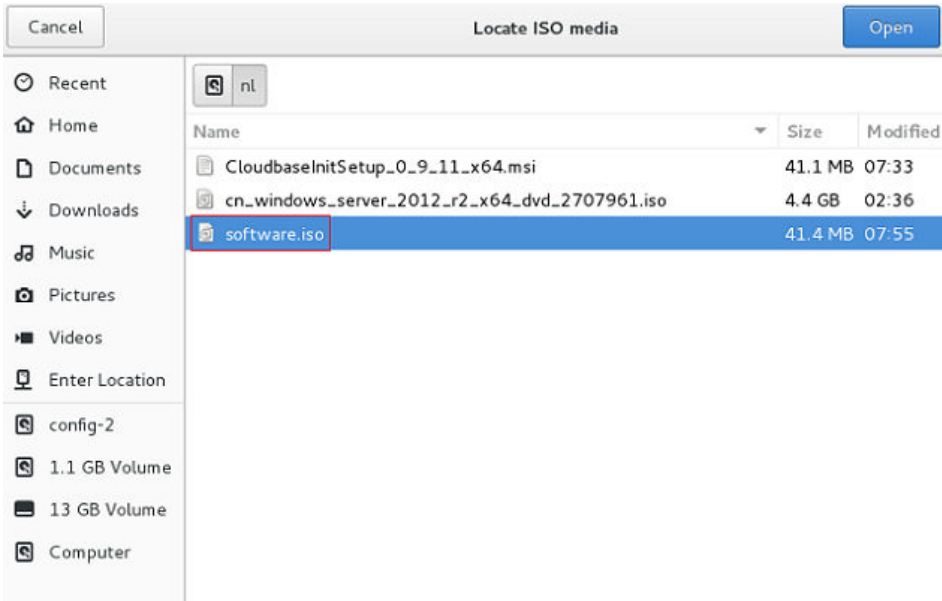
1. 参考[硬件和软件准备](#)章节下载Cloudbase-Init安装包及SHA256校验码并完成完整性校验，详情请参考[如何对软件进行完整性校验？](#)。
2. 将Cloudbase-Init的安装包上传至宿主机，制作成ISO文件。  
先将安装包上传至宿主机。

```
[root@server nl]# ll
total 4390172
-rw-r--r--. 1 root root 41070592 Sep 26 07:33 CloudbaseInitSetup_x64.msi
-rw-r--r--. 1 qemu qemu 4413020160 Sep 26 02:36 cn_windows_server_2012_r2_x64_dvd_2707961.iso
```

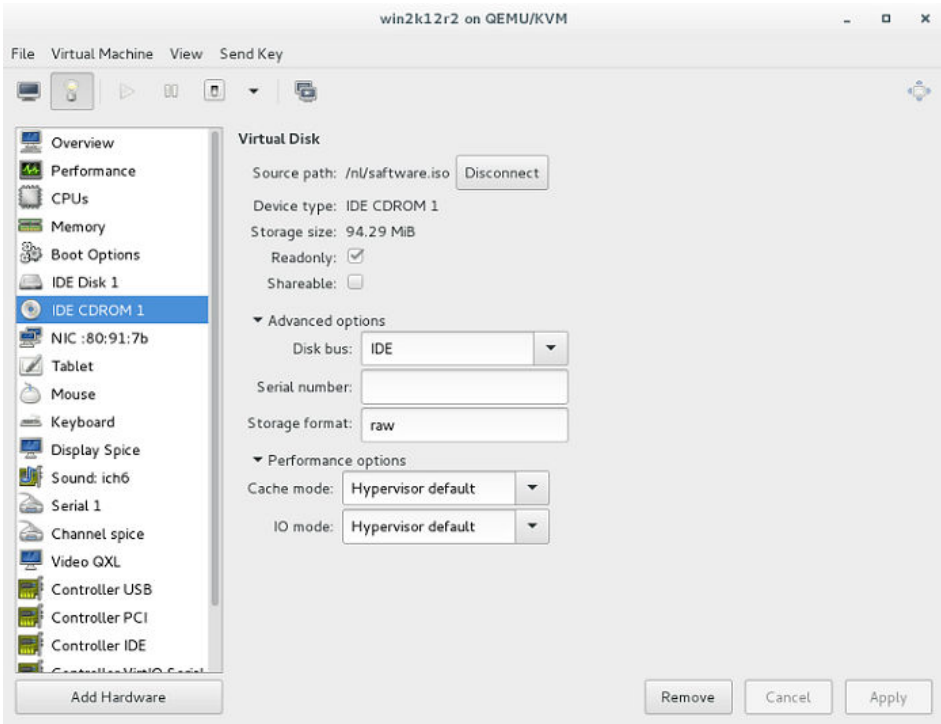
在安装包所在目录执行命令：

```
mkisofs -L -R -J -T -V system-sp2 -o software.iso
CloudbaseInitSetup_x64.msi
```

3. 在虚拟机中挂载制作完成的ISO文件。具体操作如下：
- a. 在virt-manager页面，选择“View > Details”。
- b. 在左侧导航树中选择“IDE CDROM 1”，单击右侧的“Disconnect”。
- c. 再单击“Connect”。
- d. 选择本地的ISO文件，单击右上角的“Open”。



- e. 检查无误后单击“Apply”。



4.   Cloudbase-Init安装。
- 在光驱中找到Cloudbase-Init软件包，双击进行安装。Cloudbase-Init的默认安装路径为：
- C:\Program Files\Cloudbase Solutions\Cloudbase-Init
5.   编辑Cloudbase-Init配置文件“cloudbase-init.conf”。
- 使用文本编辑工具（如Notepad），打开“C:\Program Files\Cloudbase Solutions\Cloudbase-Init\conf”文件，按照以下要求进行配置：
- a.   创建用户Administrator，并加入组，修改如下参数的取值：  
username=Administrator  
groups=Administrators

b.   设置hostname，修改如下参数（若不存在则需要添加）：  
netbios\_host\_name\_compatibility=false

c.   查找并修改logging\_serial\_port\_settings参数，修改端口设置如下：  
logging\_serial\_port\_settings=COM1,115200,N,8

d.   添加metadata\_services参数，设置加载服务如下：  
metadata\_services=cloudbaseinit.metadata.services.httpservice.HttpService,cloudbaseinit.metadat  
a.services.configdrive.ConfigDriveService

e.   添加plugins参数，设置要加载的模块，以“,”为分隔符，加粗的部分为模块关键字。
- 其中必选加载模块，请参考如下配置（**必选**）：  
plugins=cloudbaseinit.plugins.common.localscripts.**LocalScriptsPlugin**,cloudbaseinit.plugins  
.common.mtu.**MTUPlugin**,cloudbaseinit.plugins.windows.createuser.**CreateUserPlugin**,clou  
dbaseinit.plugins.common.setuserpassword.**SetUserPasswordPlugin**,cloudbaseinit.plugins.c  
ommon.sshpublickeys.**SetUserSSHPublicKeysPlugin**,cloudbaseinit.plugins.common.sethost  
name.**SetHostNamePlugin**,cloudbaseinit.plugins.windows.extendvolumes.**ExtendVolumes  
Plugin**,cloudbaseinit.plugins.common.userdata.**UserDataPlugin**,cloudbaseinit.plugins.wind  
ows.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
- 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。

4.8 安装 bms-network-config 软件包 (可选)

操作背景

安装network-config软件包, 与Cloudbase-Init配合完成裸金属服务器的网络配置。

前提条件

- 已登录虚拟机。
- 已安装好Cloudbase-Init软件。

操作步骤

1. 参考[硬件和软件准备](#)章节下载对应版本的bms-network-config软件包及SHA256校验码并完成完整性校验, 详情请参考[如何对软件进行完整性校验?](#)。
解压后将“bms-network-config.conf”放置于“C:\Program Files\Cloudbase Solutions\Cloudbase-Init\LocalScripts\”目录下。

说明

- “bms-network-config.conf”配置文件中的bsdtar_path默认设置为“C:\Program Files\Cloudbase Solutions\Cloudbase-Init\bin\bsdtar.exe”。若Cloudbase-Init安装到非默认路径下, 请根据实际情况配置成Cloudbase-Init的bsdtar.exe所在目录。
2. 使用文本编辑工具 (如Notepad) 打开“C:\Program Files\Cloudbase Solutions\Cloudbase-Init\conf”目录下的cloudbase-init配置文件, 查看其中的

local_scripts_path参数配置的路径，cloudbase-init会从这个路径去执行需要执行的脚本。

```
[DEFAULT]
username=Admin
groups=Administrators
inject_user_password=true
config_drive_raw_hhd=true
config_drive_cdrom=true
config_drive_vfat=true
bsdtar_path=C:\Program Files\Cloudbase Solutions\Cloudbase-Init\bin\bsdtar.exe
mtools_path=C:\Program Files\Cloudbase Solutions\Cloudbase-Init\bin\
verbose=true
debug=true
logdir=C:\Program Files\Cloudbase Solutions\Cloudbase-Init\log\
logfile=cloudbase-init.log
default_log_levels=comtypes=INFO,suds=INFO,iso8601=WARN,requests=WARN
logging_serial_port_settings=
mtu_use_dhcp_config=true
ntp_use_dhcp_config=true
local_scripts_path=C:\Program Files\Cloudbase Solutions\Cloudbase-Init\LocalScripts\
```

3. 根据cloudbase-init配置文件中的local_scripts_path参数，再将bms-network-config.py放置于“C:\Program Files\Cloudbase Solutions\Cloudbase-Init\LocalScripts”目录下。

推荐使用cloudbase-init的默认路径，若将脚本文件放置于其他路径下，请同步修改cloudbase-init的配置文件中的local_scripts_path参数。

4.9 安装 SDI 卡驱动（可选）

镜像中安装SDI卡前端驱动插件，使裸金属服务器能够支持挂载云硬盘，并且支持从云硬盘启动，实现快速发放能力。如果确认不需要使用云硬盘或裸金属服务器硬件不具备SDI卡，此项为可选。

前提条件

- 已登录虚拟机。
- 已参考[硬件和软件准备](#)章节获取SDI卡的驱动程序（scsi_ep_front），并且已上传至虚拟机。

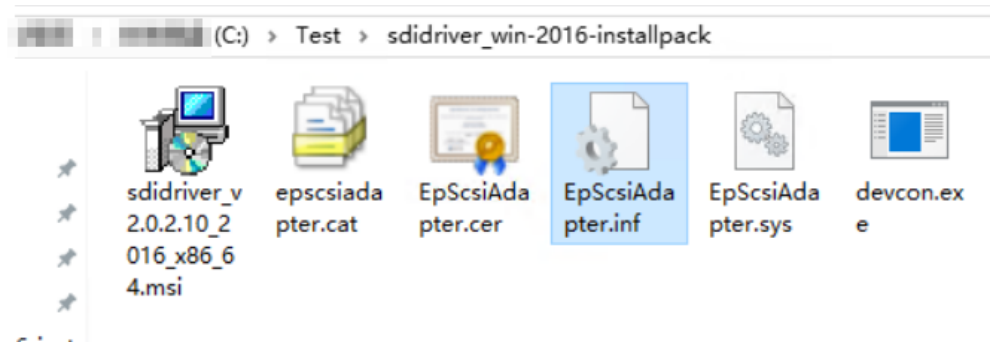
📖 说明

scsi_ep_front驱动包的版本必须为2.0.2.12及以上。

- 已安装WDK。
WDK下载链接：<http://download.microsoft.com/download/4/E/0/4E07EAAD-E394-4EA8-B2B8-D46E46A409C5/wdk/wdksetup.exe>

操作步骤

1. 从WDK目录“C:\Program Files (x86)\Windows Kits\10\Tools\x64”下获取设备管理工具devcon.exe，将工具devcon.exe和SDI驱动放在同一个目录下。



2. 找到EpScsiAdapter安装信息文件，查看PCI的ID。

```
SourceDisksNames]
=%DiskName%, , , ""

SourceDisksFiles]
源驱动文件磁盘位置
pScsiAdapter.sys=1

DiskCopyfiles]
pScsiAdapter.sys

Manufacturer]
ManufacturerName%=SD100, NTamd64

SD100, NTamd64]
驱动安装入口和设备ID
DevDesc%=EpScsiAdapterInstall.NT, PCI\VEN_19E5&DEV_1610&SUBSYS_000119E5

EpScsiAdapterInstall.NT]
copyFiles=DiskCopyfiles

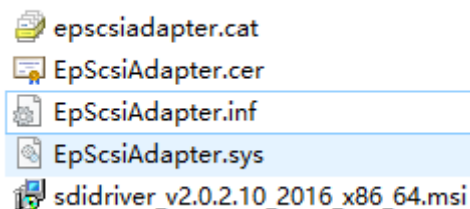
注册表中服务项的路径为HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services
EpScsiAdapterInstall.NT.Services]
ddservice=EpScsiAdapterSvc, 0x00000002, SysAddService

服务的具体选项
SysAddService]
```

3. 打开命令提示符，进入SDI驱动和安装工具所在目录，执行如下命令。

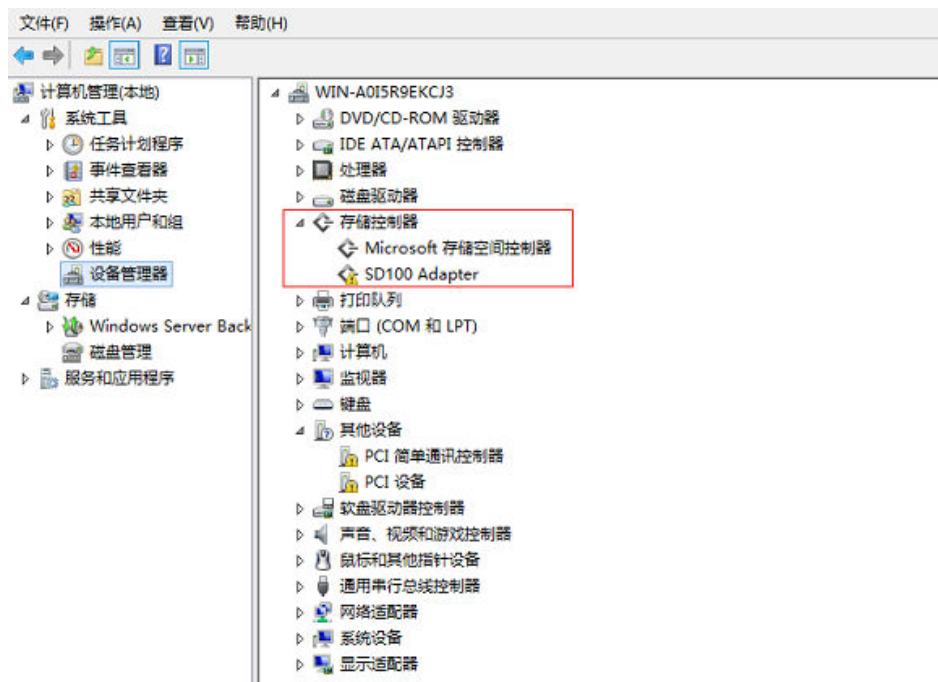
```
devcon install EpScsiAdapter.inf "PCI\VEN_19E5&DEV_1610&SUBSYS_000119E5"
```

名称



其中，PCI的ID需要根据实际情况而定。

4. （可选）删除设备节点。
 - a. 若制作镜像的环境没有SDI卡硬件支持，在安装完SDI驱动后，会发现设备管理器中的存储控制器存在异常。



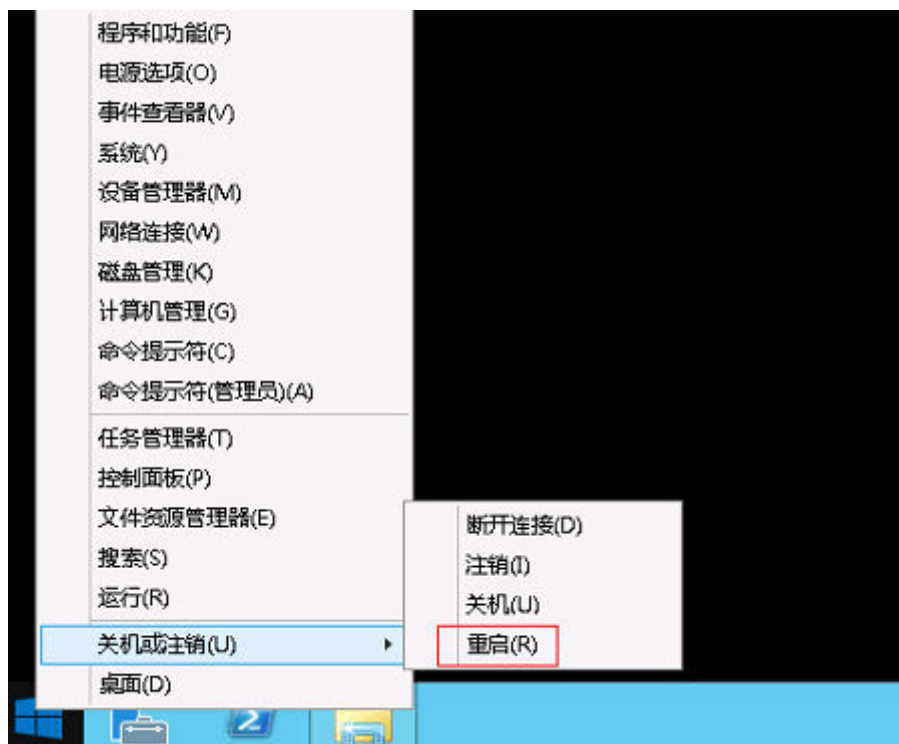
- b. 打开命令行，进入存放SDI驱动和devcon安装工具的目录下，删除异常的设备节点。

执行命令：**devcon.exe find "PCI\VEN_19E5*"**，查找到异常的设备节点。

执行命令：**devcon.exe remove "@ROOT\SCSIADAPTER\0000"**，删除异常节点。

```
C:\Users\Administrator>cd /
C:\>cd SDI-0319
C:\SDI-0319>
C:\SDI-0319>devcon.exe find "PCI\VEN_19E5*"
ROOT\SCSIADAPTER\0000 : SD100 Adapter
1 matching device(s) found.
C:\SDI-0319>devcon.exe remove "@ROOT\SCSIADAPTER\0000"
ROOT\SCSIADAPTER\0000 : Removed
1 device(s) were removed.
C:\SDI-0319>
```

- c. 在系统内部重启操作系统。



d. 检测设备管理器中存储控制器的异常是否消失。

4.10 安装一键式重置密码插件（可选）

操作场景

为了保证使用镜像创建的裸金属服务器可以实现一键式密码重置功能，建议您在制作镜像时安装重置密码插件“CloudResetPwdAgent”。

需保证虚拟机C盘可写入，且剩余空间大于600MB。

安装方法

步骤1 请参考[获取并校验一键式重置密码插件完整性](#)，下载对应的一键式重置密码插件CloudResetPwdAgent.zip并完成完整性校验。

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。

步骤2 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。

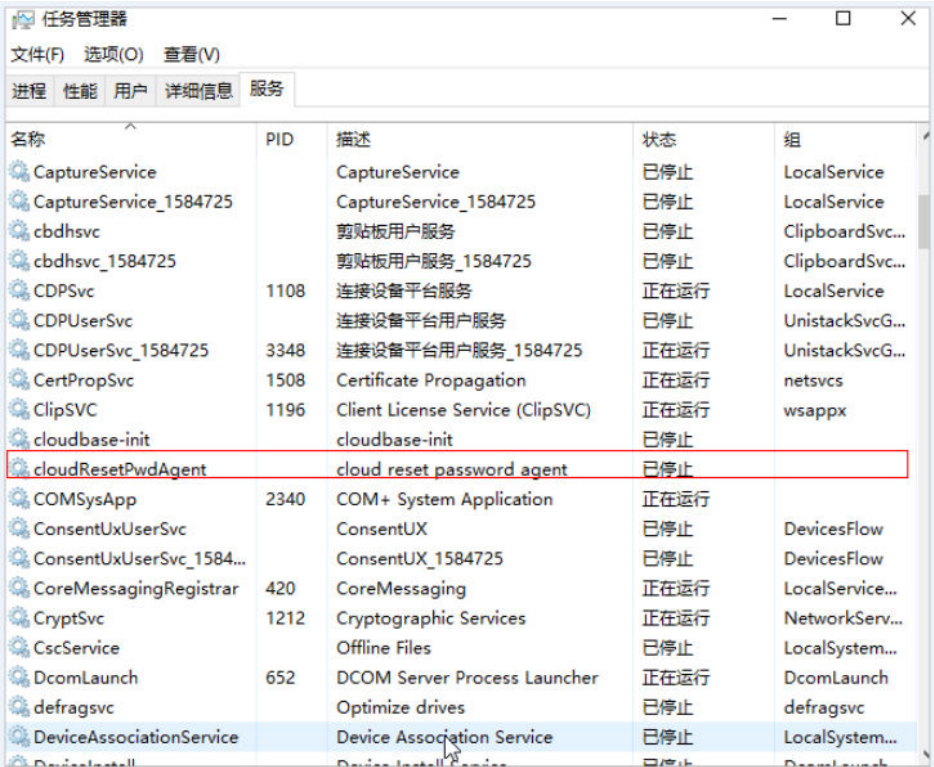
步骤3 安装一键式重置密码插件。

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4-19](#)所示，表示安装成功，否则安装失败。

图 4-19 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

卸载方法

- 1. 进入C:\CloudResetPwdAgent\bin文件夹。
- 2. 双击“UninstallApp-NT.bat”。
- 3. 删除C:\CloudResetPwdAgent的文件。
- 4. 删除C:\CloudResetPwdUpdateAgent的文件。

4.11 Windows 时区校准设置

Linux操作系统以主机板CMOS芯片的时间作为格林尼治标准时间，再根据设置的时区来确定系统的当前时间。但是一般Windows操作系统并不如此，Windows系统直接将CMOS时间认定为系统当前时间，不再根据时区进行转换，需要做如下设置：

- 1. 登录Windows虚拟机操作系统。
- 2. 单击左下角的  图标，选择“Windows PowerShell”，输入“regedit.exe”，并按“Enter”，打开注册表。
- 3. 在“注册表编辑器”页面，选择“HKEY_LOCAL_MACHINE > SYSTEM > CurrentControlSet > Control > TimeZoneInformation”。

4. 在“TimeZoneInformation”右侧区域右键单击，选择“新建 > DWORD (32-位) 值(D)”，添加一项类型为REG_DWORD的机码，名称为RealTimeIsUniversal，值设为1。

说明

若该款镜像不支持鼠标操作，右键单击可改为按键盘上的菜单键，然后用键盘进行操作。

4.12 Windows 虚拟内存设置

说明

仅Windows Server 2012 R2和Windows Server 2016需要设置虚拟内存。

裸金属服务器的内存很大，自动分配的虚拟内存会占用大量的系统盘空间，影响系统性能。建议在镜像制作过程中关闭虚拟内存或者设置上限。具体操作如下：

1. 登录Windows虚拟机操作系统。
2. 单击左下角的 图标，在“这台电脑/此电脑”上右键单击，选择“属性”。进入“系统”页面。
3. 在左侧导航中选择“高级系统设置”。弹出“系统属性”对话框，默认处于“高级”页签下。
4. 在“性能”区域，单击“设置”。弹出“性能选项”对话框。
5. 单击“高级”页签，在“虚拟内存”区域单击“更改”。弹出“虚拟内存”对话框。

图 4-20 “虚拟内存”对话框



- 6. 清除勾选“自动管理所有驱动器的分页文件大小”。
- 7. 如果需要关闭虚拟内存，请选择“无分页文件”；如果需要为虚拟内存设置上限，选择“自定义大小”，建议将虚拟内存最大值设置在2G以内。
- 8. 单击“设置”，在弹出的对话框中选择“是”。
- 9. 单击“确定”。
- 10. 重启操作系统完成虚拟内存设置。

4.13 Windows 自动更新配置（可选）

Windows Server自动更新是Windows的一项功能，当适用于您的计算机的重要更新发布时，它会提醒您下载和安装。而在默认情况下这个功能是需要手动配置的。

操作步骤

此处仅为参考，具体操作请以Windows官方提供的方式为准。

1. 登录Windows虚拟机操作系统。
2. 单击左下角的  图标，选择“服务器管理”，进入“服务器管理”页面。
3. 在左侧导航中选择“本地服务器”，在右侧区域找到“Windows 更新”，单击其后的“未配置”。
4. 在“Windows 更新”页面，单击“启用自动更新”。

说明

请用户根据实际需求选择是否启用自动更新。如果启用，在升级过程中需要耐心等待升级完成，切勿中断。

4.14 SID 配置

须知

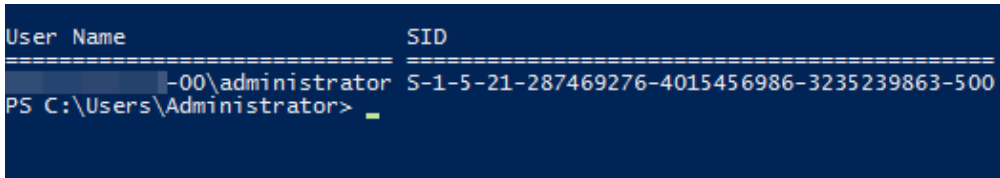
该步骤是虚拟机配置中的最后一个步骤，由于Cloudbase-init机制，在用于镜像制作的虚拟机关机后再开机会生成随机密码，导致该虚拟机无法登录，所以请您再检查一遍虚拟机内的配置流程，确保正确后，再进行该步操作。

SID也就是安全标识符（Security Identifiers），是标识用户、组和计算机账户的唯一号码。

仅Windows Server 2016需要在镜像制作时配置SID，Windows Server 2012 R2的SID配置请参考《[裸金属服务器常见问题](#)》。

1. 登录Windows虚拟机操作系统。
2. 单击左下角的  图标，选择“Windows PowerShell”，输入 **whoami /user** 命令查看机器的SID值。

图 4-21 查看 SID 初始值



```
User Name                               SID
-----
-00\administrator S-1-5-21-287469276-4015456986-3235239863-500
PS C:\Users\Administrator>
```

3. 修改Cloudbase-init的配置文件。
 - a. 分别打开“cloudbase-init.conf”和“cloudbase-init-unattend.conf”文件。
文件所在目录：C:\Program Files\Cloudbase Solutions\Cloudbase-Init\conf
 - b. 为两个配置文件都增加一行“first_logon_behaviour=no”。
[DEFAULT]
username=Administrator
groups=Administrators
first_logon_behaviour=no


```
netbios_host_name_compatibility=false
metadata_services=cloudbaseinit.metadata.services.httpser
inject_user_password=true
...
```

- c. 删除“cloudbase-init-unattend.conf”配置文件中的“cloudbaseinit.plugins.common.sethostname.SetHostNamePlugin”。

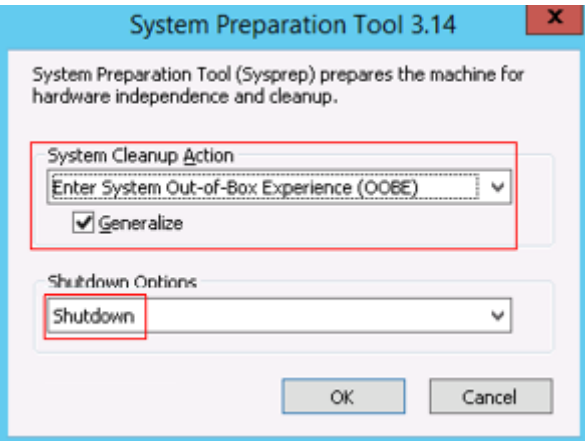
图 4-22 配置文件修改示例

```
se Solutions\Cloudbase-Init\log\
.log
0, suds=INFO, iso8601=WARN, requests=WARN
M1, 115200, N, 8

iles\Cloudbase Solutions\Cloudbase-Init\LocalScripts\
.metadata.services.configdrive.ConfigDriveService,cloudbaseinit.metadata.services.httpservi
common.mtu.MTUPlugin,cloudbaseinit.plugins.common.sethostname.SetHostNamePlugin,cloudbaseini
```

4. 打开命令提示符，输入如下命令打开Sysprep窗口。
进入目录：C:\Program Files\Cloudbase Solutions\Cloudbase-Init\conf>
输入命令：C:\Windows\System32\Sysprep\sysprep.exe /unattend:Unattend.xml
5. 按照下图进行设置。

图 4-23 System Preparation Tool 设置



6. 清理完成会自动重启，然后系统需要重新封装，重新解包。进入系统后，按照2的方法再次查看SID值。

图 4-24 查看 SID 值（修改后）

```
User Name                               SID
=====
win-ck7r022vrh0\administrator S-1-5-21-3812874840-1741028955-636704118-500
PS C:\Users\Administrator>
```

可以看到，SID已被修改为新的值。

5 关闭虚拟机并获取镜像

关闭虚拟机

可以在虚拟机操作系统中执行**sudo poweroff**进行关机。如果无效则在virt-manager中关闭虚拟机，单击“Virtual Machine Manager”，选中正在运行的虚拟机，右键单击“Shut Down”即可关闭虚拟机。

如果不能正常关闭，建议单击“Force Off”进行强制关闭。

获取镜像文件

如果在创建镜像磁盘时选择自动创建，则宿主机“/var/lib/libvirt/images”目录下，为对应的qcow2或img镜像文件。

图 5-1 镜像文件（Linux）

```
File Edit View Search Terminal Help
[root@localhost images]# pwd
/var/lib/libvirt/images
[root@localhost images]# ll -h
total 15G
-rw-----. 1 qemu qemu 5.1G Jun 28 10:21 sles11-1.qcow2
-rw-----. 1 root root 5.1G Jun 28 05:59 sles11.qcow2
-rw-----. 1 qemu qemu 11G Jun 28 11:49 vm1-1.qcow2
-rw-----. 1 qemu qemu 8.1G Jun 28 11:50 vm1-2.qcow2
-rw-----. 1 qemu qemu 6.1G Jun 28 11:50 vm1.qcow2
[root@localhost images]#
```

如果选择手动创建镜像磁盘文件，则到相应的路径下获取镜像文件。

压缩镜像文件

如果生成的镜像文件过大，可以执行以下命令进行压缩：

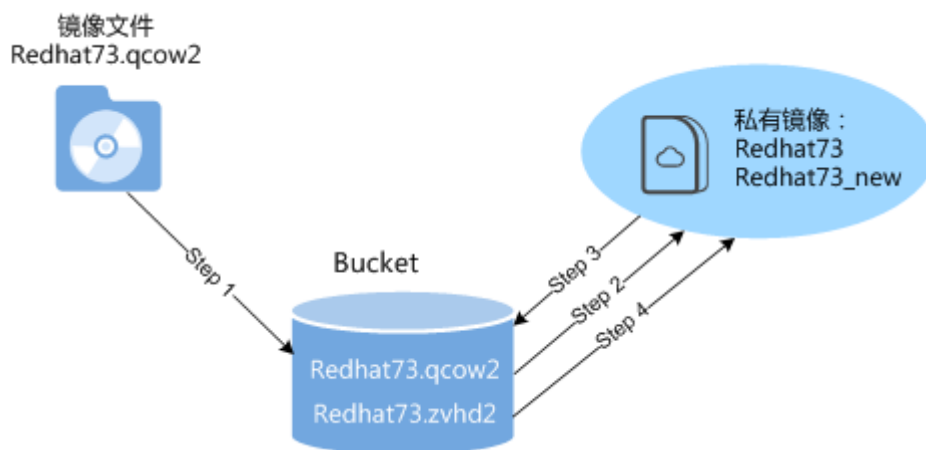
```
qemu-img convert -f raw -O qcow2 SourceFile FinallyFile.qcow2
```

其中，源镜像格式SourceFile可以为.img或.qcow2。

6 镜像格式转换

当前版本仅支持zvhd2格式的镜像，在获取镜像文件后，需要将镜像格式转换为zvhd2。

图 6-1 镜像格式转换流程



Step 1: 将制作完成的镜像文件（如Redhat73.qcow2）上传至OBS个人桶中。

Step 2: 将上传至OBS桶中的镜像文件注册为私有镜像（如Redhat73）。具体操作请参见《裸金属服务器用户指南》的“通过外部镜像文件创建私有镜像”章节。

Step 3: 导出私有镜像至OBS桶中（指定格式为zvhd2，如Redhat73.zvhd2）。具体操作请参见《镜像服务用户指南》的“导出镜像”章节。

Step 4: 将OBS桶中的zvhd2格式镜像注册为私有镜像（如Redhat73_new）。

说明

Step 3已将镜像格式转换为zvhd2格式，但是如果使用该私有镜像创建裸金属服务器，还需要完成Step 4的注册私有镜像操作。

7

FAQ

7.1 如何处理从 Linux 裸金属服务器的 bond0 的 vlan 子接口输出的报文，其源 mac 为 bond0 的 mac 问题？

上述问题为Linux操作系统已知的内核问题，会导致裸金属服务器的扩展网卡ping不通网关。[表7-1](#)中对应内核版本的操作系统均有上述内核缺陷。用户请根据使用的操作系统，例如Red Hat 7.3，向操作系统厂商获取对应的内核补丁进行修复。

表 7-1 操作系统类型及内核版本

| 操作系统类型 | 内核版本 |
|--------------|-------------|
| RedHat 7.2 | 3.10.0-327 |
| RedHat 7.3 | 3.10.0-514 |
| RedHat 7.4 | 3.10.0-693 |
| CenOS 7.0 | 3.10.0-514 |
| CentOS 7.2 | 3.10.0-327 |
| CentOS 7.3 | 3.10.0-514 |
| CentOS 7.4 | 3.10..0-693 |
| EulerOS 2.2 | 3.10.0-327 |
| Ubuntu 14.04 | 3.13.0-24 |

7.2 如何处理 SUSE 12 SP1 操作系统自带的 wicked 模块，其概率性将 bond 端口模式配置不正确，进而导致 IP 链路不通的问题？

上述问题为wicked模块已知问题，建议用户升级使用的wicked模块版本至0.6.40.28。请用户在wicked官方网站下载相应的软件包进行问题修复，wicked软件包的下载链接为：<https://software.opensuse.org/package/wicked>。

7.3 如何设置裸金属服务器 CPU 频率调节模式？

常见的CPU频率调节模式有performance、powersave、ondemand等，用户可以根据裸金属服务器的使用场景选择合适的CPU频率调节模式。

可以通过以下操作将CPU频率调节模式设置成performance：

- Red Hat 6系列操作系统
修改“/etc/sysconfig/cpuspeed”文件中的如下配置项：
GOVERNOR=performance
- Red Hat 7系列操作系统
修改“/etc/sysconfig/cpupower”文件中的如下配置项：
CPUPOWER_START_OPTS="frequency-set -g performance"
- Debian Gnu/Linux 8操作系统
修改“/etc/init.d/cpufrequtils”文件中的如下配置项：
GOVERNOR="performance"

📖 说明

- 若没有相关软件，可使用yum install xxx（xxx为软件名）安装，或者从官网下载对应rpm包进行安装。
- 对于Red Hat 6系列操作系统，如果要使intel_pstate驱动开机自动加载且使上述配置参数开机自动生效，可以配置开机执行脚本。
 1. 执行如下命令新建“intel_pstate.modules”文件。
vi /etc/sysconfig/modules/intel_pstate.modules
 2. 添加以下内容：
/sbin/modprobe intel_pstate > /dev/null 2>&1
 3. 按“Esc”，输入:wq!保存配置。
 4. 执行如下命令，更改文件权限：
chmod 755 /etc/sysconfig/modules/intel_pstate.modules

7.4 如何处理 Windows 裸金属服务器下发后 Cloudbase-init 服务异常退出的问题？

📖 说明

若无法登录和操作BMC页面，请联系华为工程师解决。

问题描述

Windows裸金属服务器下发成功后，如果Cloudbase-init异常退出，且服务器管理器中提示“无法找到来自Cloudbase-init的事件ID的描述，本地计算机上未安装引发此事件的组件，或安装已损坏，可以安装或修复本地计算机的组件”，在这种情况下，用户可以选择重新安装Cloudbase-init服务。

解决方案

1. 首先通过BMC登录裸金属服务器，备份Cloudbase-init服务安装路径“C:\Program Files\Cloudbase Solutions\Cloudbase-Init\”下的“conf”以及“LocalScripts”目录下的内容。
2. 从Cloudbase-init官网下载操作系统对应版本的Cloudbase-init软件包。
下载地址参考：<https://cloudbase.it/cloudbase-init/#download>。
3. 在BMC的主机管理页面，单击“远程控制”。
弹出远程控制窗口。
4. 在上方工具栏单击，选中“本地文件夹”，单击“浏览”。
弹出对话框。
5. 选择Cloudbase-init软件包，单击“连接”，将Cloudbase-init软件包上传至裸金属服务器中，通过进入Windows光驱获取软件包，进行软件重新安装。
6. 将步骤1中备份的“conf”和“LocalScripts”目录中的文件分别保存至重新安装的Cloudbase-init对应的路径下。
7. 重启裸金属服务器，查看Cloudbase-init是否已正常运行。

7.5 如何处理 cloud-init-local 概率性启动失败导致裸金属服务器不能正确注入数据的问题？

问题描述

对于Red Hat 7/CentOS 7系列操作系统，cloud-init-local可能会启动失败，且执行**systemctl status cloud-init-local.service**命令时，提示“OSError: [Errno 2] No such file or directory”，从而导致裸金属服务器不能正确注入数据。

解决方案

Red Hat 7/CentOS 7系列操作系统，如果自带libselinux模块版本低于2.5.7，存在已知内核缺陷（缺陷描述可参考：https://bugzilla.redhat.com/show_bug.cgi?id=1406520），该内核缺陷会概率性导致cloud-init-local启动失败。

建议在制作Red Hat 7/CentOS 7镜像时，参考“安装Cloud-lint > SUSE/Red Hat/CentOS/Oracle Linux/Ubuntu/Debian系列”章节中CentOS操作系统配置yum源后升级libselinux版本至2.5.7或者更高，来修复该问题。

7.6 如何激活 Windows 裸金属服务器？

对于Windows系列操作系统，目前需要手动激活。

1. 登录Windows裸金属服务器操作系统。
2. 单击“开始”菜单，在“搜索程序和文件”中输入“cmd”，并按“Enter”，打开命令提示符。
3. 执行以下命令，配置KMS服务器地址。
slmgr -skms 100.125.1.2
4. 执行以下命令激活裸金属服务器。
slmgr -ato
 - 如果出现类似“成功地激活了产品”的提示信息，表示激活成功，任务结束。
 - 如果出现错误：0xC004F074 软件授权服务器报告无法激活该裸金属服务器。密钥管理服务（KMS）不可用，说明无法激活，需要执行步骤5。
5. 查看裸金属服务器时间与标准时间是否一致，时间相差较大会出现无法激活的情况，将其设置为一致。
6. 执行以下命令，检查裸金属服务器到KMS服务器端口是否可达。
telnet 100.125.1.2 1688

如果无法连接，说明裸金属服务器内部防火墙没有放通1688端口，需要关闭或者放通防火墙TCP 1688端口。如果有安全狗之类的安全软件也请暂时停止使用。
7. 执行以下命令，重试裸金属服务器是否激活。
slmgr -ato

7.7 x86 EulerOS 2.3 UDP 报文性能调优

操作场景

用iperf3工具测试网络丢包最高带宽（UDP 1440字节）和最大pps（UDP 64字节），使用如下指导可以在UDP报文场景下提高网络带宽和pps指标。

操作步骤

1. SSH登录x86 EulerOS 2.3操作系统，并切换到root用户。
2. 备份“/etc/sysctl.conf”。
cp /etc/sysctl.conf /root/sysctl.conf_bak
3. 使用vim编辑“/etc/sysctl.conf”，在文件末尾添加如下内容并保存退出。
net.core.wmem_default=21299200
net.core.rmem_default=21299200
4. 执行以下命令，使配置生效。
/sbin/sysctl -p

7.8 如何对软件进行完整性校验？

前提条件

- 已创建裸金属服务器。
- 已获取需要校验完整性的软件及SHA256校验码。

校验软件完整性 (Linux)

1. 以root用户登录裸金属服务器。
2. 执行以下命令，获取软件的Hash值。
sha256sum {软件包本地目录}/{软件包名}
{软件包本地目录}：请根据实际下载目录进行替换。
{软件包名}：请根据实际下载软件包名进行替换。例如：qemu-img-hw.zip
3. 将[前提条件](#)中获取的SHA256 Hash值和步骤2获取的Hash值进行比较。
 - 一致：通过校验。
 - 不一致：请重新下载对应版本的软件，重复步骤2~步骤3进行校验。

校验软件完整性 (Windows)

1. 登录裸金属服务器。
2. 以管理员权限打开cmd，执行以下命令，获取软件的Hash值。
certutil -hashfile {软件包本地目录}\{软件包名} SHA256
{软件包本地目录}：请根据实际下载目录进行替换。
{软件包名}：请根据实际下载软件包名进行替换。例如：qemu-img-hw.zip
3. 将[前提条件](#)中获取的SHA256 Hash值和步骤2获取的Hash值进行比较。
 - 一致：通过校验。
 - 不一致：请重新下载对应版本的软件，重复步骤2~步骤3进行校验。

7.9 如何检查 device 是否正常运行

1. 以root用户登录运行环境，查询软件包安装路径。
cat /etc/ascend_install.info
系统显示示例如下：
Driver_Install_Path_Param=/usr/local/Ascend
2. 进入Driver安装路径，通过upgrade-tool工具查看下device侧运行文件系统版本。
cd /usr/local/Ascend/driver/tools/
./upgrade-tool --device_index -1 --system_version
如果能正常查询，则说明device侧已经正常启动，正常查询返回信息类似如下。

```
[root@localhost tools]# ./usr/local/Ascend/driver/tools/upgrade-tool --device_index -1 --system_version
{
  Get system version (device_id:0, "version") succeed, deviceId(0)
  {"device_id":0, "version": "1.0.0"}
  Get system version (device_id:1, "version") succeed, deviceId(1)
  {"device_id":1, "version": "1.0.0"}
}
```

A 附录

A.1 配置 SSH 服务

1. 安装宿主机的Linux操作系统后，打开终端terminal，输入以下命令，查看SSH服务状态。

service sshd status

查看回显信息，确认是否自启动（enabled），并且是否处于开启状态（active）。

- 如果没有自启动，执行命令：**service enable sshd**
- 如果没有处于开启状态，执行命令：**service sshd start**

2. 配置SSH服务的配置文件，使root用户能够使用SSH方式登录。

文件位置：/etc/ssh/sshd_config

设置PermitRootLogin yes

3. 配置完成后重启SSH服务。

service sshd restart

A.2 配置 vsftpd 服务

安装宿主机的Linux操作系统后，打开终端terminal，输入以下命令，查看vsftpd服务状态。

service vsftpd status

观察是否自启动（enabled），并且是否处于开启状态（active）。

- 如果没有自启动，执行命令：**service enable vsftpd**
- 如果没有处于开启状态，执行命令：**service vsftpd start**

A.3 配置 VNC 服务

1. 确认是否已安装VNC服务器。执行以下命令：

rpm -qa | grep tigervnc-server

回显如下信息：

```
tigervnc-server-1.1.0-5.e16.x86_64
```

- 如果没有安装，则不回显信息或显示没有安装，需要将软件安装到宿主机中。执行2。
- 如果已安装，执行3。

2. （可选）安装tigervnc-server。

- a. 执行**lsblk**命令（前提是ISO操作系统文件依然挂载在BMC的虚拟光驱中，如果没有需要重新挂载，和安装操作系统时的方式一致）。

 说明

若无法登录和操作BMC页面，请联系华为工程师解决。

- b. 找到name为sr0，type为rom的块设备，观察挂载点是否有文件目录。如果没有，执行以下命令：

```
mount /dev/sr0 /mnt
```

- c. 进入配置文件所在目录“/etc/yum.repos.d”，将其他后缀名为.repo的文件进行备份，然后**vim**新建一个repo文件，例如tiger.repo。文件内容如下：

```
[rhel-local]
baseurl=file:///mnt
enabled=1
gpgcheck=0
```

保存该文件，并执行以下命令：

```
yum repolist
```

```
yum install tigervnc-server
```

3. 启动VNC服务。执行以下命令：

```
vncserver
```

根据回显提示设置root用户的VNC登录密码。

回显信息中出现类似这样的一条信息：Log file is /root/.vnc/rhel:1.log。其中，“1”表示当前用户分配的是VNC的第一个虚拟桌面。

4. 查看进程。执行以下命令可以查看Xvnc进程是否存在：

```
ps -ef | grep Xvnc
```

```
[root@SZX1000318596 yum.repos.d]# ps -ef | grep Xvnc
root      3494      1   0 May24 ?        01:54:02 /usr/bin/Xvnc :2 -desktop SZX1000318596:2 (root) -auth /root
/.Xauthority -geometry 1920x1080 -rfbwait 30000 -rfbauth /root/.vnc/passwd -rfbport 5902 -fp catalogue:/etc/X1
1/fontpath.d -pn
root      36069     1   0 15:49 pts/3    00:00:00 /usr/bin/Xvnc :1 -desktop SZX1000318596:1 (root) -auth /root
/.Xauthority -geometry 1024x768 -rfbwait 30000 -rfbauth /root/.vnc/passwd -rfbport 5901 -fp catalogue:/etc/X11
/fontpath.d -pn
root      37017     1   0 15:51 pts/3    00:00:00 /usr/bin/Xvnc :5 -desktop SZX1000318596:5 (root) -auth /root
/.Xauthority -geometry 1024x768 -rfbwait 30000 -rfbauth /root/.vnc/passwd -rfbport 5905 -fp catalogue:/etc/X11
/fontpath.d -pn
root      43853 119671   0 16:36 pts/3    00:00:00 grep --color=auto Xvnc
You have new mail in /var/spool/mail/root
[root@SZX1000318596 yum.repos.d]#
```

可以看到进程号为36069，使用的端口号为5901，虚拟桌面号码为1。

5. 查看当前用户的VNC虚拟桌面。执行以下命令：

```
vncserver -list
```

```
[root@SZX1000318596 yum.repos.d]# vncserver -list

TigerVNC server sessions:

X DISPLAY #      PROCESS ID
:1           36069
:5           37017
:2           3494
[root@SZX1000318596 yum.repos.d]#
```

当前用户有三个虚拟桌面，分别是1、5、2。

6. 在本地Windows上使用VNC Viewer连接VNC服务器，登录宿主机。
首次安装VNC Viewer时，如果VNC Viewer为5.3.2版本，请设置“ColorLevel”为“rgb222”，设置位置为：Options > Expert > ColorLevel。
7. 为当前用户增加虚拟桌面。执行以下命令：

```
vncserver :6

执行vncserver -list命令查看是否添加成功，如下图所示，添加号码为6的虚拟桌面成功。

TigerVNC server sessions:

X DISPLAY #      PROCESS ID
:1           36069
:5           37017
:2           3494
:6           45675
You have new mail in /var/spool/mail/root
[root@SZX1000318596 yum.repos.d]#
```

A.4 裸金属服务器规格与驱动对应关系

说明

- 表中“-”表示不需要。
- 使用X86 V4版本（CPU采用Intel Broadwell架构）的服务器请参考[表A-2](#)。
 - 使用X86 V5版本（CPU采用Intel Skylake架构）的服务器请参考[表A-3](#)。
 - 使用X86 V6版本（CPU采用Intel Cascade Lake架构）的服务器请参考[表A-4](#)。
 - 使用鲲鹏CPU的服务器请参考[表A-5](#)。

表 A-1 裸金属服务器规格与驱动对应关系

| 裸金属服务器类型 | | 驱动种类 | | | | | | | |
|----------|------|-------|--------|------------------|---------------|------|-------|-------|--------------------|
| 裸金属服务器机型 | OS类型 | 服务器驱动 | SDI卡驱动 | Hi182 2标卡驱动/管理组件 | Hi1 822 卸载卡驱动 | IB驱动 | GPU驱动 | 多路径软件 | bms-network-config |

| 裸金属服务器类型 | | 驱动种类 | | | | | | | |
|----------|--|-------------|----|---|---|---|---|-----------|----|
| s4系列 | Windows | 需要安装v5服务器驱动 | 需要 | - | - | - | - | 带HBA的规格需要 | 需要 |
| | Cent OS/
Euler OS/
RedHat/
Oracle | 需要安装v5服务器驱动 | 需要 | - | - | - | - | 带HBA的规格需要 | 需要 |
| | Ubuntu | 需要安装v5服务器驱动 | 需要 | - | - | - | - | 带HBA的规格需要 | 需要 |
| | SLES | 需要安装v5服务器驱动 | 需要 | - | - | - | - | 带HBA的规格需要 | 需要 |
| | Debian | 需要安装v5服务器驱动 | 需要 | - | - | - | - | 带HBA的规格需要 | 需要 |
| d2/io2系列 | Windows | 需要安装v5服务器驱动 | - | - | - | - | - | - | 需要 |
| | Cent OS/
Euler OS/
RedHat/
Oracle | 需要安装v5服务器驱动 | - | - | - | - | - | - | 需要 |

| 裸金属服务器类型 | | 驱动种类 | | | | | | | |
|----------|--|-------------|----|----|---|----|----|---|----|
| | Ubuntu | 需要安装v5服务器驱动 | - | - | - | - | - | - | 需要 |
| | SLES | 需要安装v5服务器驱动 | - | - | - | - | - | - | 需要 |
| | Debian | 需要安装v5服务器驱动 | - | - | - | - | - | - | 需要 |
| h2系列 | Cent OS/
Euler OS/
RedH
at/
Oracle | 需要安装v5服务器驱动 | 需要 | - | - | 需要 | - | - | 需要 |
| | Ubuntu | 需要安装v5服务器驱动 | 需要 | - | - | 需要 | - | - | 需要 |
| p3系列 | Cent OS/
Euler OS/
RedH
at/
Oracle | 需要安装v5服务器驱动 | 需要 | 需要 | - | 需要 | 需要 | - | 需要 |
| | Ubuntu | 需要安装v5服务器驱动 | 需要 | 需要 | - | 需要 | 需要 | - | 需要 |

| 裸金属服务器类型 | | 驱动种类 | | | | | | | |
|-------------|---------------------------------|------------------|---|----|--------------------|---|---|---|--------------------|
| a/r/k系列（鲲鹏） | Cent OS/
Euler OS/
Ubuntu | 需要安装TaiShan服务器驱动 | - | 需要 | 带SDI卡的规格需要，其他规格不需要 | - | - | - | 带SDI卡的规格不需要，其他规格需要 |

表 A-2 裸金属服务器规格与驱动对应关系（V4）

| 裸金属服务器机型 | OS类型 | 服务器驱动 | SDI卡驱动 | Hi1822网卡驱动/管理组件 | IB驱动 | GPU驱动 | 多路径软件 | bms-network-config |
|----------|--|-------------|--------|-----------------|------|-------|-----------|--------------------|
| s4系列 | Windows | 需要安装v5服务器驱动 | 需要 | - | - | - | 带HBA的规格需要 | 需要 |
| | CentOS/
EulerOS/
RedHat/
Oracle | 需要安装v5服务器驱动 | 需要 | - | - | - | 带HBA的规格需要 | 需要 |
| | Ubuntu | 需要安装v5服务器驱动 | 需要 | - | - | - | 带HBA的规格需要 | 需要 |
| | SLES | 需要安装v5服务器驱动 | 需要 | - | - | - | 带HBA的规格需要 | 需要 |

| 裸金属服务器机型 | OS类型 | 服务器驱动 | SDI卡驱动 | Hi1822网卡驱动/管理组件 | IB驱动 | GPU驱动 | 多路径软件 | bms-network-config |
|----------|--------|-------------|--------|-----------------|------|-------|-----------|--------------------|
| | Debian | 需要安装v5服务器驱动 | 需要 | - | - | - | 带HBA的规格需要 | 需要 |

表 A-3 裸金属服务器规格与驱动对应关系（V5）

| 裸金属服务器机型 | OS类型 | 服务器驱动 | SDI卡驱动 | Hi1822网卡驱动/管理组件 | IB驱动 | GPU驱动 | 多路径软件 | bms-network-config |
|----------|------------------------------|-------------|--------|-----------------|------|-------|-----------|--------------------|
| s4系列 | Windows | 需要安装v5服务器驱动 | 需要 | - | - | - | 带HBA的规格需要 | 需要 |
| | CentOS/EulerOS/RedHat/Oracle | 需要安装v5服务器驱动 | 需要 | - | - | - | 带HBA的规格需要 | 需要 |
| | Ubuntu | 需要安装v5服务器驱动 | 需要 | - | - | - | 带HBA的规格需要 | 需要 |
| | SLES | 需要安装v5服务器驱动 | 需要 | - | - | - | 带HBA的规格需要 | 需要 |
| | Debian | 需要安装v5服务器驱动 | 需要 | - | - | - | 带HBA的规格需要 | 需要 |

表 A-4 裸金属服务器规格与驱动对应关系（V6）

| 裸金属服务器机型 | OS类型 | 服务器驱动 | SDI卡驱动 | Hi182 2网卡驱动/管理组件 | IB驱动 | GPU驱动 | 多路径软件 | bms-network-config |
|----------|--|-------------|--------|------------------|------|-------|-----------|--------------------|
| s4系列 | Windows | 需要安装v5服务器驱动 | 需要 | - | - | - | 带HBA的规格需要 | 需要 |
| | CentOS/
EulerOS/
RedHat/
Oracle | 需要安装v5服务器驱动 | 需要 | - | - | - | 带HBA的规格需要 | 需要 |
| | Ubuntu | 需要安装v5服务器驱动 | 需要 | - | - | - | 带HBA的规格需要 | 需要 |
| | SLES | 需要安装v5服务器驱动 | 需要 | - | - | - | 带HBA的规格需要 | 需要 |
| | Debian | 需要安装v5服务器驱动 | 需要 | - | - | - | 带HBA的规格需要 | 需要 |

表 A-5 裸金属服务器规格与驱动对应关系（鲲鹏）

| 裸金属服务器机型 | OS类型 | 服务器驱动 | Hi182 2标卡驱动/管理组件 | Hi182 2卸载卡驱动 | IB驱动 | GPU驱动 | 多路径软件 | bms-network-config |
|----------|-------------------------------|------------------|-------------------------|--------------------|------|-------|-----------|--------------------|
| a/r/k系列 | CentOS/
EulerOS/
Ubuntu | 需要安装TaiShan服务器驱动 | 带Hi182 2标卡的规格需要，其他规格不需要 | 带SDI卡的规格需要，其他规格不需要 | - | - | 带HBA的规格需要 | 带SDI卡的规格不需要，其他规格需要 |