鲲鹏 BoostKit 虚拟化使能套件

oVirt 部署指南

文档版本03发布日期2025-02-20





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

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 鲲鹏 oVirt 轻量级虚拟化管理平台 部署指南(openEuler 22.03)	1
1.1 简介	1
1.2 环境要求	2
1.3 环境配置	4
1.3.1 配置 BIOS	4
1.3.2 配置启动参数	4
1.4 安装和部署 ovirt-engine	5
1.4.1 配置部署环境	5
1.4.2 安装和部署 ovirt-engine 软件包	6
1.4.3 部署 ovirt-engine 服务	6
1.4.4 清理环境	10
1.5 安装和部署 ovirt-host	10
1.6 部署 oVirt 集群	11
1.7 创建虚拟机	14
1.8 配置 oVirt 虚拟机网络	17
2 鲲鹏 oVirt 轻量级虑拟化管理平台 部署指南(openFuler 20.03)	21
2.1 简介	21
2.1 简介	21 23
 2.1 简介	21 23 23
 2.1 简介 2.2 环境要求	21 23 23 24
 2.1 简介	21 23 23 24 29
 2.1 简介	21 23 23 24 29 30
 2.1 简介	21 23 23 24 29 30 30
 2.1 简介	21 23 24 24 29 30 30 31
 2.1 简介	21 23 24 24 29 30 31 35
2.1 简介	21 23 24 24 30 30 31 35 35
2.1 简介	21 23 24 24 30 30 31 35 35 37
2.1 简介	21 23 24 24 30 30 31 35 35 37 37
2.1 简介	21 23 23 24 29 30 30 30 31 35 35 35 37 37 41
2.1 简介	21 23 24 24 30 30 31 35 35 35 37 37 41 46
2.1 简介	21 23 23 24 29 30 30 30 31 35 35 35 37 37 41 46 46

鲲鹏 oVirt 轻量级虚拟化管理平台 部署指南 (openEuler 22.03)

1.1 简介

鲲鹏oVirt轻量级虚拟化软件(以下简称oVirt)是一个基于KVM的轻量级虚拟化软件,整合了许多优秀的开源软件,旨在替代VMware vsphere成为企业虚拟化的解决方案。它包括ovirt-engine、主机节点和存储节点三部分,支持本地存储和外部存储。本文主要介绍如何在使用openEuler 22.03操作系统的鲲鹏服务器上安装、部署ovirt-engine服务以及部署oVirt集群。

1.2 环境要求

在部署oVirt之前,需要确保使用环境均满足要求,包括硬件和软件配置。硬件配置包括CPU和网卡等。软件配置包括操作系统和应用程序等。

1.3 环境配置

1.4 安装和部署ovirt-engine

1.5 安装和部署ovirt-host

在计算节点安装和部署ovirt-host,需要修改"/usr/libexec/vdsm/hooks/ before_vm_start/50_hostedengine"文件。

1.6 部署oVirt集群

安装和部署ovirt-host完成后,需要部署oVirt集群以管理虚拟机。部署oVirt集群包括新 建主机、新建数据域和ISO域以及创建并配置虚拟机。

1.7 创建虚拟机

1.8 配置oVirt虚拟机网络

为保证集群中虚拟机之间的网络可以互通,需要配置oVirt虚拟机网络。在配置oVirt虚 拟机网络前,需要确保已完成BIOS配置和配置启动参数。

1.1 简介

鲲鹏oVirt轻量级虚拟化软件(以下简称oVirt)是一个基于KVM的轻量级虚拟化软件,整合了许多优秀的开源软件,旨在替代VMware vsphere成为企业虚拟化的解决方案。它包括ovirt-engine、主机节点和存储节点三部分,支持本地存储和外部存储。本文主要介绍如何在使用openEuler 22.03操作系统的鲲鹏服务器上安装、部署ovirt-engine服务以及部署oVirt集群。

oVirt是RedHat商业版本虚拟化软件RHEV的开源版本,它整合使用了libvirt、 gluster、patternfly、ansible等一系列优秀的开源软件,目前已经成为了企业虚拟化环 境可选的解决方案,相比OpenStack的庞大和复杂,oVirt在企业私有云建设的部署和 维护中具备了使用简单的优势。

oVirt部署架构如图1-1所示。





oVirt项目中的不同组件主要包含三个部分:

- ovirt-engine,用来进行管理虚拟机(创建、开关启停)、配置网络和存储等操作。
- 一个或多个主机(Host node),用来运行虚拟机。
 主机节点安装有VDSM和libvirt组件的Linux发行版,也包含一些用来实现网络虚拟化和其它系统服务的组件。
- 一个或多个存储节点(Local storage或Shared storage),用来存放虚拟机镜像和ISO镜像。

存储节点可以使用块存储或文件存储,可以利用主机节点自身的存储做存储节点 (local on host模式),或者使用外部的存储,例如通过NFS访问,或者是IP-SAN/FC-SAN。还有一种就是超融合架构,通过Gluster将主机节点自身的磁盘组 成池来使用,同时能够实现高可用和冗余。

关于oVirt的更多信息请访问oVirt官网。

1.2 环境要求

在部署oVirt之前,需要确保使用环境均满足要求,包括硬件和软件配置。硬件配置包括CPU和网卡等。软件配置包括操作系统和应用程序等。

文档版本 03 (2025-02-20)

组网环境



软硬件要求

部署oVirt需要满足如表1-1和表1-2所示的硬件配置要求和软件配置要求。

表 1-1 硬件配置要求

项目	管理节点/计算节点/存 储节点	计算节点
服务器	鲲鹏服务器	鲲鹏服务器
CPU	2*鲲鹏920处理器	2*鲲鹏920处理器

🛄 说明

本文档中计算节点和存储节点部署在同一台服务器上。

表 1-2 操作系统及软件版本

软件名称	版本	获取方式
openEuler	22.03 LTS SP2	获取链接
ovirt-engine	4.4.4.1	通过配置Yum源安装
ovirt-host	4.4.1	通过配置Yum源安装
vdsm	4.40.60.7	通过配置Yum源安装
libvirt	6.2.0	通过配置Yum源安装
Qemu-KVM	6.2.0	通过配置Yum源安装

1.3 环境配置

1.3.1 配置 BIOS

对于不同的硬件设备,通过在BIOS中设置一些高级选项,可以有效提升服务器的性能。

BIOS配置项	选项含义	建议配置值	修改路径
SRIOV	启用或禁用 Single Root Input/Output Virtualization (SRIOV)。	Enabled	BIOS > Advanced > PCIe Configuration > SRIOV
Support Smmu	启用或禁用 SMMU功能。	Enabled	BIOS > Advanced > MISC Configuration > Support Smmu

按照上表推荐的BIOS配置项,配置操作步骤如下:

步骤1 重启服务器,进入BIOS设置界面。

具体操作请参见《TaiShan 服务器 BIOS 参数参考(鲲鹏920处理器)》中"进入 BIOS界面"的相关内容。

步骤2 开启SRIOV。

在BIOS中,依次选择"Advanced > PCIe Configuration > SRIOV",设置为 Enabled。

步骤3开启SMMU。

须知

此优化项只在虚拟化场景使用,非虚拟化场景需关闭。

在BIOS中,依次选择"Advanced > MISC Configuration > Support Smmu",设置为 Enabled。

步骤4 按F10保存BIOS设置,并重启服务器。

----结束

1.3.2 配置启动参数

启用IOMMU(Input/Output Memory Management Unit)的直通模式,使得虚拟机可以直接访问物理设备。

步骤1 打开"/etc/grub2-efi.cfg"文件。

vim /etc/grub2-efi.cfg

步骤2 按"i"进入编辑模式,在下图位置添加如下配置。

iommu.passthrough=1

menuentry 'openEuler (5.10.0-153.12.0.92.0e2203sp2.aarch64) 22.03 (LTS-SP2)'class openeulerclass gnu-linuxclass gn
uclass osunrestricted \$menuentry_id_option 'gnulinux-5.10.0-153.12.0.92.0e2203sp2.aarch64-advanced-cfc5d7e3-e927-457a
-982d-3ca0e97b3f08' {
load_video
set gfxpayload=keep
insmod gzio
insmod part gpt
insmod ext2
set root='hd0.gpt2'
if [x\$feature platform search hint = xy]: then
<pre>searchno-floppyfs-uuidset=roothint-ieee1275='ieee1275//sas/disk@20000.gpt2'hint-bios=hd0.gpt2hi</pre>
nt-efi=hd0.gpt2hint-baremetal=ahci0.gpt2 2c09f3lc-14df-4e06-ba53-146496fff562
else
searchno-floppyfs-uuidset=root 2c09f31c-14df-4e06-ba53-146496fff562
fi
echo 'Loading Linux 5 10 0-153 12 0 92 oe2203sp2 aarch64'
linux /ymlinuz_5_10_0-153_12_0_92_0e2203sp2_aarch64_root=/dev/mapper/openeuler-root_ro_rd_lym_ly=openeuler/root_r
d lym ly=onepuler/swap yideo=VGA-1:640x480-32060me croup disable=files apparentered crashkerpel=1024M high smmu bypassdey=0
x1000:0x17 smmu bypassdev=0x1000:0x15 consol==tv0 iommu passtrouph=1
echo il padine initial randisk
initra /initramfs-5.10.0-153.12.0.92.0e2203sp2.aarch64.img

步骤3 按"Esc"键退出编辑模式,输入:wq!,按"Enter"键保存并退出文件。

步骤4 重启服务器生效配置。

----结束

1.4 安装和部署 ovirt-engine

1.4.1 配置部署环境

为了方便后续在不同的节点进行配置操作,请在所有节点修改主机名称及映射。

步骤1 设置各个节点的主机名。

管理节点:

hostnamectl set-hostname controller hostnamectl --transient set-hostname controller echo "controller" > /etc/hostname

计算节点:

hostnamectl set-hostname compute hostnamectl --transient set-hostname compute echo "compute" > /etc/hostname

修改完成后,重新登录操作系统后生效。

步骤2 修改所有节点的"/etc/hosts"文件。

- 1. 打开文件。 vi /etc/hosts
- 按"i"进入编辑模式,添加如下内容。 ^{管理书点IP地址} controller *计算节点IP地址* compute
- 3. 按 "Esc" 键退出编辑模式,输入:wq!,按 "Enter" 键保存并退出文件。

----结束

1.4.2 安装和部署 ovirt-engine 软件包

在管理节点安装和部署ovirt-engine。

步骤1 安装ovirt-engine。

yum install ovirt-engine python3-distro vdsm-jsonrpc-java-1.5.5 -y

步骤2 从openEuler 20.03-LTS-SP1的源里面下载42.2.4-4版本的postgresql-jdbc。

wget https://repo.huaweicloud.com/openeuler/openEuler-20.03-LTS-SP1/everything/aarch64/Packages/ postgresql-jdbc-42.2.4-4.oe1.noarch.rpm rpm -Uvh --oldpackage postgresql-jdbc-42.2.4-4.oe1.noarch.rpm

步骤3 手动加载Open vSwitch内核模块。

modprobe openvswitch

🛄 说明

- vdsm-jsonrpc-java-1.5.5和postgresql-jdbc-42.2.4-4.oe1.noarch软件包的版本务必须与步骤
 1和步骤2中的版本保持一致,否则ovirt-engine服务将无法启动。
- Open vSwitch内核会因为路径错误而无法自动加载,需要手动操作。

----结束

1.4.3 部署 ovirt-engine 服务

安装ovirt-engine软件包后,还需要在管理节点部署ovirt-engine服务,才能使ovirt-engine软件可用。

步骤1 创建软链接。

In -sf /usr/lib64/libpq.so.private13-5.13 /usr/lib64/libpq.so.5

步骤2 执行部署命令。

engine-setup --offline

部署过程中返回如下信息,此外还需要执行一定的操作,请参考如下信息配置。

[INFO] Stage: Initializing

[INFO] Stage: Environment setup

Configuration files: /etc/ovirt-engine-setup.conf.d/10-packaging-jboss.conf, /etc/ovirt-engine-setup.conf.d/10-packaging.conf

Log file: /var/log/ovirt-engine/setup/ovirt-engine-setup-20211109101534-5yokxt.log Version: otopi-1.9.4 (otopi-1.9.4-2.oe1)

[INFO] Stage: Environment packages setup

- [INFO] Stage: Programs detection
- [INFO] Stage: Environment setup (late)
- [WARNING] Unsupported distribution disabling nfs export
- [INFO] Stage: Environment customization

--== PRODUCT OPTIONS ==--

Configure Cinderlib integration (Currently in tech preview) (Yes, No) [No]: Configure Engine on this host (Yes, No) [Yes]:

Configuring ovirt-provider-ovn also sets the Default cluster's default network provider to ovirtprovider-ovn.

Non-Default clusters may be configured with an OVN after installation. Configure ovirt-provider-ovn (Yes, No) [Yes]: **No**

Configure WebSocket Proxy on this host (Yes, No) [Yes]:

* Please note * : Data Warehouse is required for the engine. If you choose to not configure it on this host, you have to configure it on a remote host, and then configure the engine on this host so that it can access the database of the remote Data Warehouse host. Configure Data Warehouse on this host (Yes, No) [Yes]: Configure Grafana on this host (Yes, No) [Yes]: Configure VM Console Proxy on this host (Yes, No) [Yes]:

--== NETWORK CONFIGURATION ==--

Host fully qualified DNS name of this server [controller]: [WARNING] Host name controller has no domain suffix

[WARNING] Failed to resolve controller using DNS, it can be resolved only locally

Setup can automatically configure the firewall on this system. Note: automatic configuration of the firewall may overwrite current settings. Do you want Setup to configure the firewall? (Yes, No) [Yes]: The following firewall managers wer detected on this system: firewalld Firewall manager to configure (firewalld): **firewalld** [INFO] firewalld will be configured as firewall manager. [WARNING] Host name controller has no domain suffix [WARNING] Host name controller has no domain suffix

--== DATABASE CONFIGURATION ==--

Where is the DWH database located? (Local, Remote) [Local]:

Setup can configure the local postgresql server automatically for the DWH to run. This may conflict with existing applications.

Would you like Setup to automatically configure postgresql and create DWH database, or prefer to perform that manually? (Automatic, Manual) [Automatic]:

Where is the Engine database located? (Local, Remote) [Local]:

Setup can configure the local postgresql server automatically for the engine to run. This may conflict with existing applications.

Would you like Setup to automatically configure postgresql and create Engine database, or prefer to perform that manually? (Automatic, Manual) [Automatic]:

--== OVIRT ENGINE CONFIGURATION ==--

Engine admin password:

Confirm engine admin password: [WARNING] Password is weak: The password is shorter than 8 characters Use weak password? (Yes, No) [No]: **yes**

Application mode (Virt, Gluster, Both) [Both]:

--== STORAGE CONFIGURATION ==--

Default SAN wipe after delete (Yes, No) [No]:

--== PKI CONFIGURATION ==--

Organization name for certificate [Test]:

--== APACHE CONFIGURATION ==--

Setup can configure the default page of the web server to present the application home page. This may conflict with existing applications.

Do you wish to set the application as the default page of the web server? (Yes, No) [Yes]:

Setup can configure apache to use SSL using a certificate issued from the internal CA. Do you wish Setup to configure that, or prefer to perform that manually? (Automatic, Manual) [Automatic]:

--== SYSTEM CONFIGURATION ==--

--== MISC CONFIGURATION ==--

Please choose Data Warehouse sampling scale: (1) Basic (2) Full (1, 2)[1]: Use Engine admin password as initial Grafana admin password (Yes, No) [Yes]:

--== END OF CONFIGURATION ==--

[INFO] Stage: Setup validation

--== CONFIGURATION PREVIEW ==--

A	Application mode : both	
D	Default SAN wipe after delete : Fa	lse
Н	Host FQDN : control	ler
Fi	Firewall manager : firewa	alld
U	Update Firewall : True	
50	Set up Cinderlib integration : Fals	e
	Configure local Engine database : I	rue
26	Set application as default page : In	Je
Ei Fi	Engine database host : local	: Ihost
F	Engine database nost 5432)
E	Engine database secured connection	- False
Ei	Engine database host name validation	: False
Ei	Engine database name : eng	gine
Ei	Engine database user name : e	ngine
Ei	Engine installation : True	
PI	PKI organization : Test	
Se	Set up ovirt-provider-ovn : False	2
G	Grafana Integration : Irue	wirt ongine history grafana
G	Grafana database user name : C	wirt_engine_history_grafana
	DWH installation	ue
D	DWH database host : loca	lhost
D	DWH database port : 543	2
D	DWH database secured connection	: False
D	DWH database host name validation	: False
D	DWH database name : ov	irt_engine_history
C	Configure local DWH database :	True
C	Configure VMConsole Proxy : Ti	rue
וח	Place confirm installation sattings (OK	Cancel) [OK]:
	1 Stage: Transaction setup	
INFO	Stopping engine service	
INFO	1 Stopping ovirt-fence-kdump-listener s	ervice
INFO] Stopping dwh service	
INFO] Stopping vmconsole-proxy service	
INFO] Stopping websocket-proxy service	
INFO] Stage: Misc configuration (early)	
INFO] Stage: Package installation	
INFO	Stage: Misc configuration	
] Upgrading CA	
	Configuring PostgresQL engine database	2
INFO	Creating PostgreSQL	tory' database
INFO	1 Configuring PostgresQL ovint_engine_ms	
INFO	Creating CA: /etc/pki/ovirt-engine/ca.	bem
INFO] Creating CA: /etc/pki/ovirt-engine/qen	nu-ca.pem
INFO] Updating OVN SSL configuration	
INFO] Updating OVN timeout configuration	
INFO] Creating/refreshing DWH database sc	hema
INFO] Setting up ovirt-vmconsole proxy help	er PKI artifacts
INFO] Setting up ovirt-vmconsole SSH PKI ai	tifacts
INFO	Configuring WebSocket Proxy	
	Creating/refreshing Engine database s	cnema
INFO	Creating/refreshing Engine 'internal' d	omain database schema
INFO	Creating default mac pool range	
INFO	Adding default OVN provider to datab	base
INFO	Adding OVN provider secret to databa	ISE
INFO] Setting a password for internal user a	dmin

 [INFO] Generating post install configuration file '/etc/ovirt-engine-setup.conf.d/20-setup-ovirt-post.conf' [INFO] Stage: Transaction commit [INFO] Stage: Closing up [INFO] Starting engine service [INFO] Starting dwh service [INFO] Starting Grafana service [INFO] Restarting ovirt-vmconsole proxy service
== SUMMARY ==
[INFO] Restarting httpd Please use the user 'admin@internal' and password specified in order to login Web access is enabled at: http://controller:80/ovirt-engine https://controller:443/ovirt-engine Internal CA BF:AC:74:64:80:7F:8A:A1:22:91:C3:D7:79:9A:E8:18:3F:D6:BD:39 SSH fingerprint: SHA256:VT2Ygy2uTEA4h+s0RfhLjaGu33Zd2o9De7uJSOohB/0 Web access for grafana is enabled at: https://controller/ovirt-engine-grafana/ Please run the following command on the engine machine controller, for SSO to work: systemctl restart ovirt-engine == END OF SUMMARY ==
 [INFO] Stage: Clean up Log file is located at /var/log/ovirt-engine/setup/ovirt-engine-setup-20211109101534-5yokxt.log [INFO] Generating answer file '/var/lib/ovirt-engine/setup/answers/20211109102045-setup.conf' [INFO] Stage: Pre-termination [INFO] Stage: Termination [INFO] Execution of setup completed successfully
返回 successfully 信息表示成功部署ovirt-engine服务。

- 步骤3 验证ovirt-engine是否已经部署成功。
 - 1. 打开本地PC的"C:\windows\system32\drivers\etc\hosts"文件,添加如下内容。

节点IP地址 controller

- 2. 通过浏览器访问"https://controller/ovirt-engine",进入oVirt管理前端页面。您也可以在此页面切换语言。
- 3. 在oVirt管理前端页面单击"管理门户"进入ovirt-engine登录界面。如果可以进入 ovirt-engine登录界面,表示ovirt-engine已经部署成功。

登录到您的账户	oVirt
用户名	
	oVirt是一个开源的分布式虚拟化解决方案,用于管理整个企业级别的基础架 ^约
密码	12.
域	
internal 👻	
登录	

4. 在ovirt-engine登录界面输入登录用户名和密码。密码是安装ovirt-engine如下过 程中自行设置的,根据实际情况修改。在本例中用户名为admin,密码为 123456。



🗀 说明

- 本例中主机名是"controller", Web管理界面URL是"http://controller:80/ovirtengine"或者"https://controller:443/ovirt-engine"。
- 如果无法访问URL,需要在ovirt engine节点关闭防火墙。 systemctl disable firewalld.service systemctl stop firewalld.service

----结束

1.4.4 清理环境

如果需要重新安装ovirt-engine软件,安装前需要先清理环境。

执行以下命令清理环境。

engine-cleanup

返回successfully表示成功清理环境。预期结果示例:

lroot@loc	alhost multipath]# engine-cleanup
[INFO]	Stage: Initializing
[INFO]	Stage: Environment setup
	Configuration files: /etc/ovirt-engine-setup.conf.d/10-packaging.conf, /etc/ovirt-engine-setup.conf.d/20-setup-ovirt-post.conf
	Log file: /var/log/ovirt-engine/setup/ovirt-engine-remove-20180626122314-u9w0ab.log
	Version: otopi-1.9.1 (otopi-1.9.1-1.el8)
[INFO]	Stage: Environment packages setup
I TNEO 1	Stane Programs detection
I TNEO 1	Stang: Environment customization
[WARNING]	Stage. Environment customization
[WARNING]	Host name hosted-engine-testi has no domain suffix
[WARNING]	Failed to resolve nosted-engine-testi using DNS, it can be resolved only tocally
	PRODUCT OPITIONS
E TNEO 1	Stane. Setur validation
I THEO I	During execution energies will be stepped (or Cancel) [or];
	buing execution engine service with be stopped (ok, cancel) [ok]:
	All the installed ovirt components are about to be removed, data will be lost (OK, Cancel) [Cancel]: OK
I INFO I	Stage: Transaction setup
L INFO J	Stopping engine service
[INFO]	Stopping ovirt-fence-kdump-listener service
[INFO]	Stopping dwh service
[INFO]	Stopping vmconsole-proxy service
[INFO]	Stopping websocket-proxy service
INFO 1	Stage: Misc configuration (early)
I TNEO 1	Stade: Package installation
I TNEO 1	Stand: Misc configuration
TNEO 1	Backing up DKL configuration and kove
	Backing up chi con iguiation and keys
TNFO 1	Backing up database tocathost.engine to /var/tib/ovirt-engine/backups/engine-zoisoozoizzsso.indomo_o.dump .
I INFO J	Clearing Engine database engine
[INFO]	Backing up database localhost;ovirt_engine_history to '/var/lib/ovirt-engine-dwh/backups/dwh-20180626122339.zdh/z31h.dump'.
L INFO J	Clearing DWH database ovirt_engine_history
[INFO]	Removing files
[INFO]	Reverting changes to files
[INFO]	Stage: Transaction commit
[INFO]	Stage: Closing up
	== SUMMARY ==
	Engine setup successfully cleaned up
	A backup of PKT configuration and keys is available at /var/lib/ovirt-engine/backups/engine-pki-201806261223290a2dzd61.tar.gz
	ovirt-engine has been removed
	A backun of the Engine database is available at /var/lib/ovirt.engine/backuns/engine.20180626122330 rrdhm6 o dumo
	A backup of the highe database is available at /var/lib/ovirt-engine/backup/engine-zrioodzorzzzo.righmodump
	A backup of the Dwn database is avaitable at /var/tib/ovirt-engine-uwn/backups/dwn-20100020122339.20172311.dump
	END OF SUMMARY
[TNEO 1	
1 11110 1	Jong i clean up
L THEO 1	Log File is tocated at yvar tog/ovrit-engine/setup/ovrit-engine-remove-zotoovz0122514-uswoat.tog
I INFO]	Otherset De there in the Vary Clover C-engine/set up/answers/20180626122410-cleanup.cont
I INFO J	Stage: Pre-termination
I INFO]	Stage: Termination
I INFO]	Execution of cleanup completed successfully

1.5 安装和部署 ovirt-host

在计算节点安装和部署ovirt-host,需要修改"/usr/libexec/vdsm/hooks/ before_vm_start/50_hostedengine"文件。 步骤1 安装ovirt-host。

yum install -y ovirt-host glibc ovirt-hosted-engine-setup

- 步骤2 手动加载Open vSwitch内核驱动。 modprobe openvswitch
- **步骤3** 打开"/usr/libexec/vdsm/hooks/before_vm_start/50_hostedengine"文件。 vim/usr/libexec/vdsm/hooks/before_vm_start/50_hostedengine
- 步骤4 按"i"进入编辑模式,将文件头"python3"修改为"/usr/bin/python3"。



步骤5 按 "Esc" 键退出编辑模式, 输入:wq!, 按 "Enter" 键保存并退出文件。

----结束

1.6 部署 oVirt 集群

安装和部署ovirt-host完成后,需要部署oVirt集群以管理虚拟机。部署oVirt集群包括新 建主机、新建数据域和ISO域以及创建并配置虚拟机。

- 步骤1 进入oVirt管理前端页面,新建主机。
 - 1. 从页面左侧导航栏选择"计算 > 主机",进入主机页面。
 - 在主机页面单击"新建",打开如图1-3所示页面,填写主机名/IP地址和密码。 页面中的其他参数保持默认即可。
 - "主机名/IP"中填写Host节点的主机名或IP地址。
 - "密码"中填写登录Host节点的密码。

图 1-3 编辑新建主机信息

新建主机				×
常规	>	主机集群	Default	~
电源管理		□ 使用 Foreman/Satellite	<i>数据中心:</i> Default	
SPM		名称		
控制台和 GPU		注释		
内核		主机名/IP 🟮		
承载的引擎		SSH 端口	22	
关联		✔ 安装后激活主机		
		验证 用户名	root	
		● 密码		
		 ○ SSH 公共密钥 ② 高级参数 		

确定取消

3. 单击"确定"保存配置信息。

步骤2 新建数据域。

- 在存储节点新建路径。 mkdir -p /home/domain/data chown vdsm:kvm /home/domain/data
- 2. 新增NFS共享规则。 echo "/home/domain/data *(rw,sync,no_subtree_check,all_squash,anonuid=36,anongid=36)" >> /etc/ exports
- 3. 启动NFS服务。 systemctl enable nfs systemctl start nfs
- 4. 新建域。
 - a. 从页面左侧导航栏选择"存储 > 域",进入存储域页面。
 - b. 单击"新建域",打开如<mark>图1-4</mark>所示页面,选择主机名、填写名称和导出路 径。页面中的其他参数保持默认即可。
 - 主机: 图1-4中的compute为计算节点的hostname,请根据实际场景修 改为主机实际的hostname或者IP地址。
 - 名称:用户根据需要自定义。
 - 导出路径:用户根据需要自定义。

图 1-4 新建数据域

新建域					×
数据中心	Default	~	名称	data	
域功能	数据	~	描述		
存儲类型	NFS	~	注释		
主机 0	compute	~ ~			
导出路径	compute:/home/domain/da 例如:myserver.mydomain.co	ata om:/my/local/p	path		
自定义连接参数					
▶ 高级参数					

c. 单击"确定"保存配置信息。

步骤3 新建ISO域。

- 在存储节点新建路径。 mkdir -p /home/domain/iso chown vdsm:kvm /home/domain/iso
- 2. 新增NFS共享规则。 echo "/home/domain/iso *(rw,sync,no_subtree_check,all_squash,anonuid=36,anongid=36)" >> /etc/ exports
- 3. 重启NFS服务。 systemctl restart nfs
- 4. 新建ISO域。
 - a. 从页面左侧导航栏选择"存储 > 域",进入存储域页面。
 - b. 单击"新建域",打开如<mark>图1-5</mark>所示页面,选择主机名、填写名称和导出路 径。页面中的其他参数保持默认即可。
 - 域功能:选择ISO。
 - 图1-5中的compute为计算节点的hostname,请根据实际场景修改为主机实际的hostname或者IP地址。
 - 名称:用户根据需要自定义。
 - 导出路径:用户根据需要自定义。

图 1-5 新建 ISO 域

新建域			:	×
数据中心	Default ~	名称	ISO	
域功能	ISO ~	描述		
存储类型	NFS ~	注释		
主机 📵	~			
导出路径	compute:/home/domain/iso			
	例如 : myserver.mydomain.com:/my/loca	l/path		
自定义连接参数				
高級参数				

🗀 说明

```
目录中的${iso_domain_id}为ISO域自动创建的ID,images/
11111111-1111-1111-11111111111/为<mark>步骤2.4</mark>操作后自动创建,且不可更改。
```

- **步骤4** 从页面左侧导航栏选择"计算>集群",进入集群页面,检查数据中心状态。

Ħ	计算 > 集群											
C	luster:				* 🛱 ~ Q							
	C ~											
!升缐	2集群兼容级 		注释	兼容性版本	描述	集群 CPU 类型	主机数	VM 数量				
	1	Default		4.4	The default server cluster	Kunpeng 920	1	1				

----结束

1.7 创建虚拟机

步骤1 配置虚拟机基础选项。

1. 从页面左侧导航栏选择"计算 > 虚拟机",进入虚拟机页面。

OVITE OPEN VIRTUALIZATION MANAGER													ı	•	≡• ▲•	0~ 4 ~
① 仪表板	计算》 虚拟机															
	Vms:			x 1	17 V Q				964 <u>8</u> 9643	▶运行 ∨	6. 器停	导出 ■关闭 ~	C重启	早控制台		照迁移
iii 119. >	σ -															1-1 <> 1
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□	名称	注释	主机	IP 地址	FQDN	集群	数据中心	内存	CPU	网络	图形界面	秋志	上线时间	描述	_	-

 在虚拟机页面单击"新建",打开如图1-6所示页面,并在该页面的"名称"右侧 的输入框中填写虚拟机名称(例如VM1)。

图 1-6	创建虚拟机
-------	-------

普通 >	集群		Default ~
系统			数据中心:Default
初始运行	模板		Blank (0) ~
控制台	操作系统 实例类型	63	Other OS ~
主机	优化目标		服务器
高可用性	名称		VM1
资源分配	描述		
引导选项	注释		
随机数生成器	VM Id		
自定义属性	□ 无状态 □ 以暂停模式启动 □ 删除保护 实例随像		
图标			附加 创建 + 一
Foreman/Satellite	选择—个 vNIC 配置集来实例化 VM 网络接口。		+ -
关联	间应金一个坝	Ý	
隐藏高级选项			施定 取消

- 3. 在创建虚拟机页面单击"创建",打开编辑虚拟机信息页面。
- 4. 在"实例镜像"处选择附加或创建磁盘,系统盘需要勾选"可引导的"。

图 1-7 新建虚拟磁盘

新建虚拟磁盘		:
镜像 直接 LUN Cinder	管理的块	
大小 (GB)	50	□ 删除后清理
别名	vm1_Disk1	
描述		
接口	VirtIO-SCSI	
存储域	data (空闲 944 GiB / 总计 1006 GiB)	~
分配策略	精測置备	~
磁盘配置集	data	~
		确定 取消

单击"确定"保存配置信息。

🛄 说明

- 虚拟机磁盘大小需限制为不超过主机的实际可用空间,请根据实际情况设置。
- 请避免添加三块及以上的磁盘。
- 5. 单击"确定"保存配置信息。
- 步骤2 配置控制台选项。
 - 单击"显示高级选项",在创建虚拟机页面选择"控制台",打开如图1-8所示页 面,并在该页面的"图形界面协议"右侧的下拉菜单中选择"VNC",页面中的 其他参数保持默认即可。

图 1-8	编辑控制台信息
-------	---------

新建虚拟机				×
普通	集群		Default	~
系统			数据中心:Default	
初始运行	模板		Blank (0)	~
控制台	操作系统		Other OS	~
主机	(优化目标	83	自定义	~
高可用性	图形控制台・			
资源分配	□ 无头模式 1		0/4	
引导选项	图形界面协议		VNC	
随机数生成器	VNC 键盘格式		默认的[en-us]	~
自定义属性	控制台斷开操作		锁定屏幕	~
图标	监控器		1 ~	
Foreman/Satellite	□ 启用智能卡 单点登陆方法			
关联	 禁用单点登录 使用 Guest Agent 			
	▶ 高级参数			
隐藏高级选顶	日田西卡			确定取消

2. 单击"确定"保存配置信息。

步骤3 按需配置引导选项。

- 1. 继续在创建虚拟机页面选择"引导选项",打开如图1-9所示页面。
- 2. 可选:在该页面的"第二个设备"右侧的下拉菜单中选择"CD-ROM"。
- 3. 勾选"附加CD"。
- 4. 勾选"启用引导菜单"。

图 1-9 编辑引导选项	۶.						
编辑虚拟机				×			
普通	集群		Default	~			
系统			数据中心: Default Blank (0)				
初始运行	模板						
控制台	操作系统		Other OS	~			
主机	优化目标		目定义 服务器	~			
高可用性	引导序列:						
资源分配	第一个设备		硬盘	~			
引导选项	第二个设备		CD-ROM	~			
随机数生成器	✓ 月別目号菜单		openEuler-22.03-L1S-SP2-aarch64-dv V	2			
自定义属性							
图标							
Foreman/Satellite							
关联							
隐藏高级选项			确定	取消			

步骤4 (可选)按需配置自定义属性qemucmdline,详细配置操作请参见《鲲鹏BoostKit虚 拟化使能套件 故障案例》中的如何在oVirt创建虚拟机时自定义属性。

-----结束

1.8 配置 oVirt 虚拟机网络

为保证集群中虚拟机之间的网络可以互通,需要配置oVirt虚拟机网络。在配置oVirt虚 拟机网络前,需要确保已完成BIOS配置和配置启动参数。

步骤1 确认已完成BIOS配置和配置启动参数。详细信息请参见1.3 环境配置中的相关操作。

可通过以下命名确认环境是否配置成功。 dmesg | grep iommu

预期结果示例:

[root@localbost ~]# dmess grep jommu												
[0.000000] Kernel command line: BOOT IMAGE=/vmlinuz-5.10.0-153.12.0.92.0e2203sp2.aarch64 roo												
t=/dev/mapped/company company company company and company and company comp												
x^{4} (a) x^{2} (b) x^{2} (b) x^{2} (c) x^{2												
control sectoring (1) dup_uisable=lifes appaintor-o crashering(=10244, high smmu) bypassdev=0x1000.0x1/												
Simula by assume - 0.000 console - clyo de laute nugepages2-10240 nugepages2-10240 nugepages-000												
10mmu, passim ough=1												
[2.419483] 10mmu: Default domain type: Passtnrougn (set via kernel command line)												
1 14.225000] ehci-pci 0000:7a:01.0: Adding to iommu group 0												
[14.371682] ehci-pci 0000:ba:01.0: Adding to iommu group 1												
[14.541374] ohci-pci 0000:7a:00.0: Adding to iommu group 2												
[14.728498] ohci-pci 0000:ba:00.0: Adding to iommu group 3												
[14.944009] xhci_hcd 0000:7a:02.0: Adding to iommu group 4												
[15.187038] xhci hcd 0000:ba:02.0: Adding to iommu group 5												
[16.247656] pcieport 0000:00:00.0: Adding to iommu group 6												
[16.274601] pcieport 0000:00:04.0: Adding to iommu group 7												
[16.300987] pcieport 0000:00:08.0: Adding to iommu group 8												
[16.326906] pcieport 0000:00:0c.0: Adding to iommu group 9												
[16.352386] pcieport 0000:00:10.0: Adding to iommu group 10												
[16.377651] pcieport 0000:00:11.0: Adding to iommu group 11												
[16.402369] pcieport 0000:00:12.0: Adding to iommu group 12												
[16.426983] pcieport 0000:04:00.0: Adding to iommu group 13												
[16.435019] pcieport 0000:05:00.0: Adding to iommu group 14												
[16.442970] pcieport 0000:05:01.0: Adding to jommu group 15												
[16.450869] prieport 0000:05:02.0: Adding to jommu group 16												
[16.458697] prieport 0000.05:03.0: Adding to jommu group 17												

步骤2 单击虚拟机名称进入虚拟机配置页面,在虚拟机配置页面选择"主机设备",进入主机设备页面。



步骤3 单击在主机设备页面右上角的"添加设备",打开添加主机设备页面,如下<mark>图1-10</mark>所

示。在添加主机设备页面勾选物理机上可用的Mellanox CX5网口,然后单击 🤍 添加 到主机设备中,最后单击"确定"保存信息。

🛄 说明

物理机上可用的Mellanox CX5网口可以登录服务器后使用如下命令查看。

lspci | grep -i net

82:00.0 Ethernet controller: Mellanox Technologies MT27800 Family [ConnectX-5] 82:00.1 Ethernet controller: Mellanox Technologies MT27800 Family [ConnectX-5]

倫定 取消

图 1-10 添加主机设备页面

Ē	到主机	controller	~					
ł	3	pci	~					
\$	的主机设备							
	名称		厂商	产品	驱动程序	当前被使用	附加到虚拟机	IOMMU 组
	pci_0000_b	4_02_0	Huawei Technologies Co., Ltd. (0	HiSilicon SAS 3.0 HBA (0xa230)	hisi_sas_v3_hw			40
	pci_0000_7	/d_00_0	Huawei Technologies Co., Ltd. (0	HNS GE/10GE/25GE RDMA Network Controller (0xa222)	hns3			22
	pci_0000_7	/d_00_2	Huawei Technologies Co., Ltd. (0	HNS GE/10GE/25GE RDMA Network Controller (0xa222)	hns3			27
	pci_0000_7	/d_00_1	Huawei Technologies Co., Ltd. (0	HNS GE/10GE/25GE Network Controller (0xa221)	hns3			26
	pci_0000_b	oa_02_0	Huawel Technologies Co., Ltd. (0	HiSilicon USB 3.0 Host Controller (0xa238)	xhci_hcd			5
	pci_0000_0	2_00_0	Broadcom / LSI (0x1000)	MegaRAID Tri-Mode SAS3508 (0x0016)	megaraid_sas			21
pci_0000_7a_02_0 Huawei Technologies Co., Ltd. (0			Huawei Technologies Co., Ltd. (0	HiSilicon USB 3.0 Host Controller (0xa238)	xhci_hcd			4
1	的主机设备		F #	<u>*</u> "	897h42sh	N56018645.00	84th0750eth11140	IOMMU #
	12147		1 88	/~nn	36914313	=1094000640	PE17月1日1月1日20月1日	IOMINO H

步骤4 主机设备添加成功后,单击"运行",启动虚拟机。

🛄 说明

创建虚拟机后,虚拟机处于关闭状态,显示为Down。

计测	计算,曲时间																
Vn	Vma X 0 × 0. 588 × 0 × 0. 588											B体照 迁移 :					
2	Ø → 112 ()									1-2 <> 1							
			名称	注释	主机	IP 地址	FQDN	樂群	数据中心	内存	CPU	网络	图形界面	808	上线时间	描述	
	1	-	VM1		host			Default	Default				VNC	Up	20 min		A.
			VM2		node1			Default	Default	36%		196	VINC	Ho	33 min		

步骤5 验证网络是否配置成功。

- 查看虚拟机xml文件,确认已存在<hostdev>标签。 virsh edit VM1
- 2. 进入虚拟机即可正常配置网口的IP地址。

8	ei		Q	VM1:1		ĸ ^a	=		×
1.1.1									
"test	" [Net	J] 1L,	5B ы	ritten					
[root	@loca	lhost	home 1	t scp -r test root0192.168.200.12:/home					
The a	uthent	ticity	of ho	ost '192.168.200.12 (192.168.200.12)' can't be established.					
EDZ53	keu is	j fing	kpoum	it is SHH256; jMbrXcyHY2UKPHJUITu/VrS/hKVu/b7KJrjcVwydUII.					
Are L	IOU SU	e uou	want	to continue connecting (ues/no/[fingerprint])? ues					
Warni	ing: Pe	ermane	ntly a	added '192.168.200.12' (ED25519) to the list of known hosts.					
Autho	prized	users	only	. All activities may be monitored and reported.					
root	192.10	68.200	.12's	password:					
test					100%	5	3.5KB/s	00:	00
Lroo1	loca.	lhost	home 1						
Iroot	ay ar.	host	poss:	t ning 192 168 200 12					
PING	192 16	58 200	12 (1	192 168 200 12) 56(84) butes of data					
64 bu	ites fi	om 19	2.168	.200.12: icmp seg=1 ttl=64 time=0.566 ms					
64 bu	ites fi	om 19	2.168	200.12: icmp seg=2 ttl=64 time=0.078 ms					
64 bu	ites fi	om 19	2.168	.200.12: icmp_seq=3 ttl=64 time=0.078 ms					
^C -									
1	92.168	3.200.	12 pin	ng statistics					
З рас	kets t	transm	itted.	, 3 received, 0% packet loss, time 2038ms					
rtt n	nin/ave	1/max/	mdev :	0.078/0.240/0.566/0.230 ms					
Lroot	loca	lhost	home It	t ip a					
1: 10); (LU	JPBHCK		JWER_UP) MTU 55535 qdisc noqueue state URKNUWR group default gien 1999					
	net 1	oppac oppac	1/8	2000 00.00.00 Dru 00.00.00.00.00					
	ual	A 1ft	fore	er preferred lft forever					
	net6	:1/12	8 scm	host					
	val	id lft	fores	ver wreferred lft forever					
2: er	np5s0m	DØ: <₿	ROADCA	AST,MULTICAST,UP,LOWER UP> mtu 1500 gdisc mg state UP group default glen	1000				
i i i	ink/e	ther 8	c:2a:8	Be:57:ff:97 brd ff:ff:ff:ff:ff:ff					
1	inet 19	32.168	.200.3	34/24 scope global enp5s0np0					
	val	id_lft	fore	ver preferred_lft forever					
Iroot	@loca	lhost	home]	t ping 192.168.200.12					
PING	192.10	68.200	.12 (192.168.200.12) 56(84) bytes of data.					
04 DU	ites fi	OM 19	2 160	200.12: icmp_seq=1 ttl=04 time=0.546 ms					
01 DL	res 11	011 13	a.100	courter temp_seq=2 til=01 time=0.000 ms					
1	92 168	299	12 nii	ng statistics					
2 pac	kets	ransm	itted	Z received, 0% packet loss, time 1025ms					
rtt n	nin/ave	/max/	mdev :	= 0.095/0.320/0.546/0.225 ms					

----结束

2 鲲鹏 oVirt 轻量级虚拟化管理平台 部署指南 (openEuler 20.03)

2.1 简介

鲲鹏oVirt轻量级虚拟化软件(以下简称oVirt)是一个基于KVM的轻量级虚拟化软件,整合了许多优秀的开源软件,旨在替代VMware vsphere成为企业虚拟化的解决方案。它包括ovirt-engine、主机节点和存储节点三部分,支持本地存储和外部存储。本文主要介绍如何在使用openEuler 20.03操作系统的鲲鹏服务器上安装、部署ovirt-engine服务以及部署oVirt集群。

2.2 环境要求

本文基于鲲鹏服务器和openEuler操作系统提供指导,在正式操作前请确保软硬件均满 足要求。

2.3 配置oVirt软件源 请在所有节点上完成本章节的操作。

2.4 编译oVirt相关软件包

2.5 修改主机名称

2.6 安装ovirt-engine

2.7 安装和部署ovirt-host

在计算节点安装和部署ovirt-host,需要修改"/usr/libexec/vdsm/hooks/ before_vm_start/50_hostedengine"文件。

2.8 部署oVirt集群

安装和部署ovirt-host完成后,需要部署oVirt集群以管理虚拟机。部署oVirt集群包括新 建主机、新建数据域和ISO域以及创建并配置虚拟机。

2.9 创建虚拟机

2.10 故障排除

2.1 简介

鲲鹏oVirt轻量级虚拟化软件(以下简称oVirt)是一个基于KVM的轻量级虚拟化软件, 整合了许多优秀的开源软件,旨在替代VMware vsphere成为企业虚拟化的解决方案。 它包括ovirt-engine、主机节点和存储节点三部分,支持本地存储和外部存储。本文主 要介绍如何在使用openEuler 20.03操作系统的鲲鹏服务器上安装、部署ovirt-engine 服务以及部署oVirt集群。

oVirt是RedHat商业版本虚拟化软件RHEV的开源版本,它整合使用了libvirt、 gluster、patternfly、ansible等一系列优秀的开源软件,目前已经成为了企业虚拟化环 境可选的解决方案,相比OpenStack的庞大和复杂,oVirt在企业私有云建设的部署和 维护中具备了使用简单的优势。

oVirt部署架构如图2-1所示。





oVirt项目中的不同组件主要包含三个部分:

- 管理节点部署ovirt-engine,用来进行管理虚拟机(创建、开关启停)、配置网络和存储等操作。
- 一个或多个计算节点(Host node),用来运行虚拟机。
 计算节点安装有VDSM和libvirt组件的Linux发行版,也包含一些用来实现网络虚拟化和其它系统服务的组件。
- 一个或多个存储节点(Local storage或Shared storage),用来存放虚拟机镜像和ISO镜像。

存储节点可以使用块存储或文件存储,可以利用计算节点自身的存储做存储节点 (local on host模式),或者使用外部的存储,例如通过NFS访问,或者是IP-SAN/FC-SAN。还有一种就是超融合架构,通过Gluster将各节点的磁盘组成池来 使用,同时能够实现高可用和冗余。

关于oVirt的更多信息请访问oVirt官网。

2.2 环境要求

本文基于鲲鹏服务器和openEuler操作系统提供指导,在正式操作前请确保软硬件均满 足要求。

鲲鹏oVirt轻量级部署涉及的主要软件版本和物理机要求如表2-1所示。

表 2-1	软件版本和物理机要求
-------	------------

软件名称	版本	获取方式
openEuler	20.03 LTS SP1	https://www.openeuler.org/zh/ download/
ovirt-engine	4.4.4.1	通过编译获取,详细信息请参见 <mark>2.4 编译</mark> oVirt相关软件包
ovirt-host	4.4.1.2	通过编译获取,详细信息请参见 <mark>2.4 编译</mark> oVirt <mark>相关软件包</mark>
vdsm	4.40.60	通过编译获取,详细信息请参见 <mark>2.4 编译</mark> oVirt <mark>相关软件包</mark>
libvirt	6.2.0	ovirt部署时自动安装
QEMU-KVM	4.1.0	ovirt部署时自动安装

表 2-2 硬件环境

项目	要求
服务器	鲲鹏服务器
СРО	鲲鹏920处理器

2.3 配置 oVirt 软件源

请在所有节点上完成本章节的操作。

- 步骤1 修改OS、everything、EPOL软件源。
 - 1. 打开文件。 vi /etc/yum.repos.d/openEuler.repo
 - 2. 按"i"进入编辑模式,添加"priority=1"参数,优先使用OS、everything、 EPOL软件源。

[OS]
name=OS
baseurl=http://repo.openeuler.org/openEuler-20.03-LTS-SP1/OS/\$basearch/
enabled=1
gpgcheck=1
gpgkey=http://repo.openeuler.org/openEuler-20.03-LTS-SP1/0S/\$basearch/RPM-GPG-KEY-openEuler
priority=1
[everything]
name=everything
baseurl=http://repo.openeuler.org/openEuler-20.03-LTS-SP1/everything/\$basearch/
enabled=1
gpgcheck=1
<pre>gpgkey=http://repo.openeuler.org/openEuler-20.03-LTS-SP1/everything/\$basearch/RPM-GPG-KEY-openEuler</pre>
priority=1
LEPOL J
name=EPOL
baseurl=http://repo.openeuler.org/openEuler-20.03-LTS-SP1/EPOL/\$basearch/
enabled=1
gpgcheck=1
<pre>gpgkey=http://repo.openeuler.org/openEuler-20.03-LTS-SP1/OS/\$basearch/RPM-GPG-KEY-openEuler</pre>
priority=1

3. 按"Esc"键退出编辑模式,输入:wq!,按"Enter"键保存并退出文件。

步骤2新增SP1-Epol-update软件源。

echo "[openEuler-EPOL-update] name=openEuler-epol-update baseurl=https://mirrors.huaweicloud.com/openeuler/openEuler-20.03-LTS-SP1/EPOL/update/aarch64/ enabled=1 gpgcheck=0 " >> /etc/yum.repos.d/openEulerOS.repo

步骤3 新增oVirt本地软件源。

echo "[ovirt-local] name=ovirt-local baseurl=file:///root/ovirt_rpm/ enabled=0 gpgcheck=0 priority=1 " >> /etc/yum.repos.d/openEulerOS.repo

步骤4 将openEuler的update源设置为"enable"。

yum-config-manager --enable update

步骤5 清理缓存并重新创建缓存。

yum clean all && yum makecache

----结束

2.4 编译 oVirt 相关软件包

须知

- 以下编译的软件包,都汇总至"/root/ovirt_rpm"目录,此目录可直接作为本地源 使用。
- 因编译耗时较久,可以在一次编译完成后,将"/root/ovirt_rpm"目录备份,且在 后续的部署中跳过编译,直接使用。
- 部分包的编译依赖为编译过程中实时更新在本地源中。如果发现安装依赖失败,提示找不到包,可以执行以下命令更新yum缓存后,重新安装。 yum makecache

步骤1 取消环境变量,加速RPM包编译。

unset GOMP_CPU_AFFINITY

步骤2 创建Maven缓存目录。

mkdir ~/.m2

步骤3 添加Maven源。

```
echo "
<?xmlversion=\"1.0\"?>
<settings xmlns=\"http://maven.apache.org/POM/4.0.0\"
xmlns:xsi=\"http://www.w3.org/2001/XMLSchema-instance\"
xsi:schemaLocation=\"http://maven.apache.org/POM/4.0.0
http://maven.apache.org/xsd/settings-1.0.0.xsd\">
<mirrors>
<mirrors>
<mirrors>
<mirrors>
<mirror>
<id>huaweicloud</id>
<mirrorS>
<url>http://mirrors.huaweicloud.com/repository/maven/</url>
<mirrorOf>*</mirrorOf>
</mirrors>
</settings>" > /root/.m2/settings.xml
```

门 说明

如外网访问需要代理,则在本步骤所修改的内容中,<settings>标签下,添加<proxies>标签,内 容如下:

<proxies> <proxy> <active>true</active> <protocol>http</protocol> <username>代理用户名</username> <password>代理密码</password> <host>代理地址</host> <port>代理端口</port> </proxy> <proxy> <active>true</active> <protocol>https</protocol> <username>代理用户名</username> <password>代理密码</password> <host>代理地址</host> <port>代理端口</port> </proxy> </proxies>

步骤4 创建缓存目录。

mkdir /root/ovirt_build mkdir /root/ovirt_rpm mkdir -p /root/rpmbuild/SOURCES cd /root/ovirt build

步骤5 安装基础软件。

yum install tar git vim rpm-build createrepo_c java-11-openjdk -y

步骤6 编译ovirt-jboss-modules-maven-plugin。

下载源码

git clone https://gitee.com/src-openeuler/ovirt-jboss-modules-maven-plugin.git -b openEuler-20.03-LTS-SP1 # 将源码拷贝到构建目录 cp ovirt-jboss-modules-maven-plugin/* ~/rpmbuild/SOURCES # 安装构建所需依赖 yum-builddep -y ovirt-jboss-modules-maven-plugin/ovirt-jboss-modules-maven-plugin.spec # 构建rpm包 rpmbuild -bb ovirt-jboss-modules-maven-plugin/ovirt-jboss-modules-maven-plugin.spec # 将产物复制到缓存目录 cp /root/rpmbuild/RPMS/noarch/ovirt-jboss-modules-maven-plugin-2.0.2-*.noarch.rpm ../ovirt_rpm # 清理构建残留
 rm -f ~/rpmbuild/SOURCES/*
 # 刷新本地源内容
 createrepo_c ../ovirt_rpm
 yum-config-manager --enable ovirt-local
 yum makecache

步骤7 编译ovirt-engine。

git clone https://gitee.com/src-openeuler/ovirt-engine.git -b openEuler-20.03-LTS-SP1

cp ovirt-engine/* ~/rpmbuild/SOURCES

yum-builddep -y ovirt-engine/ovirt-engine.spec

rpmbuild -bb ovirt-engine/ovirt-engine.spec cp /root/rpmbuild/RPMS/noarch/ovirt-engine-vmconsole-proxy-helper-4.4.4.1-*.noarch.rpm ../ovirt_rpm

cp /root/rpmbuild/RPMS/noarch/ovirt-engine-setup-4.4.4.1-*.noarch.rpm ../ovirt_rpm

cp /root/rpmbuild/RPMS/noarch/ovirt-engine-health-check-bundler-4.4.4.1-*.noarch.rpm ../ovirt_rpm

cp /root/rpmbuild/RPMS/noarch/ovirt_engine-websocket-proxy-4.4.4.1-*.noarch.rpm ../ovirt_rpm

cp /root/rpmbuild/RPMS/noarch/ovirt-engine-tools-backup-4.4.4.1-*.noarch.rpm ../ovirt_rpm

cp /root/rpmbuild/RPMS/noarch/ovirt-engine-setup-plugin-imageio-4.4.4.1-*.noarch.rpm ../ovirt_rpm cp /root/rpmbuild/RPMS/noarch/ovirt-engine-setup-plugin-vmconsole-proxy-helper-4.4.4.1-*.noarch.rpm ../

cp/root/pmbulu/kPMS/hoarch/ovirt-engine-setup-plugin-vinconsole-proxy-netper-4.4.4.1-*.hoarch.rpm .../ ovirt_rpm

cp /root/rpmbuild/RPMS/noarch/python3-ovirt-engine-lib-4.4.4.1-*.noarch.rpm ../ovirt_rpm

cp /root/rpmbuild/RPMS/noarch/ovirt-engine-setup-plugin-websocket-proxy-4.4.4.1-*.noarch.rpm ../ovirt_rpm

cp /root/rpmbuild/RPMS/noarch/ovirt-engine-setup-plugin-cinderlib-4.4.4.1-*.noarch.rpm ../ovirt_rpm

cp /root/rpmbuild/RPMS/noarch/ovirt-engine-setup-base-4.4.4.1-*.noarch.rpm ../ovirt_rpm

cp /root/rpmbuild/RPMS/noarch/ovirt-engine-setup-plugin-ovirt-engine-common-4.4.4.1-*.noarch.rpm ../ ovirt_rpm

cp /root/rpmbuild/RPMS/noarch/ovirt-engine-tools-4.4.4.1-*.noarch.rpm ../ovirt_rpm

cp /root/rpmbuild/RPMS/noarch/ovirt-engine-setup-plugin-ovirt-engine-4.4.4.1-*.noarch.rpm ../ovirt_rpm

cp /root/rpmbuild/RPMS/noarch/ovirt-engine-dbscripts-4.4.4.1-*.noarch.rpm ../ovirt_rpm

cp /root/rpmbuild/RPMS/noarch/ovirt-engine-restapi-4.4.4.1-*.noarch.rpm ../ovirt_rpm

cp /root/rpmbuild/RPMS/noarch/ovirt-engine-backend-4.4.4.1-*.noarch.rpm ../ovirt_rpm

cp /root/rpmbuild/RPMS/noarch/ovirt-engine-4.4.4.1-*.noarch.rpm ../ovirt_rpm

cp /root/rpmbuild/RPMS/noarch/ovirt-engine-webadmin-portal-4.4.4.1-*.noarch.rpm ../ovirt_rpm rm -f ~/rpmbuild/SOURCES/*

createrepo_c ../ovirt_rpm

步骤8 编译ovirt-ansible-collection。

git clone -b openEuler-20.03-LTS-SP1 https://gitee.com/src-openeuler/ovirt-ansible-collection.git cp ovirt-ansible-collection/* ~/rpmbuild/SOURCES yum-builddep -y ovirt-ansible-collection/ovirt-ansible-collection.spec rpmbuild -bb ovirt-ansible-collection/ovirt-ansible-collection.spec cp /root/rpmbuild/RPMS/noarch/ovirt-ansible-collection-1.5.3-*.noarch.rpm ../ovirt_rpm/ rm -f ~/rpmbuild/SOURCES/* createrepo_c ../ovirt_rpm

步骤9 编译ovirt-dependencies。

git clone -b openEuler-20.03-LTS-SP1 https://gitee.com/src-openeuler/ovirt-dependencies.git cp ovirt-dependencies/* ~/rpmbuild/SOURCES yum-builddep -y ovirt-dependencies/ovirt-dependencies.spec rpmbuild -bb ovirt-dependencies/ovirt-dependencies.spec cp /root/rpmbuild/RPMS/noarch/ovirt-dependencies-4.4.2-*.noarch.rpm ../ovirt_rpm/ rm -f ~/rpmbuild/SOURCES/* createrepo_c ../ovirt_rpm

步骤10 编译ovirt-engine-nodejs-modules。

git clone -b openEuler-20.03-LTS-SP1 https://gitee.com/src-openeuler/ovirt-engine-nodejs-modules.git cp ovirt-engine-nodejs-modules/* ~/rpmbuild/SOURCES yum-builddep -y ovirt-engine-nodejs-modules/ovirt-engine-nodejs-modules.spec rpmbuild -bb ovirt-engine-nodejs-modules/ovirt-engine-nodejs-modules.spec cp /root/rpmbuild/RPMS/noarch/ovirt-engine-nodejs-modules-2.0.53-*.noarch.rpm ../ovirt_rpm/ rm -f ~/rpmbuild/SOURCES/* createrepo_c ../ovirt_rpm yum makecache

步骤11 编译ovirt-engine-ui-extensions。

git clone -b openEuler-20.03-LTS-SP1 https://gitee.com/src-openeuler/ovirt-engine-ui-extensions.git cp ovirt-engine-ui-extensions/* ~/rpmbuild/SOURCES yum-builddep -y ovirt-engine-ui-extensions/ovirt-engine-ui-extensions.spec rpmbuild -bb ovirt-engine-ui-extensions/ovirt-engine-ui-extensions.spec cp /root/rpmbuild/RPMS/noarch/ovirt-engine-ui-extensions-1.2.6-*.noarch.rpm ../ovirt_rpm rm -f ~/rpmbuild/SOURCES/* createrepo_c ../ovirt_rpm

步骤12 编译ovirt-engine-wildfly。

git clone -b openEuler-20.03-LTS-SP1 https://gitee.com/src-openeuler/ovirt-engine-wildfly.git cp ovirt-engine-wildfly/* ~/rpmbuild/SOURCES yum-builddep -y ovirt-engine-wildfly/ovirt-engine-wildfly.spec rpmbuild -bb ovirt-engine-wildfly/ovirt-engine-wildfly.spec cp /root/rpmbuild/RPMS/aarch64/ovirt-engine-wildfly-19.1.0-*.aarch64.rpm ../ovirt_rpm rm -f ~/rpmbuild/SOURCES/* createrepo_c ../ovirt_rpm

步骤13 编译ovirt-engine-wildfly-overlay。

git clone -b openEuler-20.03-LTS-SP1 https://gitee.com/src-openeuler/ovirt-engine-wildfly-overlay.git cp ovirt-engine-wildfly-overlay/* ~/rpmbuild/SOURCES yum-builddep -y ovirt-engine-wildfly-overlay/ovirt-engine-wildfly-overlay.spec rpmbuild -bb ovirt-engine-wildfly-overlay/ovirt-engine-wildfly-overlay.spec cp /root/rpmbuild/RPMS/noarch/ovirt-engine-wildfly-overlay-19.1.0-*.noarch.rpm ../ovirt_rpm rm -f ~/rpmbuild/SOURCES/* createrepo_c ../ovirt_rpm

步骤14 编译ovirt-imageio。

git clone -b openEuler-20.03-LTS-SP1 https://gitee.com/src-openeuler/ovirt-imageio.git cp ovirt-imageio/* ~/rpmbuild/SOURCES yum-builddep -y ovirt-imageio/ovirt-imageio.spec rpmbuild -bb ovirt-imageio/ovirt-imageio-daemon-2.1.1-*.aarch64.rpm ../ovirt_rpm cp /root/rpmbuild/RPMS/aarch64/ovirt-imageio-client-2.1.1-*.aarch64.rpm ../ovirt_rpm cp /root/rpmbuild/RPMS/aarch64/ovirt-imageio-client-2.1.1-*.aarch64.rpm ../ovirt_rpm rm -f ~/rpmbuild/SOURCES/* createrepo_c ../ovirt_rpm

步骤15 编译ovirt-provider-ovn。

git clone -b openEuler-20.03-LTS-SP1 https://gitee.com/src-openeuler/ovirt-provider-ovn.git cp ovirt-provider-ovn/* ~/rpmbuild/SOURCES yum-builddep -y ovirt-provider-ovn/ovirt-provider-ovn.spec rpmbuild -bb ovirt-provider-ovn/ovirt-provider-ovn.spec cp /root/rpmbuild/RPMS/noarch/ovirt-provider-ovn-driver-1.2.29-*.noarch.rpm ../ovirt_rpm cp /root/rpmbuild/RPMS/noarch/ovirt-provider-ovn-1.2.29-*.noarch.rpm ../ovirt_rpm rm -f ~/rpmbuild/SOURCES/* createrepo_c ../ovirt_rpm

步骤16 编译ovirt-cockpit-sso。

git clone -b openEuler-20.03-LTS-SP1 https://gitee.com/src-openeuler/ovirt-cockpit-sso.git cp ovirt-cockpit-sso/* ~/rpmbuild/SOURCES yum-builddep -y ovirt-cockpit-sso/ovirt-cockpit-sso.spec rpmbuild -bb ovirt-cockpit-sso/ovirt-cockpit-sso.spec cp /root/rpmbuild/RPMS/noarch/ovirt-cockpit-sso-0.1.4-*.noarch.rpm ../ovirt_rpm/ rm -f ~/rpmbuild/SOURCES/* createrepo_c ../ovirt_rpm

步骤17 编译ovirt-engine-metrics。

git clone -b openEuler-20.03-LTS-SP1 https://gitee.com/src-openeuler/ovirt-engine-metrics.git cp ovirt-engine-metrics/* ~/rpmbuild/SOURCES yum-builddep -y ovirt-engine-metrics/ovirt-engine-metrics.spec rpmbuild -bb ovirt-engine-metrics/ovirt-engine-metrics.spec cp /root/rpmbuild/RPMS/noarch/ovirt-engine-metrics-1.3.6.4-*.noarch.rpm ../ovirt_rpm/ rm -f ~/rpmbuild/SOURCES/* createrepo_c ../ovirt_rpm

步骤18 编译ovirt-engine-dwh。

git clone -b openEuler-20.03-LTS-SP1 https://gitee.com/src-openeuler/ovirt-engine-dwh.git cp ovirt-engine-dwh/* ~/rpmbuild/SOURCES yum-builddep -y ovirt-engine-dwh/ovirt-engine-dwh.spec rpmbuild -bb ovirt-engine-dwh/ovirt-engine-dwh.spec cp /root/rpmbuild/RPMS/noarch/ovirt-engine-dwh-grafana-integration-setup-4.4.4.1-*.noarch.rpm ../ ovirt_rpm/ cp /root/rpmbuild/RPMS/noarch/ovirt-engine-dwh-setup-4.4.4.1-*.noarch.rpm ../ovirt_rpm/ cp /root/rpmbuild/RPMS/noarch/ovirt-engine-dwh-4.4.4.1-*.noarch.rpm ../ovirt_rpm/ rm -f ~/rpmbuild/SOURCES/* createrepo_c ../ovirt_rpm

步骤19 编译collectd。

git clone -b openEuler-20.03-LTS-SP1 https://gitee.com/src-openeuler/collectd.git cp collectd/* ~/rpmbuild/SOURCES yum-builddep -y collectd/collectd.spec rpmbuild -bb collectd/collectd.spec --nocheck cp /root/rpmbuild/RPMS/aarch64/collectd-*.rpm ../ovirt_rpm/ cp /root/rpmbuild/RPMS/aarch64/libcollectdclient-*.rpm ../ovirt_rpm/ cp /root/rpmbuild/RPMS/aarch64/perl-Collectd-*.rpm ../ovirt_rpm/ rm -f ~/rpmbuild/SOURCES/* createrepo_c ../ovirt_rpm

步骤20 编译ovirt-host。

git clone -b openEuler-20.03-LTS-SP1 https://gitee.com/src-openeuler/ovirt-host.git cp ovirt-host/* ~/rpmbuild/SOURCES yum-builddep -y ovirt-host/ovirt-host.spec rpmbuild -bb ovirt-host/ovirt-host.spec cp /root/rpmbuild/RPMS/aarch64/ovirt-host-dependencies-4.4.1-*.aarch64.rpm ../ovirt_rpm/ cp /root/rpmbuild/RPMS/aarch64/ovirt-host-4.4.1-*.aarch64.rpm ../ovirt_rpm/ rm -f ~/rpmbuild/SOURCES/* createrepo_c ../ovirt_rpm

步骤21 编译cockpit-ovirt。

git clone -b openEuler-20.03-LTS-SP1 https://gitee.com/src-openeuler/cockpit-ovirt.git cp cockpit-ovirt/* ~/rpmbuild/SOURCES yum-builddep -y cockpit-ovirt/cockpit-ovirt.spec rpmbuild -bb cockpit-ovirt/cockpit-ovirt.spec cp /root/rpmbuild/RPMS/noarch/cockpit-ovirt-dashboard-0.15.0-*.noarch.rpm ../ovirt_rpm rm -f ~/rpmbuild/SOURCES/* createrepo_c ../ovirt_rpm

步骤22 编译FreeIPA。

🛄 说明

如果编译发生错误,请参考2.10.1 编译FreeIPA失败的解决方法排除故障。

git clone -b openEuler-20.03-LTS-SP1 https://gitee.com/src-openeuler/freeipa.git cp freeipa/* ~/rpmbuild/SOURCES yum-builddep -y freeipa/freeipa.spec --nobest rpmbuild -bb freeipa/freeipa.spec cp /root/rpmbuild/RPMS/noarch/freeipa-client-samba-4.9.3-*.noarch.rpm ../ovirt_rpm cp /root/rpmbuild/RPMS/noarch/freeipa-client-samba-4.9.3-*.noarch.rpm ../ovirt_rpm cp /root/rpmbuild/RPMS/aarch64/freeipa-client-common-4.9.3-*.noarch.rpm ../ovirt_rpm cp /root/rpmbuild/RPMS/aarch64/freeipa-client-epn-4.9.3-*.aarch64.rpm ../ovirt_rpm cp /root/rpmbuild/RPMS/aarch64/freeipa-client-epn-4.9.3-*.aarch64.rpm ../ovirt_rpm cp /root/rpmbuild/RPMS/aarch64/freeipa-client-4.9.3-*.noarch.rpm ../ovirt_rpm cp /root/rpmbuild/RPMS/noarch/python3-ipalib-4.9.3-*.noarch.rpm ../ovirt_rpm cp /root/rpmbuild/RPMS/noarch/python3-ipaclient-4.9.3-*.noarch.rpm ../ovirt_rpm m -f ~/rpmbuild/RPMS/noarch/freeipa-common-4.9.3-*.noarch.rpm ../ovirt_rpm rm -f ~/rpmbuild/SOURCES/* createrepo_c ../ovirt_rpm

步骤23 编译ovirt-hosted-engine-setup。

git clone -b openEuler-20.03-LTS-SP1 https://gitee.com/src-openeuler/ovirt-hosted-engine-setup.git cp ovirt-hosted-engine-setup/* ~/rpmbuild/SOURCES yum-builddep -y ovirt-hosted-engine-setup/ovirt-hosted-engine-setup.spec rpmbuild -bb ovirt-hosted-engine-setup/ovirt-hosted-engine-setup.spec cp /root/rpmbuild/RPMS/noarch/ovirt-hosted-engine-setup-2.5.0-*.noarch.rpm ../ovirt_rpm rm -f ~/rpmbuild/SOURCES/* createrepo_c ../ovirt_rpm

步骤24 编译certmonger。

git clone -b openEuler-20.03-LTS-SP1 https://gitee.com/src-openeuler/certmonger.git cp certmonger/* ~/rpmbuild/SOURCES yum-builddep -y certmonger/certmonger.spec rpmbuild -bb certmonger/certmonger.spec --nocheck cp /root/rpmbuild/RPMS/aarch64/certmonger-0.79.11-*.aarch64.rpm ../ovirt_rpm rm -f ~/rpmbuild/SOURCES/* createrepo_c ../ovirt_rpm

步骤25 编译vdsm-jsonrpc-java。

git clone -b openEuler-20.03-LTS-SP1 https://gitee.com/src-openeuler/vdsm-jsonrpc-java.git cp vdsm-jsonrpc-java/* ~/rpmbuild/SOURCES yum-builddep -y vdsm-jsonrpc-java/vdsm-jsonrpc-java.spec rpmbuild -bb vdsm-jsonrpc-java/vdsm-jsonrpc-java.spec cp /root/rpmbuild/RPMS/noarch/vdsm-jsonrpc-java-1.5.5-*.noarch.rpm ../ovirt_rpm rm -f ~/rpmbuild/SOURCES/* createrepo_c ../ovirt_rpm

步骤26 编译vdsm。

git clone -b openEuler-20.03-LTS-SP1 https://gitee.com/src-openeuler/vdsm.git cp vdsm/* ~/rpmbuild/SOURCES yum-builddep -y vdsm/vdsm.spec rpmbuild -bb vdsm/vdsm.spec cp -f /root/rpmbuild/RPMS/noarch/vdsm-*.rpm ../ovirt_rpm cp -f /root/rpmbuild/RPMS/aarch64/vdsm-*.rpm ../ovirt_rpm rm -f ~/rpmbuild/SOURCES/* createrepo_c ../ovirt_rpm yum makecache

步骤27 编译ovirt-hosted-engine-ha。

git clone -b openEuler-20.03-LTS-SP1 https://gitee.com/src-openeuler/ovirt-hosted-engine-ha.git cp ovirt-hosted-engine-ha/* ~/rpmbuild/SOURCES yum-builddep -y ovirt-hosted-engine-ha/ovirt-hosted-engine-ha.spec rpmbuild -bb ovirt-hosted-engine-ha/ovirt-hosted-engine-ha.spec cp /root/rpmbuild/RPMS/noarch/ovirt-hosted-engine-ha-2.4.5-*.noarch.rpm ../ovirt_rpm rm -f ~/rpmbuild/SOURCES/* createrepo_c ../ovirt_rpm

步骤28 拷贝oVirt软件包到所有部署ovirt的节点。

拷贝/root/ovirt_rpm目录到所有部署ovirt的节点的/root目录下。

在所有部署ovirt的节点执行如下操作,清理缓存并重新创建缓存。 yum-config-manager --enable ovirt-local yum clean all && yum makecache

----结束

2.5 修改主机名称

步骤1 设置各个节点的主机名。

hostnamectl set-hostname HOSTNAME

须知

HOSTNAME: 主机名。不能包含".#\$"等特殊字符。

修改完成后,重新登录操作系统后生效。

- 步骤2 基于组网要求,修改所有节点的"/etc/hosts"文件。
 - 1. 打开文件。

vi /etc/hosts

2. 按"i"进入编辑模式,添加如下内容。 HOSTIP1 HOSTNAME1 HOSTIP2 HOSTNAME2

🛄 说明

HOSTIP: 主机ip *HOSTNAME:* 主机名。 例如: 192.168.14.2 controller

3. 按"Esc"键退出编辑模式,输入:wq!,按"Enter"键保存并退出文件。

----结束

2.6 安装 ovirt-engine

在管理节点安装和部署ovirt-engine。

2.6.1 安装和部署 ovirt-engine 软件包

安装和部署ovirt-engine软件包过程中,vdsm-jsonrpc-java-1.5.5、postgresqljdbc-42.2.4-4.oe1.noarch软件包的版本务必与列表一致,否则ovirt-engine服务将无法 启动。

步骤1 安装ovirt-engine。

yum install ovirt-engine python3-distro postgresql-jdbc-42.2.4-4.oe1.noarch vdsm-jsonrpc-java-1.5.5 python3-mod_wsgi-4.6.4-3.oe1 python3-pyOpenSSL -y

步骤2 检查vdsm-jsonrpc-java版本信息。

rpm -qa|grep vdsm-jsonrpc-java

查看是否为1.5.5版本:

[root@localhost ~]# rpm -qa|grep vdsm-jsonrpc-java vdsm-jsonrpc-java-1.5.5-2.noarch

- 是,进入步骤3。
- 否,重新安装,建议安装vdsm-jsonrpc-java-1.5.5。
 yum install vdsm-jsonrpc-java-1.5.5 -y
- 步骤3 检查ovirt-engine-wildfly的版本信息。

rpm -qa|grep ovirt-engine-wildfly

查看是否为19.1.0版本:

```
[root@localhost ovirt_build]# rpm -qa|grep ovirt-engine-wildfly
ovirt-engine-wildfly-19.1.0-2.aarch64
ovirt-engine-wildfly-overlay-19.1.0-2.noarch
```

- 是,进入<mark>步骤4</mark>。
- 否,重新安装。 yum install ovirt-engine-wildfly-19.1.0 -y
- **步骤4** 升级OpenSSH与sudo。OpenSSH与sudo版本过低会导致oVirt服务无法正常启动。 yum update -y openssh sudo

步骤5 手动加载Open vSwitch内核模块。

modprobe openvswitch

▲ 注意

操作系统重启后需要重新加载。

----结束

2.6.2 部署 ovirt-engine 服务

安装ovirt-engine软件包后,还需要在管理节点部署ovirt-engine服务,才能使ovirt-engine软件可用。

步骤1 创建软链接。

In -sf /usr/lib64/libpq.so.private13-5.13 /usr/lib64/libpq.so.5

步骤2 执行部署命令。

engine-setup --offline

[INFO] Stage: Initializing

[INFO] Stage: Environment setup

Configuration files: /etc/ovirt-engine-setup.conf.d/10-packaging-jboss.conf, /etc/ovirt-engine-setup.conf.d/10-packaging.conf

Log file: /var/log/ovirt-engine/setup/ovirt-engine-setup-20211109101534-5yokxt.log

Version: otopi-1.9.4 (otopi-1.9.4-2.oe1) [INFO] Stage: Environment packages setup

- [INFO] Stage: Programs detection
- [INFO] Stage: Environment setup (late)
- [WARNING] Unsupported distribution disabling nfs export
- [INFO] Stage: Environment customization

--== PRODUCT OPTIONS ==--

Configure Cinderlib integration (Currently in tech preview) (Yes, No) [No]: Configure Engine on this host (Yes, No) [Yes]:

Configuring ovirt-provider-ovn also sets the Default cluster's default network provider to ovirt-provider-ovn.

Non-Default clusters may be configured with an OVN after installation. Configure ovirt-provider-ovn (Yes, No) [Yes]: **No** Configure WebSocket Proxy on this host (Yes, No) [Yes]:

* Please note * : Data Warehouse is required for the engine. If you choose to not configure it on this host, you have to configure it on a remote host, and then configure the engine on this host so that it can access the database of the remote Data Warehouse host. Configure Data Warehouse on this host (Yes, No) [Yes]: Configure Grafana on this host (Yes, No) [Yes]: Configure VM Console Proxy on this host (Yes, No) [Yes]:

--== NETWORK CONFIGURATION ==--

Host fully qualified DNS name of this server [controller]: [WARNING] Host name controller has no domain suffix

[WARNING] Failed to resolve controller using DNS, it can be resolved only locally

Setup can automatically configure the firewall on this system.

Note: automatic configuration of the firewall may overwrite current settings.

Do you want Setup to configure the firewall? (Yes, No) [Yes]: [INFO] firewalld will be configured as firewall manager.

[WARNING] Host name controller has no domain suffix

[WARNING] Host name controller has no domain suffix

--== DATABASE CONFIGURATION ==--

Where is the DWH database located? (Local, Remote) [Local]:

Setup can configure the local postgresql server automatically for the DWH to run. This may conflict with existing applications.

Would you like Setup to automatically configure postgresql and create DWH database, or prefer to perform that manually? (Automatic, Manual) [Automatic]:

Where is the Engine database located? (Local, Remote) [Local]:

Setup can configure the local postgresql server automatically for the engine to run. This may conflict with existing applications.

Would you like Setup to automatically configure postgresql and create Engine database, or prefer to perform that manually? (Automatic, Manual) [Automatic]:

--== OVIRT ENGINE CONFIGURATION ==--

Engine admin password:

Confirm engine admin password: [WARNING] Password is weak: The password is shorter than 8 characters Use weak password? (Yes, No) [No]: Yes Application mode (Virt, Gluster, Both) [Both]:

--== STORAGE CONFIGURATION ==--

Default SAN wipe after delete (Yes, No) [No]:

--== PKI CONFIGURATION ==--

Organization name for certificate [Test]:

--== APACHE CONFIGURATION ==--

Setup can configure the default page of the web server to present the application home page. This may conflict with existing applications.

Do you wish to set the application as the default page of the web server? (Yes, No) [Yes]:

Setup can configure apache to use SSL using a certificate issued from the internal CA. Do you wish Setup to configure that, or prefer to perform that manually? (Automatic, Manual) [Automatic]:

--== SYSTEM CONFIGURATION ==--

--== MISC CONFIGURATION ==--

Please choose Data Warehouse sampling scale: (1) Basic

- (2) Full
- (1, 2)[1]:

Use Engine admin password as initial Grafana admin password (Yes, No) [Yes]:

--== END OF CONFIGURATION ==--

[INFO] Stage: Setup validation

--== CONFIGURATION PREVIEW ==--

Application mode	: both
Default SAN wipe after delete	: False
Host FQDN	: controller
Firewall manager	: firewalld
Update Firewall	: True
Set up Cinderlib integration	: False
Configure local Engine database	: True
Set application as default page	: True
Configure Apache SSL	: True

Engine database host	: localhost
Engine database port	: 5432
Engine database secured connect	tion : False
Engine database host name vali	dation : False
Engine database name	: engine
Engine database user name	: engine
Engine installation	: Irue
PKI organization	: lest
Grafana integration	
Grafana database user name	: ovirt engine history grafana
Configure WebSocket Proxy	: True
DWH installation	: True
DWH database host	: localhost
DWH database port	: 5432
DWH database secured connect	on : False
DWH database host name valida	ation : False
DWH database name	: ovirt_engine_history
Configure local DWH database	: Irue
Configure viviConsole Proxy	. Irue
Please confirm installation settir	ngs (OK_Cancel) [OK].
[INFO] Stage: Transaction setup	
[INFO] Stopping engine service	
[INFO] Stopping ovirt-fence-kdump-l	istener service
[INFO] Stopping dwh service	
[INFO] Stopping vmconsole-proxy se	rvice
[INFO] Stopping websocket-proxy set	rvice
[INFO] Stage: Misc configuration (ea	rly)
[INFO] Stage: Package installation	
[INFO] Stage: Misc configuration	
[INFO] Upgrading CA	databasa
[INFO] Creating PostgresQL engine	ualabase
[INFO] Creating PostgreSQL 'ovirt_er	igine history' database
[INFO] Configuring PostgresQ2 of the_er	gine_instory dutabase
[INFO] Creating CA: /etc/pki/ovirt-en	gine/ca.pem
[INFO] Creating CA: /etc/pki/ovirt-en	gine/qemu-ca.pem
[INFO] Updating OVN SSL configurat	tion
[INFO] Updating OVN timeout config	guration
[INFO] Creating/refreshing DWH dat	abase schema
[INFO] Setting up ovirt-vmconsole pr	oxy helper PKI artifacts
[INFO] Setting up ovirt-vmconsole S	oH PKI artifacts
[INFO] Configuring WebSocket Proxy	tahaca cahama
[INFO] Creating a user for Grafana	
[INFO] Creating/refreshing Engine 'ir	ternal' domain database schema
[INFO] Creating default mac pool rai	lae
[INFO] Adding default OVN provider	to database
[INFO] Adding OVN provider secret t	o database
[INFO] Setting a password for intern	al user admin
[INFO] Install selinux module /usr/sh	are/ovirt-engine/selinux/ansible-runner-service.cil
[INFO] Generating post install config	uration file '/etc/ovirt-engine-setup.conf.d/20-setup-ovirt-post.conf'
[INFO] Stage: Transaction commit	
[INFO] Stage: Closing up	
[INFO] Starting engine service	
[INFO] Starting dwn service	
[INFO] Starting Galaria service	any contico
	oxy service
== SUMMARY ==	
[INFO] Restarting httpd	
Please use the user 'admin@inte	rnal' and password specified in order to login
Web access is enabled at:	
http://controller:80/ovirt-engi	ne
https://controller:443/ovirt-en	gine
Internal CA BF:AC:74:64:80:7F:8A	x:A1:22:91:C3:D7:79:9A:E8:18:3F:D6:BD:39

SSH fingerprint: SHA256:VT2Ygy2uTEA4h+s0RfhLjaGu33Zd2o9De7uJSOohB/0

Web access for grafana is enabled at:

https://controller/ovirt-engine-grafana/ Please run the following command on the engine machine controller, for SSO to work: systemctl restart ovirt-engine

--== END OF SUMMARY ==--

[INFO] Stage: Clean up

Log file is located at /var/log/ovirt-engine/setup/ovirt-engine-setup-20211109101534-5yokxt.log [INFO] Generating answer file '/var/lib/ovirt-engine/setup/answers/20211109102045-setup.conf'

- [INFO] Stage: Pre-termination [INFO] Stage: Termination

[INFO] Execution of setup completed successfully

步骤3 web登录验证。

打开本地PC "c:\windows\system32\drivers\etc\hosts" 文件,添加<节点 IP><hostname>。

Restarting httpd	
Please use the user 'admin@internal' and password specified in order to login	
Web access is enabled at:	
http://controller:80/ovirt-engine	
https://controller:443/ovirt-engine	
Internal CA_BF:AC:74:64:80:7F:8A:A1:22:91:C3:D7:79:9A:E8:18:3F:D6:BD:39	
SSH fingerprint: SHA256:VT2Ygv2uTEA4h+s0RfhLiaGu33Zd2o9De7uJS0ohB/0	
Web access for grafana is enabled at:	
https://controller/ovirt-engine-grafana/	
Please run the following command on the engine machine controller. for SSO to work:	
systemctl restart ovirt-engine	
== END OF SUMMARY ==	
Stage: Clean up	
Log file is located at /var/log/ovirt-engine/setup/ovirt-engine-setup-20211109101534-5yokxt.log	1
Generating answer file '/var/lib/ovirt-engine/setup/answers/20211109102045-setup.conf'	1
Stage: Pre-termination	
Stage: Termination	
Execution of setup completed successfully	

使用浏览器访问https://controller/ovirt-engine可以进入Ovirt-engine登录界面。

登录到您的账户		o\∕irt		
用户名				
±10		oVirt是一个开源的分布式虚拟化解决方案,用于管理	理整个企业级别的基础深	
0510		139		
域				
internal	-			



	== OVIRT ENGINE CONFIGURATION ==
	Engine admin password: Confirm engine admin password:
[WARNING]	Password is weak: The password is shorter than 8 characters Use weak password? (Yes, No) [No]: Yes Application mode (Virt, Gluster, Both) [Both]: Use default credentials (admin@internal) for ovirt-provider-ovn (Yes, No) [Yes]:

🛄 说明

- 本示例手册中主机名是"controller", web管理界面url是"http://controller:80/ovirtengine"或者"https://controller:443/ovirt-engine", 实际情况根据环境替换。
- 如果无法访问,需要在ovirt engine节点关闭防火墙。 systemctl disable firewalld.service systemctl stop firewalld.service

----结束

2.6.3 备份 engine

如果需要清理安装,可以进行备份engine。

执行以下命令进行备份engine。

engine-backup

返回Done表示成功备份engine。预期结果示例:

```
[root@localhost multipath]# engine-backup
Start of engine-backup with mode 'backup'
scope: all
archive file: /var/lib/ovirt-engine-backup/ovirt-engine-backup-20180626122236.backup
log file: /var/log/ovirt-engine-backup/ovirt-engine-backup-20180626122236.log
Backing up:
Notifying engine
- Files
- Engine database 'engine'
- DWH database 'ovirt_engine_history'
Packing into file '/var/lib/ovirt-engine-backup/ovirt-engine-backup-20180626122236.backup'
Notifying engine
Done.
```

2.6.4 清理环境

如果需要重新安装ovirt-engine软件,安装前需要先清理环境。

执行以下命令清理环境。

engine-cleanup

返回successfully表示成功清理环境。预期结果示例:

[INFO] Stage: Initializing [INFO] Stage: Environment setup Configuration files: /etc/ovirt-engine-setup.conf.d/10-packaging.conf, /etc/ovirt-engine-setup.conf.d/20setup-ovirt-post.conf Log file: /var/log/ovirt-engine/setup/ovirt-engine-remove-20180626122314-u9w0ab.log Version: otopi-1.9.1 (otopi-1.9.1-1.el8) [INFO] Stage: Environment packages setup [INFO] Stage: Programs detection [INFO] Stage: Environment customization [WARNING] Host name hosted-engine-test1 has no domain suffix [WARNING] Failed to resolve hosted-engine-test1 using DNS, it can be resolved only locally --== PRODUCT OPTIONS ==--[INFO] Stage: Setup validation During execution engine service will be stopped (OK, Cancel) [OK]: All the installed ovirt components are about to be removed, data will be lost (OK, Cancel) [Cancel]: OK [INFO] Stage: Transaction setup [INFO] Stopping engine service

- [INFO] Stopping ovirt-fence-kdump-listener service
- [INFO] Stopping dwh service
- [INFO] Stopping vmconsole-proxy service
- [INFO] Stopping websocket-proxy service

- [INFO] Stage: Misc configuration (early)
- [INFO] Stage: Package installation
- [INFO] Stage: Misc configuration
- [INFO] Backing up PKI configuration and keys
- [INFO] Backing up database localhost:engine to '/var/lib/ovirt-engine/backups/
- engine-20180626122330.rrqbm6_o.dump'.
- [INFO] Clearing Engine database engine
- [INFO] Backing up database localhost:ovirt_engine_history to '/var/lib/ovirt-engine-dwh/backups/
- dwh-20180626122339.zdh7z31h.dump'.
- [INFO] Clearing DWH database ovirt_engine_history
- [INFO] Removing files
- [INFO] Reverting changes to files
- [INFO] Stage: Transaction commit
- [INFO] Stage: Closing up

--== SUMMARY ==--

Engine setup successfully cleaned up A backup of PKI configuration and keys is available at /var/lib/ovirt-engine/backups/enginepki-20180626122329oa2dzd61.tar.gz ovirt-engine has been removed A backup of the Engine database is available at /var/lib/ovirt-engine/backups/ engine-20180626122330.rrqbm6_o.dump A backup of the DWH database is available at /var/lib/ovirt-engine-dwh/backups/ dwh-20180626122339.zdh7z31h.dump

--== END OF SUMMARY ==--

[INFO] Stage: Clean up Log file is located at /var/log/ovirt-engine/setup/ovirt-engine-remove-20180626122314-u9w0ab.log [INFO] Generating answer file '/var/lib/ovirt-engine/setup/answers/20180626122410-cleanup.conf' [INFO] Stage: Pre-termination [INFO] Stage: Termination [INFO] Stage: Termination [INFO] Execution of cleanup completed successfully [root@localhost multipath]# engine-cleanup [TMFO] Stage: Initializing [INFO] Stage: Initializing

TNEO 1	Stage Environment setup
I THEO I	Scape Environment setup
	Configuration lites: /etc/ovirt-engine-setup.con//o/packaging.con/, /etc/ovirt-engine-setup.con/.d/20-setup-ovirt-post.com
	Log file: /var/tog/ovirt-engine/setup/ovirt-engine-remove-zoiz80020122314-uaw0ab.tog
	Version: otopi-1.9.1 (otopi-1.9.1-1.et8)
[INFO]	Stage: Environment packages setup
[INFO]	Stage: Programs detection
[INFO]	Stage: Environment customization
[WARNING]	Host name hosted-engine-testl has no domain suffix
[WARNING]	Failed to resolve hosted-engine-testlysing DNS, it can be resolved only locally
[
	= PRODUCT OPTIONS ==
E TNEO 1	Stage: Setup validation
1 100 1	Suger sector variation
	burning execution engine service with be stopped (ok, cancer) [ok]:
	All the installed ovirt components are about to be removed, data will be lost (UK, Cancel) [Cancel]: UK
L INFO J	Stage: Transaction setup
[INFO]	Stopping engine service
[INFO]	Stopping ovirt-fence-kdump-listener service
[INFO]	Stopping dwh service
[INFO]	Stopping vmconsole-proxy service
[TNEO]	Stopping websocket-proxy service
I TNEO 1	Stage: Misc configuration (early)
TNEO 1	Stage, hist configuration (care),
I INFO 1	Stage. Package installation
I INFO J	Stage: Misc configuration
I INFO J	Backing up PKI configuration and keys
[INFO]	Backing up database localhost:engine to '/var/lib/ovirt-engine/backups/engine-20180626122330.rrqbm6_o.dump'.
[INFO]	Clearing Engine database engine
[INFO]	Backing up database localhost:ovirt engine history to '/var/lib/ovirt-engine-dwh/backups/dwh-20180626122339.zdh7z31h.dump'.
TNEO 1	Clearing DWH database ovirt engine history
I TNEO 1	Removing files
TNEO 1	Poverting changes to files
I INFO 1	Reverting changes to files
I INFO J	Stage: Plansaction commit
L TNEO 1	Stage: closing up
	== SUMMARY ==
	Engine setup successfully cleaned up
	A backup of PKI configuration and keys is available at /var/lib/ovirt-engine/backups/engine-pki-20180626122329oa2dzd61.tar.gz
	ovirt-engine has been removed
	A backup of the Engine database is available at /var/lib/ovirt-engine/backups/engine-20180626122330.rrqbm6_o.dump
	A backup of the DWH database is available at /var/lib/ovirt-engine-dwh/backups/dwh-20180626122339.zdh7z31h.dump
	== END OF SUMMARY ==
[INFO]	Stage: Clean up
	Log file is located at /var/log/ovirt-engine/setup/ovirt-engine-remove-20180626122314-u9w0ab log
[TNEO]	Generating answer file '/yar/lih/ovirt.engine/setun/answers/20180626122410.cleanun conf'
[TNEO]	Stage Provide
I INFO 1	Stage. Pre-termination
I INFO]	Stage: Termination
L INFO	Execution of cleanup completed successfully

2.7 安装和部署 ovirt-host

在计算节点安装和部署ovirt-host,需要修改"/usr/libexec/vdsm/hooks/ before_vm_start/50_hostedengine"文件。

- 步骤1 安装ovirt-host。 yum install -y ovirt-host glibc ovirt-hosted-engine-setup
- **步骤2** 手动加载Open vSwitch内核驱动。 modprobe openvswitch

⚠ 注意

操作系统重启后需要重新加载

步骤3 升级OpenSSH与sudo。

OpenSSH与sudo版本过低会导致oVirt服务无法正常启动。 yum update -y openssh sudo

- 步骤4 修改"/usr/libexec/vdsm/hooks/before_vm_start/50_hostedengine"文件。
 - 1. 打开文件。 vi /usr/libexec/vdsm/hooks/before_vm_start/50_hostedengine
 - 2. 按"i"进入编辑模式,将文件头"python3"修改为"/usr/bin/python3"。



3. 按"Esc"键退出编辑模式,输入:wq!,按"Enter"键保存并退出文件。

----结束

2.8 部署 oVirt 集群

安装和部署ovirt-host完成后,需要部署oVirt集群以管理虚拟机。部署oVirt集群包括新 建主机、新建数据域和ISO域以及创建并配置虚拟机。

步骤1 在管理节点修改"/usr/share/ovirt-engine/ansible-runner-service-project/project/ roles/ovirt-host-deploy-facts/tasks/host-os.yml"文件。

vi /usr/share/ovirt-engine/ansible-runner-service-project/project/roles/ovirt-host-deploy-facts/tasks/hostos.yml

将如下内容:

```
- name: Check if system is OpenEuler
set_fact:
ansible_python_interpreter: "{{ 'usr/bin/python3' }}"
when:
ansible_distribution == 'openEuler'
```

改为:

<pre>- name: Check if system is OpenEuler set_fact: ansible_python_interpreter: "{{ '/usr/bin/python3' }}" when: ansible_distribution == 'openEuler'</pre>	
<pre> name: Detect host operating system set_fact: el_ver: "{{ ansible_distribution_major_version int if ansible_distribution == 'RedHat' or ansible_dist else 0 }}" fc_ver: "{{ ansible_distribution_major_version int if (ansible_distribution == 'Fedora' or ansible_dist else 0 }}"</pre>	ribution == 'CentOS' tribution == 'openEuler')
<pre>- name: Check if system is OpenEuler set_fact: ansible_python_interpreter: "{{ '/usr/bin/python3' }}" when: ansible_distribution == 'openEuler'</pre>	

步骤2 在管理节点修改"/var/lib/ovirt-engine/external_truststore"的权限组。

🛄 说明

如文件/var/lib/ovirt-engine/external_truststore不存在,则跳过此步骤。 chown ovirt:ovirt /var/lib/ovirt-engine/external_truststore

步骤3 在管理节点安装必要软件包。

yum install -y python3-flask python3-flask-restful python3-ansible-runner

步骤4 在管理节点修改"/usr/share/ovirt-engine/ansible-runner-service-project/project/ roles/ovirt-host-deploy-vdsm/tasks/restart_services.yml"。

vi /usr/share/ovirt-engine/ansible-runner-service-project/project/roles/ovirt-host-deploy-vdsm/tasks/ restart_services.yml

```
在 "name: Start and enable services"之前添加如下内容:
```

 name: Reconfigure vdsm tool command: vdsm-tool configure --force changed when: True

```
    vdsmd.service

10

    supervdsmd.service

11
     - libvirtd.service
12

    ovirt-imageio.service

13

    NetworkManager.service

14
15
     when: "item in ansible_facts.services"
16
     name: Reconfigure vdsm tool
17
18
     command: vdsm-tool configure --force
19
     changed_when: True
20
21 - name: Start and enable services
     service:
22
23
       name: "{{ item }}"
24
       state: started
25
       enabled: yes
```

步骤5 进入oVirt管理前端页面,新建主机。

- 1. 进入"管理门户",从页面左侧导航栏选择"计算 > 主机",进入主机页面。
- 2. 在主机页面单击"新建",打开如<mark>图2-2</mark>所示页面,填写名称、主机名/IP地址和 密码。页面中的其他参数保持默认即可。
 - "主机名/IP"中填写Host节点的主机名或IP地址。
 - "密码"中填写登录Host节点的密码。

图 2-2 编辑新建主机信息

新建主机			×
常规	>	主机集群	Default ~ 数据中心: Default
电源管理		使用 Foreman/Satellite	
SPM		名称	
控制台和 GPU		注释	
内核		主机名/IP 🟮	
承载的引擎		SSH 端口	22
关联		✔ 安装后激活主机	
		验证 用户名	root
		● 密码	
		 ○ SSH 公共密钥 ⑧ 高级参数 	

- **确定** 取消
- 3. 单击"确定"保存配置信息,提示的"电源管理配置"可根据需要进行配置。
- 4. 主机创建后,ovirt将自动安装配置,请等待主机状态变为"Up"。

图 2-3 主机状态

C	~							
		名称	注释	主机名/IP	集群	数据中心	状态	虚
<u>~</u>		controller			Default	Default	Up	

步骤6 新建数据域。

- 1. 在存储节点新建路径。 mkdir -p /home/domain/data chown vdsm:kvm /home/domain/data
- 2. 新增NFS共享规则。 echo "/home/domain/data *(rw,sync,no_subtree_check,all_squash,anonuid=36,anongid=36)" >> /etc/ exports
- 3. 启动NFS服务。 systemctl enable nfs systemctl start nfs

- 4. 新建域。
 - a. 从页面左侧导航栏选择"存储>域",进入存储域页面。
 - b. 单击"新建域",打开如<mark>图2-4</mark>所示页面,选择主机名、填写名称和导出路 径。页面中的其他参数保持默认即可。
 - 主机: 图2-4中的compute为计算节点的hostname,请根据实际场景修 改为主机实际的hostname或者IP地址。
 - 名称:用户根据需要自定义。
 - 导出路径:用户根据需要自定义。

图 2-4 新建数据域

新建域			×
数据中心	Default ~	名称	data
域功能	数据 ~	描述	
存储类型	NFS ~	注释	
主机 ()	compute ~		
导出路径			
	例如:myserver.mydomain.com:/my/loca	l/path	
自定义连接参数			
▶ 高级参数			

- c. 单击"确定"保存配置信息。
- d. 数据域创建后,ovirt将自动安装配置,请等待跨数据中心状态变为"活跃的"。

状态	域名	注释	域类型	存储类型	格式	▼ 跨数据中心状态
	data		数据 (主)	NFS	V5	活跃的
_	ISO		ISO	NFS	V1	活跃的

步骤7 新建ISO域。

 在存储节点新建路径。 mkdir -p /home/domain/iso chown vdsm:kvm /home/domain/iso

2. 新增NFS共享规则。

echo "/home/domain/iso *(rw,sync,no_subtree_check,all_squash,anonuid=36,anongid=36)" >> /etc/ exports

- 3. 重启NFS服务。 systemctl restart nfs
- 4. 新建ISO域。
 - a. 从页面左侧导航栏选择"存储>域",进入存储域页面。
 - b. 单击"新建域",打开如<mark>图</mark>2-5所示页面,选择主机名、填写名称和导出路 径。页面中的其他参数保持默认即可。
 - 域功能:选择ISO。
 - 图2-5中的compute为计算节点的hostname,请根据实际场景修改为主机实际的hostname或者IP地址。
 - 名称:用户根据需要自定义。

■ 导出路径:用户根据需要自定义。

图 2-5 新建 ISO 域

新建域			×
数据中心	Default ~	名称	ISO
域功能	ISO ~	描述	
存储类型	NFS ~	注释	
主机 0	~		
导出路径	compute:/home/domain/iso 例如:myserver.mydomain.com:/my/loca	l/path	
自定义连接参数			
高级参数			

c. ISO域创建后,ovirt将自动安装配置,请等待跨数据中心状态变为"活跃的"。

	data		数据 (主)	NFS	V5	活跃的	
状态	域名	注释	域类型	存储类型	格式	▼ 跨数排	时心状态

门 说明

目录中的\${iso_domain_id}为ISO域自动创建的ID,images/ 11111111-1111-1111-111111111111/为<mark>步骤6.4</mark>操作后自动创建,且不可更改。

6. 修改镜像的权限组。

步骤8 从页面左侧导航栏选择"计算>集群",进入集群页面,检查数据中心状态。

计算 > 集群							
Cluster:				x 🕸 - Q			
2 -							
!升级集群兼容编		注释	兼容性版本	描述	集群 CPU 类型	主机数	VM 数量
1	Default		4.4	The default server cluster	Kunpeng 920	1	1

----结束

2.9 创建虚拟机

- 步骤1 从页面左侧导航栏选择"计算>虚拟机",进入虚拟机页面。
- 步骤2 单击"新建",打开新建虚拟机页面。单击"显示高级选项",显示更多配置项。
- 步骤3 配置虚拟机基础选项。
 - 1. 在新建虚拟机页面选择"普通"选项卡,打开如<mark>图2-6</mark>所示页面,并在该页面的 "名称"右侧的输入框中填写虚拟机名称(例如vm1)。

图 2-6 基础选项		
新建虚拟机		×
普通 >	集群	Default
系统		数据中心:Default
初始运行	模板	Blank (0)
控制台	操作系统 实例类型 (3)	Other OS ~ 自定义 ~
主机	优化目标	服务器
高可用性	名称	vm1
资源分配	描述	
引导选项	注释	
随机数生成器		
自定义属性	□ 无状态 □ 以暫停模式启动 □ 删除保护 实例镜像	
图标		附加创建 + 一
Foreman/Satellite	选择一个 vNIC 配置集来实例化 VM 网络接口。	+-
¥₩	ovirtingino ovirtingini	
隐藏高级选项		輸定 取消

2. 在"实例镜像"处选择附加或创建磁盘,系统盘需要勾选"可引导的"。

图 2-7 新建虚拟磁盘

新建虚拟磁盘			×
镜像 直接 LUN Cinder 管理的	ŧ.		
大小 (GB)	50		
别名	vm1_Disk1	 ✓ 미키号印 ○ 可共享的 ○ 只读 	
描述			
接口	VirtlO-SCSI ~		
存储域	data (空闲 944 GiB / 总计 1006 GiB)		
分配策略	精简置备		
磁盘配置集	data 🗸		

3. 单击"确定"保存配置信息。

门 说明

- 虚拟机磁盘大小需限制为不超过主机的实际可用空间,请根据实际情况设置。
- 请避免添加三块及以上的磁盘。
- 步骤4 配置虚拟机规格。

在新建虚拟机页面选择"系统"选项卡,配置内存和CPU数。

确定 取消

图 2-8 系统			
新建虚拟机			×
普通	集群		Default
系统			数据中心:Default
初始运行	模板		Blank (0) V
次制台	操作系统		Other OS ~
고메니	实例类型	83	自定义
主机	优化目标		服务器
高可用性	内存大小	63	8196 MB
资源分配	最大内存 🔒	63	32784 MB
引导选项	保证的物理内存 🟮	63	8196 MB
随机数生成器	虚拟 CPU 的总数 3 ● 高级参数	83	4
占古以已件	常规		
日定又商住	硬件时钟时间偏移 🕄		默认的: (GMT+00:00) GMT Standard Time ~
图标	序列号策略		集群默认 (主机 ID) ~
Foreman/Satellite	自定义序列号		
关联			
隐藏高级选项			喻定 取消

步骤5 配置控制台选项。

 在新建虚拟机页面选择"控制台"选项卡,打开如图2-9所示页面,并在该页面的 "图形界面协议"右侧的下拉菜单中选择"VNC",页面中的其他参数保持默认 即可。

新建虚拟机				
普通	集群		Default	~
系统			数据中心:Default	
叨始运行	模板		Blank (0)	~
空制台	操作系统		Other OS	~
	实例类型	63	自定义	~
±1/1	优化目标		服务器	~
高可用性	图形控制台:			
资源分配	 元头模式 ① 无头模式 ① 元头模式 		QXL	~
引导选项	图形界面协议		VNC	Ý
随机数生成器	VNC 键盘格式		默认的[en-us]	~
自定义属性	控制台断开操作		锁定屏幕	~
劉标	监控器		1 ~	
	 USB 支持 □ 启用智能卡 			
Foreman/Satellite	单点登陆方法			
关联	 ● 葉用甲点登录 ● 使用 Guest Agent 			
	● 吉切 台北			
	2 高级参数			

2. 点击"确定"保存配置信息。

步骤6 按需配置引导选项。

- 1. 在新建虚拟机页面选择"引导选项"选项卡,打开如<mark>图2-10</mark>所示页面。
- 2. 可选:在该页面的"第二个设备"右侧的下拉菜单中选择"CD-ROM"。
- 3. 勾选"附加CD"。
- 4. 勾选"启用引导菜单"。

			>
集群		Default	~
		数据中心:Default	
模板		Blank (0)	~
操作系统		Other OS	~
实例类型	63	自定义	~
优化日标		服务器	~
引导序列: 第一个设备		価色	Y
第二个设备		CD-ROM	~
✓ 附加 CD		openEuler-22.03-LTS-SP2-aarch64-dv 🗸	e
☑ 启用引导菜单			
	集群 侯板 操作系統 实例类型 优化目标 引导序列: 第一个设备 第二个设备 配 附加 CD 【 応用引导菜单	 集群 模板 操作系统 实例类型 次例类型 优化目标 引导序列: 第一个设备 第二个设备 第二个设备 第二个设备 第二个设备 第二个设备 第二个设备 	集群 Default 数据中心: Default 数据中心: Default 模板 Blank [(0) 操作系统 Other OS 实例类型 63 自定义 优化目标 服务器 引导序列: 第一个设备 硬盘 第二个设备 CD-ROM 管 附加 CD openEuler-22.03-LTS-SP2-aarch64-dv ~ 自用引导菜单

- **步骤7** (可选)按需配置自定义属性qemucmdline,详细配置操作请参见《鲲鹏BoostKit虚 拟化使能套件 故障案例》中的如何在oVirt创建虚拟机时自定义属性。
- **步骤8** 配置完成后单击"确定"完成虚拟机创建。右上角单击"运行"启动虚拟机并安装系统。

	ξ	2-	11 运行	F虚打	以机											
ł	+貫 >	虚拟机														
	Vms:					x 🟠	~ Q				新建编辑	▶运行 ∨	€ 暂停	导出 ■关闭 ~	C 重启 ↓ 控	月台 -
ĺ			名称	注释	主机	IP 地址	FQDN	集群	数据中心	内存	CPU	网络	图形界面	状态	上线时间	描述
			vm1					Default	Default				无	Down		

图形化安装系统需要打开控制台,单击"控制台"会自动下载名为"console.vv"的文件,可使用Remote Viewer或VNC客户端打开此文件。

----结束

2.10 故障排除

2.10.1 编译 FreeIPA 失败的解决方法

问题现象描述

编译FreeIPA软件包时提示"Installed (but unpackaged) file(s) found: /usr/share/ selinux/packages/targeted/ipa.pp.bz2"。具体截图如下:

conflicts: freeipa-python < 4.2.91 ipa-common ipa-python < 4.9.3	
Obsoletes: ipa-common < 4.9.3	
Processing files: freeipa-debuginfo-4.9.3-4.aarch64	
Provides: freeipa-debuginfo = 4.9.3-4 freeipa-debuginfo(aarch-64) = 4.9.3-4	
Requires(rpmlib): rpmlib(CompressedFileNames) <= 3.0.4-1 rpmlib(FileDigests) <= 4.6.0-1 rpmlib(PayloadFilesHavePrefix) <= 4.	0-1
Recommends: freeipa-debugsource(aarch-64) = 4.9.3-4	
Processing files: freeipa-debugsource-4.9.3-4.aarch64	
Provides: freeipa-debugsource = 4.9.3-4 freeipa-debugsource(aarch-64) = 4.9.3-4	
Requires(rpmlib): rpmlib(CompressedFileNames) <= 3.0.4-1 rpmlib(FileDigests) <= 4.6.0-1 rpmlib(PayloadFilesHavePrefix) <= 4. Checking for unpackaged file(s): /usr/lib/rpm/check-files /root/rpmbuild/BUILDROOT/freeipa-4.9.3-4.aarch64	θ-1
erof: Installed (but unpackaged) file(s) found:	
/usr/share/sellnux/packages/targeted/lpa.pp.bz2	
PDM huild errors	
Month efforts.	
Macro expanded in comment on time os. s{:python_disable_dependency_generator}	
Macro expanded in comment on line 125: %{krb5_kdb_version}	
Macro expanded in comment on line 126: %{krb5_kdb_version}	
Installed (but unpackaged) file(s) found: /usr/share/selinux/packages/targeted/ipa.pp.bz2	

关键过程、根本原因分析

rpm-build软件包与FreeIPA软件包的SPEC文件存在兼容性问题。

结论、解决方案及效果

步骤1 修改"freeipa/freeipa.spec"文件,分别将图示这几行注释掉。

%if %{without seli	nux}					
# SELinux subpacka	ge					
%package selinux						
Summary:	FreeIPA SELinux policy					
BuildArch:	noarch					
Requires:	selinux-policy-%{selinuxtype}					
Requires(post):	selinux-policy-%{selinuxtype}					
%{?selinux require	s}					
%description selin Custom SELinux pol	ux icy module for FreeIPA					
%description selin Custom SELinux pol # with selinux %endif	ux icy module for FreeIPA					
%description selin Custom SELinux pol # with selinux %endif 1519 %if %{without	ux icy module for FreeIPA selinux}					
%description selin Custom SELinux pol # with selinux %endif 1519 %if %{without 1520 %files selinux	ux icy module for FreeIPA <mark>selinux</mark> }					
%description selin Custom SELinux pol # with selinux %endif 1519 %if %{without 1520 %files selinux 1521 %{_datadir}/se	ux icy module for FreeIPA <mark>selinux</mark> } <mark>linux</mark> /packages/%{ <mark>selinux</mark> type}/%{modulename}.pp.*					
<pre>%description selin Custom SELinux pol # with selinux %endif 1519 %if %{without 1520 %files selinux 1521 %{_datadir}/se 1522 %ghost %{_shar</pre>	ux icy module for FreeIPA selinux} linux/packages/%{selinuxtype}/%{modulename}.pp.* edstatedir}/ <mark>selinux</mark> /%{selinuxtype}/active/modules/200/%{modulename;					
<pre>%description selin Custom SELinux pol # with selinux %endif 1519 %if %{without 1520 %files selinux 1521 %{_datadir}/se 1522 %ghost %{_shar 1523 # with selinux</pre>	ux icy module for FreeIPA selinux} linux/packages/%{selinuxtype}/%{modulename}.pp.* edstatedir}/selinux/%{selinuxtype}/active/modules/200/%{modulename}					

步骤2 重新编译FreeIPA软件包。

----结束



发布日期	修订记录
2025-02-20	第三次正式发布。 刷新《 <mark>鲲鹏oVirt轻量级虚拟化管理平台 部署指南</mark> (openEuler 20.03)》。
2024-03-08	第二次正式发布。 新增《 <mark>鲲鹏oVirt轻量级虚拟化管理平台 部署指南</mark> (openEuler 22.03)》。
2022-03-31	第一次正式发布。