

巍巍交大 百年书香
www.jiaodapress.com.cn
bookinfo@sjtu.edu.cn



丛书策划 张荣昌
责任编辑 王 清 孟海江
封面设计 唐韵设计

新时代计算机人才培养系列教材

大数据

大数据基础
大数据采集与预处理技术
大数据集群搭建维护与数据存储
大数据采集与数据处理
Hadoop应用与开发
数据可视化技术与应用
大数据分析技术与应用
数据挖掘技术与应用

云计算

云计算基础
私有云基础架构与运维
公有云服务架构与运维
云计算基础架构平台应用
云平台配置与管理
云安全技术应用
云网络技术应用
云计算运维开发
云计算应用开发

人工智能

人工智能应用基础
人工智能数学基础
人工智能数据服务
计算机视觉应用开发
深度学习应用开发
机器学习应用开发

自然语言处理及应用开发
智能语音处理及应用开发
人工智能系统部署与运维
人工智能综合项目开发

区块链

区块链应用基础
区块链核心技术
区块链部署与运维
区块链应用设计与开发
区块链项目综合实践
智能合约开发

物联网

物联网基础
物联网工程导论
物联网嵌入式技术
物联网设备装调与维护
物联网系统部署与运维
物联网工程设计与实施
物联网终端智能应用开发基础案例教程
(基于HAL/LL库)

信息安全

信息安全基础
信息安全标准与法规
信息安全工程与管理
信息安全风险评估

密码产品部署与应用
电子商务安全技术
无线网络安全技术
大数据安全技术
电子数据取证技术
终端数据安全及防泄密
数据存储与容灾
Web应用安全与防护

专业基础

数据库应用技术(基于达梦数据库)
MySQL数据库应用技术项目教程
计算机网络基础
Linux系统管理与服务器配置
Linux服务器安全高级运维
Windows Server操作系统基础
现代通信原理
无线传输技术
自动识别应用技术
智能传感与检测技术
路由交换技术项目实战教程
虚拟化技术基础

新时代计算机人才培养系列教材
“互联网+”新形态一体化教材

云计算基础架构平台应用

主编◎来继敏 张乃月 宋学永



上海交通大学出版社

新时代计算机人才培养系列教材
“互联网+”新形态一体化教材

云计算基础架构
平台应用

主编◎来继敏 张乃月 宋学永



上海交通大学出版社
SHANGHAI JIAO TONG UNIVERSITY PRESS

本书提供教学资源包

网址: <https://www.sjhtbook.com>



新时代计算机人才培养系列教材
“互联网+”新形态一体化教材

云计算基础架构 平台应用

主编◎来继敏 张乃月 宋学永



上海交通大学出版社
SHANGHAI JIAO TONG UNIVERSITY PRESS

内容提要

本书精选“1+X”职业技能考试和高职院校“云计算应用”赛项项目案例，以实用为主，结构安排上更加符合读者的认知习惯。全书共十四个项目，其中项目一为基础内容，介绍云计算基础架构平台实施的基础系统；项目二至项目七，带领读者安装和配置 OpenStack 平台的基础服务组件；项目八至项目十，介绍平台的基本使用，包括创建和管理网络、云主机和镜像；项目十一和项目十二，扩展 Cinder 块存储服务和 Swift 对象存储服务；项目十三和项目十四，基于 Python 语言完成云计算基础架构平台的访问，包括查询和其他操作。本书可作为高职院校云计算技术应用等相关专业学习私有云技术的教材，也适合相关行业从业人员阅读与参考。

图书在版编目（CIP）数据

云计算基础架构平台应用 / 来继敏，张乃月，宋学永主编. — 上海：上海交通大学出版社，2024. 12
ISBN 978-7-313-31766-7
I. TP393.027
中国国家版本馆 CIP 数据核字第 2024H1C327 号

云计算基础架构平台应用

YUNJISUAN JICHU JIAGOU PINGTAI YINGYONG

主 编：来继敏 张乃月 宋学永	地 址：上海市番禺路 951 号
出版发行：上海交通大学出版社	电 话：021-6407 1208
邮政编码：200030	
印 制：北京荣玉印刷有限公司	经 销：全国新华书店
开 本：889 mm × 1194 mm 1/16	印 张：13
字 数：360 千字	
版 次：2024 年 12 月第 1 版	印 次：2024 年 12 月第 1 次印刷
书 号：ISBN 978-7-313-31766-7	
定 价：45.00 元	

版权所有 侵权必究

告读者：如发现本书有印装质量问题请与印刷厂质量科联系

联系电话：010-6020 6144



前言

随着 IT 技术的飞速发展，云计算已经成为不可或缺的技术力量。2012 年中华人民共和国科学技术部发布《中国云科技发展“十二五”专项规划》，对云计算相关技术做出规划，后又经历了“十三五”的夯实基础，到“十四五”的阶段性的发展。2021 年的“十四五”规划中，数字中国建设被提升到新的高度，云计算成为重点发展产业。云计算不仅改变了处理数据的方式，还重塑了 IT 基础设施的构建和管理模式。云计算基础架构平台作为云计算的核心部分，提供了高效、灵活、可扩展的计算服务，助力企业和组织快速适应变化，降低成本，提高生产力。

本教材的编写全面融入“1+X”云计算平台运维与开发职业技能标准和全国职业院校技能大赛高职组“云计算应用”赛项的私有云部分的内容，编写目的在于帮助读者全面了解云计算基础架构平台的核心概念、各组件的安装和配置、云主机的创建，以及使用 Python 进行自动化运维的方法。教材从基础知识入手，详细介绍了云计算基础架构平台的构成、工作原理和关键技术。本教材内容涵盖了基础操作系统的准备、基础环境的配置，以及 Keystone、Glance、Nova、Neutron、Horizon 等基础服务的安装和配置；讲解了云主机和镜像的管理方法；扩展了 Cinder 块存储服务和 Swift 对象存储服务，以及使用 Python 通过 OpenStack SDK 和 OpenStack API 访问和操作 OpenStack 平台的方法。

本教材具有以下特色。

1. 基于“育人为先、德技并修”的育人理念

本教材以习近平新时代中国特色社会主义思想为指导，挖掘任务内容与思政元素的“触点”和“融点”，设计“技能知识”与“课程思政”两条主线，强化课程思政的职业素养、科技意识、工匠精神、家国情怀的三维内涵，构建思政教育与专业课程同向同行的体系，实现以匠心育匠才，使课程思政教学与专业课程教学有机融合，全面贯彻党的二十大精神，落实立德树人根本任务。

2. 突出“校企合作、项目育人”的育人模式

本教材为河北省职业教育提质培优行动计划（2020—2023 年）建设项目“校企二元合作开发的职业教育规划教材”，由河北软件职业技术学院、南京第五十五所技术开发有限公司及江苏一道云科技发展有限公司合作编写，编写过程中得到了南京第五十五所技术开发有限公司和江苏一道云科技发展有限公司的大力支持。

3. 体现“赛证”融通的开发理念

本教材的主编均具有多年指导“1+X”职业技能考试和“云计算应用”大赛的经验，教材内容结合了主偏多年指导“1+X”考试和“云计算应用”赛项的心得经验。我们衷心希望本教材能为读者在学习和实践云计算基础架构平台的过程中提供有益的指导和帮助。同时，我们也期待读者在学习过程中能够不断思考和创新，为我国的中小企业发展内部私有云贡献力量，为我国乃至世界的云计算技术的发展和应用做出贡献。

在编写过程中，我们力求内容的系统性和实用性，以帮助读者快速掌握核心知识，学会搭建和使

用 OpenStack 云计算基础架构平台。此外，我们还为读者提供了丰富的参考资料和在线资源，以满足不同层次的学习需求，有需要者可发邮件至 2393867076@qq.com 领取。

作为一本专门讲解云计算基础架构平台的教材，本教材适合云计算技术应用、计算机科学与技术、软件工程、电子商务等相关专业的学生学习使用，同时也适合对云计算技术感兴趣的读者或相关行业从业者阅读与参考。通过学习本教材，读者将能够更好地掌握使用 OpenStack 搭建云计算基础架构平台的方法，提升在云计算领域的专业素养和实践能力。

由于时间仓促，加之作者水平有限，本书存在的不足和疏漏之处，敬请各位读者批评指正。

编 者

2024 年 6 月



目录



项目一 OpenStack 安装前准备

任务一 认识云计算与OpenStack	2
一、云计算与云计算基础架构平台	2
二、OpenStack 的发展	3
三、OpenStack 的版本	3
任务二 安装OpenStack所需的操作系统	4
一、系统架构	4
二、安装 openEuler	4
任务三 配置OpenStack私有仓库	14
一、配置 repository 服务器的 OpenStack 私有仓库	14
二、配置 controller 和 compute 的 yum 源	18



项目二 基础服务

任务一 安装和配置OpenStack基础环境	22
一、安装 OpenStack 基础环境	22
二、配置基础环境变量	23
三、部署 OpenStack 云平台基础环境	24
四、配置 repository 服务器作为时间同步服务器	25
任务二 安装和配置数据库和消息队列软件	26
一、了解 MariaDB	26
二、了解 RabbitMQ	27
三、安装和配置 MariaDB 和 RabbitMQ	27



项目三 身份认证服务 Keystone

任务一 了解Keystone	31
一、Keystone 架构	31
二、Keystone 服务内容	32
任务二 安装和配置Keystone	33
一、在控制节点上安装和配置 Keystone 服务	33
二、安装配置内容分析	33



项目四 镜像服务 Glance

任务一 了解Glance	40
一、Glance 的基本概念	40
二、Glance 支持的镜像格式	40
三、Glance 管理的镜像数据类型	41
任务二 认识Glance的基本组件及工作流程	41
一、Glance 基本组件	41
二、Glance 的基本工作流程	42
任务三 安装和配置Glance	43
一、在控制节点上安装和配置 Glance 服务	43
二、安装脚本内容分析	43
三、验证 Glance 服务	46



项目五 计算服务 Nova

任务一 了解Nova	50
一、Nova 的基本概念	50
二、Nova 的功能	50
任务二 认识Nova的基本组件和工作流程	50
一、Nova 的基本组件	50
二、Nova 的基本工作流程	51
任务三 安装和配置Nova	52
一、在控制节点上安装与配置 Nova 服务	52
二、在计算节点上安装与配置 Nova 服务	59



项目六 网络服务 Neutron

任务一 了解Neutron	65
一、Neutron 的基本功能	65
二、Neutron 的网络类型	65
任务二 认识Neutron组件架构与工作流程	66
一、Neutron 组件	66
二、Neutron 组件架构与工作流程	66
任务三 安装和配置Neutron	67
一、在控制节点上安装与配置 Neutron 网络服务	67
二、在计算节点上安装与配置 Neutron 网络服务	71
三、检测 Neutron 网络服务	74



项目七 管理界面 Horizon

任务一 了解Horizon	77
一、Horizon 服务	77
二、Horizon 的基本功能	77
任务二 认识Horizon组件与工作流程	78
一、Horizon 的组件	78
二、Horizon 的工作流程	79
任务三 安装和配置Horizon	79
一、在控制节点上安装与配置 Horizon 服务	80
二、安装脚本分析	80
任务四 验证Horizon服务	82
一、登录系统	83
二、查看概况页面	83



项目八 虚拟机与外部网络通信

任务一 通过图形化界面创建网络和路由	87
一、登录 Dashboard 管理界面	87
二、配置外网	87
三、配置内网	89
四、配置路由	90

五、查看网络拓扑	92
任务二 通过命令行创建网络和路由	92
一、加载环境变量信息	92
二、检查 Neutron 代理信息	92
三、创建外部网络	93
四、创建外部网络子网	94
五、创建内部网络	95
六、创建内部网络子网	95
七、查看网络和子网	96
八、创建路由	96



项目九 虚拟机实例

任务一 通过图形化界面创建虚拟机实例	101
一、登录 Dashboard 管理界面	101
二、创建镜像	101
三、创建实例类型	102
四、创建安全组和规则	104
五、创建云主机实例	105
六、绑定浮动 IP	108
七、测试网络连通性	111
任务二 通过命令行创建虚拟机实例	112
一、通过工具上传镜像文件	112
二、创建镜像	112
三、创建实例类型	114
四、创建网络和子网	115
五、创建路由	115
六、创建安全组和规则	115
七、创建云主机实例	117
八、浮动 IP 地址管理	119
九、测试网络连通性	120
十、查看命令行创建的内容	121



项目十 OpenStack 镜像

任务一 创建OpenStack镜像	127
--------------------------------	------------

任务二 管理OpenStack镜像	127
一、查看镜像	127
二、修改镜像的属性	127
三、镜像下载	128
四、镜像的删除	129



项目十一 Cinder 块存储服务

任务一 理解Cinder块存储服务	132
一、Cinder 的主要功能	132
二、Cinder 与 Nova 的交互	132
三、Cinder 架构与创建卷的基本流程	132
四、块存储组件	134
任务二 安装Cinder块存储服务	134
一、在存储节点上准备磁盘	134
二、在控制节点上安装和配置块存储服务	135
三、在计算节点上安装和配置块存储服务	138
任务三 创建与使用Cinder块存储	140
一、在 Dashboard 中为服务器扩展磁盘	141
二、在终端中为服务器扩展磁盘	143



项目十二 Swift 对象存储服务

任务一 了解Swift对象存储系统	149
一、Swift 对象存储系统	149
二、Swift 对象层次数据模型	149
三、对象层次数据模型与对象存储 API 的交互	150
四、对象存储的组件和架构	150
任务二 安装和配置Swift对象存储服务	152
一、在控制节点上安装和配置对象存储服务	152
二、在计算节点上安装和配置对象存储服务	158
任务三 测试Swift对象存储服务	162
一、在 Dashboard 中测试	162
二、在终端中测试	164



项目十三 使用 OpenStack SDK 创建虚拟机实例

任务一 安装Python开发环境	169
一、安装 Python 3 解释器	169
二、安装 OpenStack SDK 库	170
三、设置 OpenStack 域名的对应 IP	172
四、安装和设置开发工具	172
任务二 掌握OpenStack SDK的基本用法	177
任务三 使用OpenStack SDK创建云主机	180



项目十四 使用 OpenStack API 创建虚拟机实例

任务一 访问OpenStack的API	186
任务二 通过OpenStack API创建云主机	189
参考文献	195

项目一

OpenStack 安装前准备

知识目标 >

- (1) 了解云计算的分类。
- (2) 了解云计算基础架构平台。
- (3) 掌握 OpenStack 基础环境配置要求。

能力目标 >

- (1) 学会 OpenStack 基础环境的安装方法。
- (2) 学会私有仓库的配置方法。

素质目标 >

- (1) 培养独立思考能力。
- (2) 培养自学能力。
- (3) 培养严谨的逻辑思维。

项目描述 >

在本项目中，首先带领读者了解云计算技术分类及用到的技术栈，然后在学习 OpenStack 私有云部署时，掌握基础环境所需服务器的安装与配置，了解 OpenStack 私有云及其版本，学会使用 openEuler 服务器安装与配置 OpenStack Controller，使用 openEuler 服务器安装与配置 OpenStack Compute，以及使用 openEuler 服务器安装与配置 OpenStack 所需的私有仓库。通过掌握这些内容，为后续项目奠定基础。

任务一 认识云计算与OpenStack

我国的云计算经过多年的蓬勃发展取得了前所未有的成就，其中阿里云在 2023 年全球的市场份额中排名第四。目前国内云计算厂商较多，比较大的有阿里云、华为云、腾讯云、百度云等。我国的云计算作为信息产业的基础设施，在全球范围内占据了举足轻重的地位。云计算基础架构平台应用有很多种，其中 OpenStack 为常用的平台应用之一。在任务一中，将对 OpenStack 进行初步了解，掌握 OpenStack 的历史、特点、作用、版本等内容。

一、云计算与云计算基础架构平台

2012 年，我国发布了《中国云科技发展“十二五”专项规划》，对我国云计算的发展作出了前瞻性的规划，此后国家多次发布云计算发展的相关文件。国家政策支持极大地促进了我国云计算技术的发展，云计算已经成为 IT 信息行业的基础服务。

2021 年国务院印发的《“十四五”数字经济发展规划》中，再次强调了云计算作为数字经济的基础作用，大力推进“企业上云”。党的二十大报告中也提到了“推进新型工业化，加快建设制造强国、质量强国、航天强国、交通强国、网络强国、数字中国”，其中的数字中国的基础就是云计算，云计算是我国进入现代化的强力支撑。

云计算是继计算机、互联网后信息时代的又一种革新，是信息时代的一个大飞跃。各大厂商对云计算的定义各有不同，但概括来讲，其含义基本是一致的，即云计算具有很强的扩展性和需要性，可以为用户提供一种全新的体验，云计算的核心是可以将很多的计算机资源协调在一起，为用户提供服务。云计算使得用户通过网络就可以获取到无限的资源，同时获取的资源不受时间和空间限制。

云计算平台（云平台）从广义上来划分，可以分为 IaaS、PaaS 和 SaaS 三种。其中，IaaS 表示基础设施即服务，此类云平台提供给用户的产品就像一台标准的物理机器，用户可以选择机器的性能，如 CPU（中央处理器）、内存、硬盘等参数，这台机器上除了用户选择的操作系统，什么都没安装，用户可以随意安装需要的软件、硬件和网络资源，且能够随时升级；PaaS 表示平台即服务，这类云平台提供给用户的是一个现成的运行环境，用户在使用时只需要上传自己应用的代码即可使用，不用自己维护运行软件所需要的基础软件，如 Web Server、数据库等；SaaS 表示软件即服务，此类云平台在云上为终端用户直接提供特定应用类的软件服务。

目前提供云计算基础架构平台应用的软件主要有 OpenStack 和 Apache CloudStack，但在开源市场，主要应用还是 OpenStack，OpenStack 为用户提供最直接的服务就是 IaaS。比较早的大型云计算公司多采用自己的基础架构平台，像亚马逊、谷歌、微软和国内的阿里云，均采用自研的基础架构平台。阿里云在 2009 年初研发成功的“飞天”超大规模通用计算操作系统，并没有采用 OpenStack。而起步较晚的云计算服务商，为了快速追赶行业龙头，多采用 OpenStack 作为云计算基础架构平台，像国内的腾讯云、华为云、电信云等均采用 OpenStack 基础架构平台。学习 OpenStack 相关技术可以更好地为国家云计算产业做出贡献。

二、OpenStack 的发展

OpenStack 是由美国国家航空航天局（NASA）与 Rackspace 公司合作研发并发起的，是以 Apache 许可证授权的自由软件和开放源代码的云计算技术解决方案，主要用于实现云项目，作为云项目操作系统而存在。

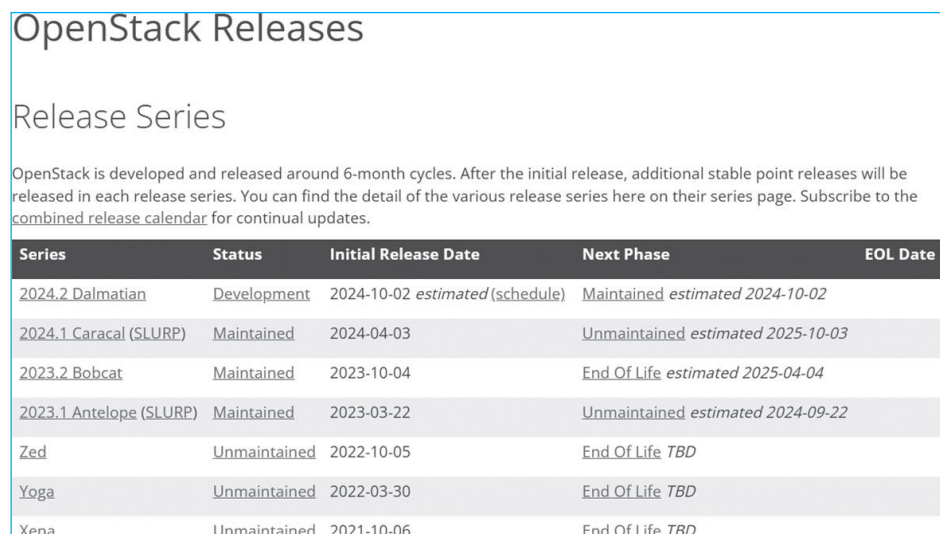
OpenStack 主要包括以下几个子项目：计算服务 OpenStack Compute（Nova）、云对象存储 Cloud Object Storage（Swift）、镜像管理（Glance）、身份识别 Identity（Keystone）、网络连接管理 Network Connectivity（Quantum）、Web 管理界面（Dashboard）等。

目前有超过 150 家公司参与了 OpenStack 项目，包括国外的 HP、Dell、AMD、Intel、Cisco、Citrix 等公司，国内的中国移动、中国银联、新浪、华胜天成、H3C 等公司。此外，微软在 2010 年 10 月表示支持 OpenStack 与 Windows Server 2008 R2 的整合，而 Ubuntu 在 11.04 版本中已开始集成 OpenStack。OpenStack 是目前最受关注与支持的开源云计算平台之一。

OpenStack 用 Python 编写，这意味着相比其他以 C/C++ 或 Java 编写的开源云计算平台，OpenStack 更容易修改与调试。OpenStack 整合了 Tornado 网页服务器、Nebula 运算平台，使用 Twisted 框架，支持的虚拟机宿主包括 KVM、XEN、VirtualBox、QEMU、LXC 等。

三、OpenStack 的版本

OpenStack 的开发和发布周期约为 6 个月。在首次发布之后，每个发布周期都将发布其稳定版本。读者可以在官网系列页面上找到各种发布系列的详细信息，如图 1-1 所示。



OpenStack Releases				
Release Series				
OpenStack is developed and released around 6-month cycles. After the initial release, additional stable point releases will be released in each release series. You can find the detail of the various release series here on their series page. Subscribe to the combined release calendar for continual updates.				
Series	Status	Initial Release Date	Next Phase	EOL Date
2024.2 Dalmatian	Development	2024-10-02 <i>estimated (schedule)</i>	Maintained	<i>estimated 2024-10-02</i>
2024.1 Caracal (SLURP)	Maintained	2024-04-03	Unmaintained	<i>estimated 2025-10-03</i>
2023.2 Bobcat	Maintained	2023-10-04	End Of Life	<i>estimated 2025-04-04</i>
2023.1 Antelope (SLURP)	Maintained	2023-03-22	Unmaintained	<i>estimated 2024-09-22</i>
Zed	Unmaintained	2022-10-05	End Of Life	<i>TBD</i>
Yoga	Unmaintained	2022-03-30	End Of Life	<i>TBD</i>
Xena	Unmaintained	2021-10-06	End Of Life	<i>TBD</i>

图 1-1 OpenStack 官网发布版本

OpenStack 的版本系列采用首字母为字母表 A 到 Z 顺序的单词进行命名，如其第一个版本名称为 Austin，2010 年 10 月发行；第二个版本名称为 Bexar，2011 年 2 月发行；以此类推，截至本书稿完成（2024 年 6 月）其最新稳定版本为 Caracal，发行时间为 2024 年 4 月。其下一个版本为 Dalmatian，预计发行时间为 2024 年 10 月。

OpenStack 的 Train 版本是企业广泛使用的稳定版本，因此本教材采用 Train 版本进行编写，所有操作均在 Train 版本下完成。

任务二 安装OpenStack所需的操作系统

在本任务中，完成运行 OpenStack 所需的操作系统 openEuler 的安装，本教材采用 openEuler 的版本为 22.03，读者需要注意安装的注意事项。

一、系统架构

一个基本的 OpenStack 平台运行至少需要两台服务器：一台为 controller（控制节点），用来运行 OpenStack 必需的 MariaDB（数据库服务）、Rabbit（消息队列服务）、Keystone（认证服务）、Glance（镜像服务）、Nova（计算服务）等组件；另外一台为 compute（计算节点），用来运行 OpenStack 必需的 Glance、Nova 等组件。

由于安装时，组件需要从互联网服务器下载，因此网速问题会影响安装，可以采用安装第三台服务器作为 repository（本地 yum 仓库）的方法，为另外两台服务器安装提供私有的 yum 源仓库。

二、安装 openEuler

1. 前期准备

为了便于读者学习，本教材将在 VMware workstation 17 中使用虚拟机安装基础操作系统，操作系统选择 openEuler 22.03，读者可以从教材的资源中自行下载 iso 光盘镜像文件（openEuler-22.03-LTS-SP2-x86_64-dvd.iso），也可以从华为官方网站进行下载。

需要安装 3 台服务器，分别完成 controller、compute 和 repository 功能，具体配置参数如表 1-1 所示，后续将根据表中的配置来完成 openEuler 的安装。

表 1-1 服务器配置参数

服务器	VMware 中的虚拟机名字	主机名	内存 /GB	VCPU	磁盘 /GB	网卡数量及类型
controller	openstack train controller 2nic	controller	9	6	60	1NAT+1 仅主机
compute	openstack train compute 2nic	compute	3	6	60	1NAT+1 仅主机
repository	openstack train repo	repository	2	2	60	1NAT+1 仅主机

2. 安装三台服务器

三台服务器的安装方法相同，在安装前需要注意 VMware 中服务器的配置需要根据表 1-1 进行，本任务以 controller 服务器为例进行讲解，compute 和 repository 服务器的安装可以参考 controller 服务器的安装步骤进行，这里不再赘述。下面开始 controller 服务器的安装。

在 VMware workstation 中添加虚拟机，在添加虚拟机向导界面中，选择“自定义（高级）（C）”，如图 1-2 所示。

继续单击“下一步（N）”两次，在向导中选择“稍后安装操作系统（S）”，如图 1-3 所示。

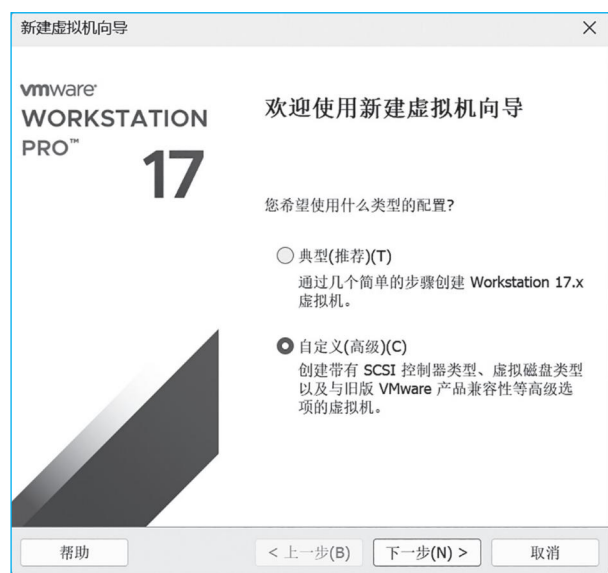


图 1-2 选择“自定义（高级）(C)”



图 1-3 选择“稍后安装操作系统 (S)”

继续单击“下一步 (N)”，在向导中选择要安装的操作系统及版本，客户机操作系统选择“Linux (L)”，版本选择“其他 Linux 5.x 内核 64 位”，如图 1-4 所示。

单击“下一步 (N)”按钮，在向导中可以选择操作系统要安装的位置，本教材将把用到的三个系统安装在“D:\openstack\”目录下。在安装目录下创建文件夹“openstack train controller 2nic”，选择该目录作为安装位置，虚拟机名称亦为“openstack train controller 2nic”，如图 1-5 所示。



图 1-4 操作系统及版本选择



图 1-5 虚拟机位置及名称

单击“下一步 (N)”按钮，输入每个处理器 (CPU) 内核数量为 6，如图 1-6 所示。

继续单击“下一步 (N)”，分配内存大小为 9 GB (9216 MB)，如图 1-7 所示。

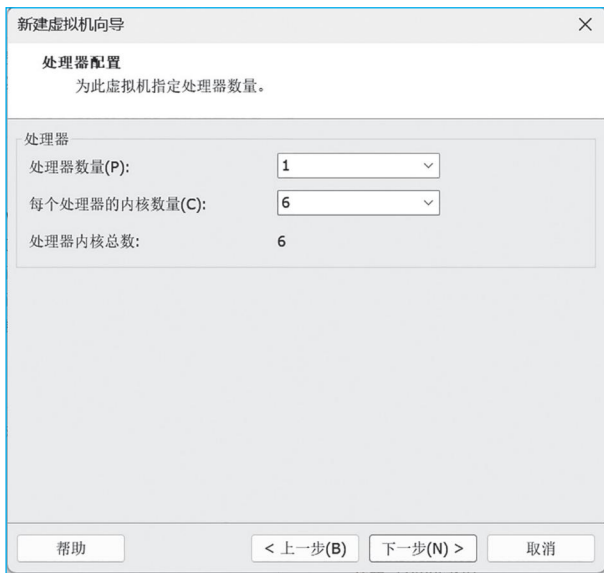


图 1-6 每个处理器的内核数量设置



图 1-7 内存设置

继续单击“下一步（N）”，在向导中设置网络类型，选择 NAT 模式，这是在 controller 中要使用的第一块网卡（网络适配器），如图 1-8 所示。

继续单击四次“下一步（N）”，在 I/O 控制器类型、磁盘类型选择及选择磁盘向导中，均使用默认设置，在指定磁盘容量向导中，指定磁盘大小为 60 GB，其他使用默认内容，如图 1-9 所示。



图 1-8 网络类型设置

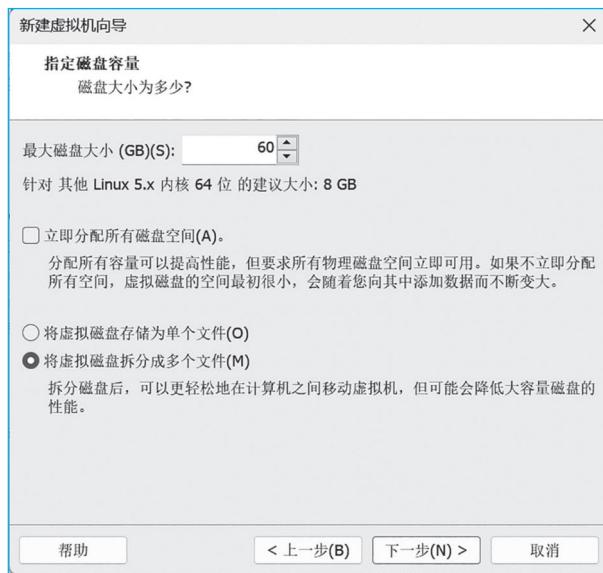


图 1-9 指定磁盘容量

单击“下一步（N）”，在指定磁盘文件向导中，使用默认名称，如图 1-10 所示；再次单击“下一步（N）”，单击“完成（D）”按钮，即可完成基本设置。

完成基本设置后，如果需要调整虚拟机硬件设置，可以单击“编辑虚拟机设置”，如图 1-11 所示。

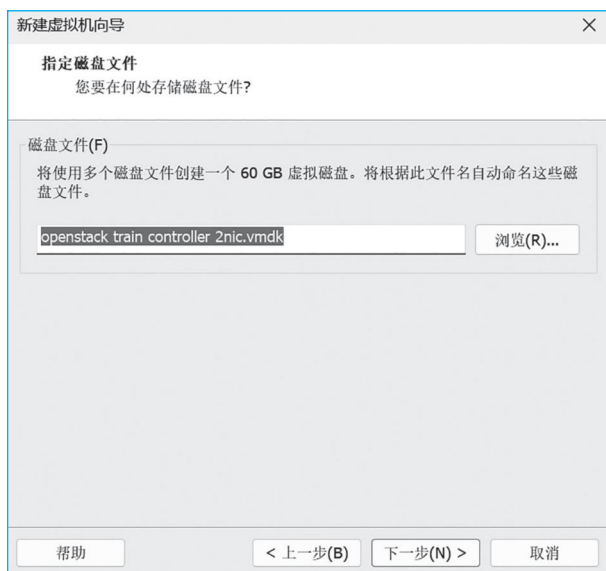


图 1-10 指定磁盘文件

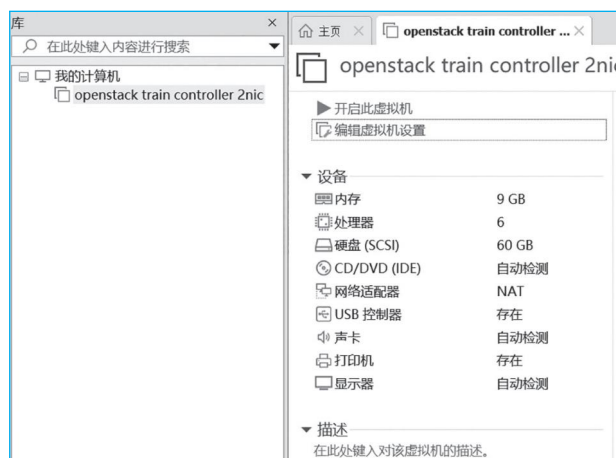


图 1-11 编辑虚拟机设置

如图 1-12 所示，可以在“虚拟机设置”中添加和修改对应的虚拟机设置项。下面我们来完成第二块网卡的添加和设置。

在图 1-12 中，单击“添加 (A) ...”按钮，打开添加硬件向导子窗口，如图 1-13 所示。

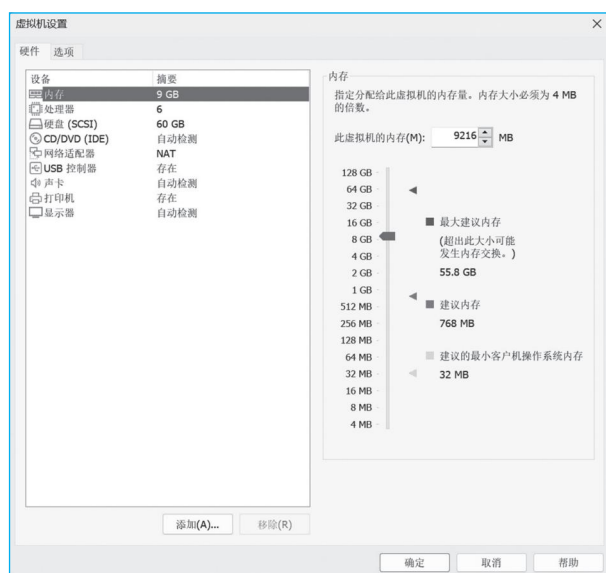


图 1-12 虚拟机设置向导

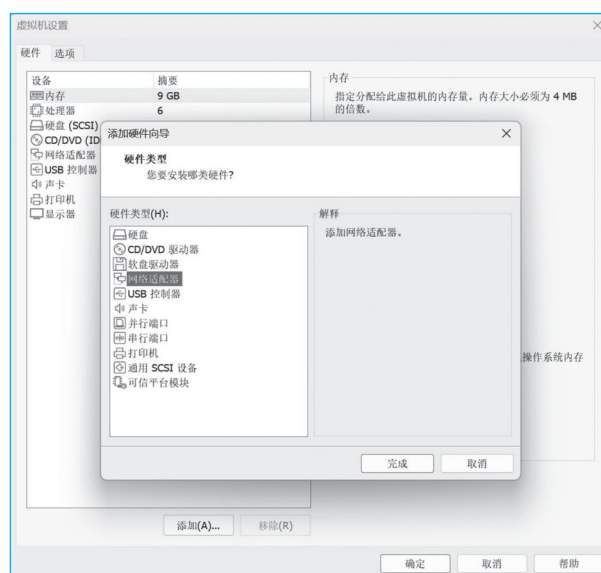


图 1-13 添加硬件向导

选中“网络适配器”，单击“完成”按钮，关闭子窗口，可以看到在父窗口中，添加了第二块网卡。选中添加的第二个网络适配器，修改网络类型为“仅主机模式 (H)”，如图 1-14 所示。

至此，完成了第二块网卡的添加和设置，下面继续完成 CPU 的设置。选中左侧硬件中的“处理器”，在右侧的“虚拟化引擎”中，选择“虚拟化 Intel VT-x/EPT 或 AMD-V/RVI (V)”，单击“确定”按钮完成 CPU 的设置，如图 1-15 所示。

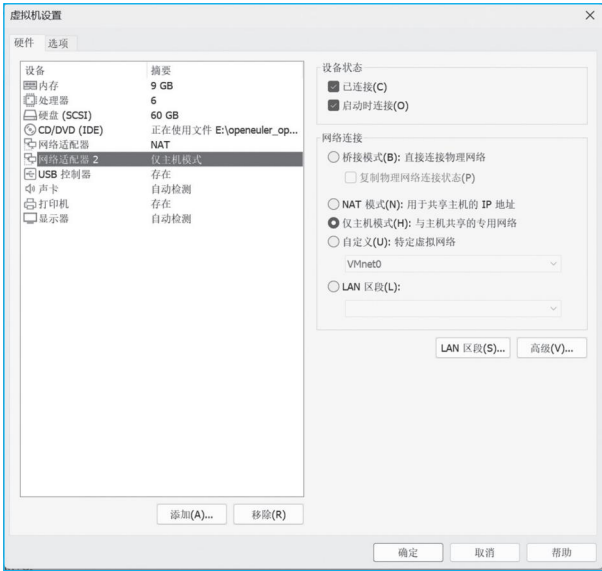


图 1-14 网络适配器 2 设置

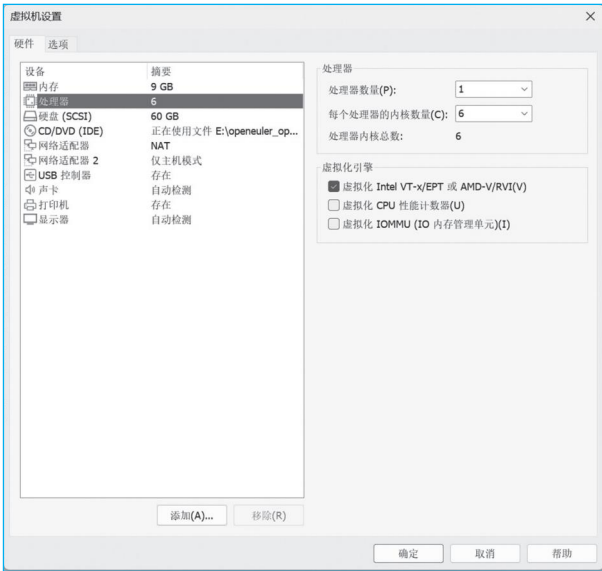


图 1-15 CPU 虚拟化设置

再次选择硬件区的“CD/DVD（IDE）”，然后在右侧的链接区域选择“使用 ISO 映像文件（M）”，单击“浏览（B）”按钮，选择下载的 openEuler-22.03-LTS-SP2-x86_64-dvd.iso 文件，如图 1-16 所示。

单击“确定”按钮，完成虚拟机的设置。单击“开启此虚拟机”即可开始 openEuler 操作系统的安装，如图 1-17 所示。

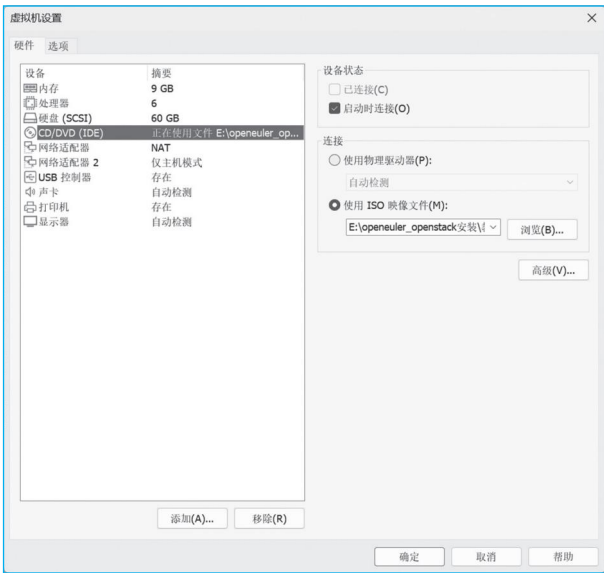


图 1-16 选择 openEuler 映像文件



图 1-17 安装 openEuler 操作系统

直接选择“Install openEuler 22.03-LTS-SP2”开始安装，此时开始自动安装，经过一段时间的自动处理，安装进入如图 1-18 所示的界面。选择中文作为操作系统的基础语言，单击“继续（C）”按钮，继续安装。

根据安装信息摘要界面最底部的提示（图 1-19），需要设置系统的“安装目的地”和“Root 账户”，单击“安装目的地（D）”，开始设置安装目的地，如图 1-20 所示。

选择唯一的 60 GB 磁盘，在“存储配置”区域的分区中，我们选择“自定义（C）”，然后单击“完成（D）”按钮，开始配置分区，如图 1-21 所示。



图 1-18 选择语言支持



图 1-19 安装摘要



图 1-20 选择安装位置



图 1-21 配置分区

在图 1-21 中，选择“点击这里自动创建它们（C）”，自动创建各个分区，然后删除不必要的分区，并调整“/”分区的大小到最大，修改设备类型为“标准分区”，文件系统选择“xfs”，如图 1-22 所示。

单击两次“完成（D）”按钮后，在“更改摘要”的窗口中，会提示修改过的内容摘要，如图 1-23 所示。

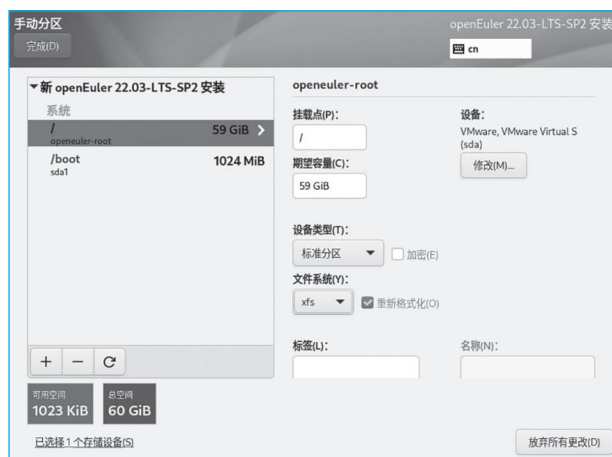


图 1-22 调整分区

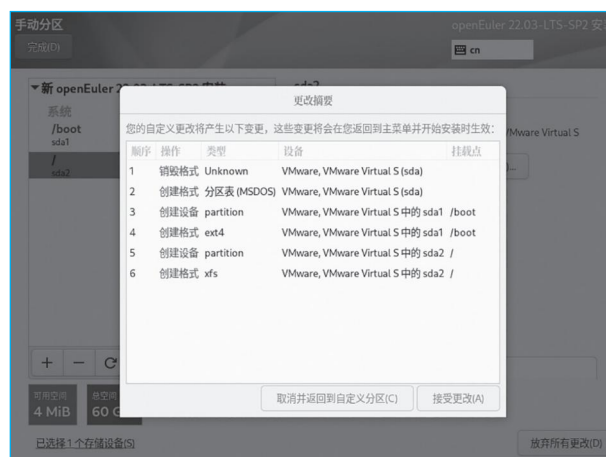


图 1-23 修改过的内容摘要

单击“接受更改（A）”按钮，返回安装信息摘要界面，如图 1-19 所示。在安装信息摘要界面中，单击“Root 账^①户”，开始设置 root 账户的密码，如图 1-24 所示。

选择“启用 root 账户（E）”，在下面的输入框中输入 root 账户的密码。根据 openEuler 服务器的密码策略要求，需要设置强度较高的密码，本书使用密码“hbsi@123”，输入密码后，单击“完成（D）”按钮，返回安装信息摘要界面（图 1-19）。单击“开始安装（B）”按钮开始安装，如图 1-25 所示。



图 1-24 root 账户密码输入界面



图 1-25 安装界面

安装完成后，单击“重启系统（R）”，重新启动系统，至此完成了系统的安装。为了后续任务的简便，root 账户登录后，使用 passwd 命令，将 root 账户密码修改为“123456”。

参照 controller 服务器的安装过程，根据表 1-1 的参数，完成 compute 服务器和 repository 服务器的安装，即可进行后续的内容学习。

3. 配置 OpenStack 基础环境

1) 配置主机名

使用 root 账户分别登录三台服务器，然后使用命令“hostnamectl set-hostname <hostname>”修改主机名。具体 <hostname> 参见表 1-1 中给出的主机名。

先执行命令“hostnamectl set-hostname controller”修改 controller 主机名，再执行命令“bash”刷新显示的主机名。

```
[root@localhost ~]# hostnamectl set-hostname controller
[root@localhost ~]# bash
[root@controller ~]#
```

compute 和 repository 主机名的修改类似，这里不再赘述，大家可以参考 controller 的修改方法自行修改。

2) 配置网络

三台服务器的网络配置如表 1-2 所示。

^① 图中帐户，本书统一采用规范写法“账户”。

表 1-2 三台服务器的网络配置

服务器	网卡 1 (ens33)		网卡 2 (ens34)
	IP	网关	
controller	192.168.11.10/24	192.168.11.2	192.168.33.10/24
compute	192.168.11.20/24		192.168.33.20/24
repository	192.168.11.199/24		192.168.33.199/24

网卡 1 (ens33) 采用 NAT 模式，需要对外提供服务，因此需要配置网关，该网关的地址是 VMware 中“虚拟网络编辑器”中 NAT 的设置，如果网关地址不对，则服务器主机无法访问网络，需要在 VMware 中进行修改，执行“编辑”→“虚拟网络编辑器”命令，打开“虚拟网络编辑器”窗口，如图 1-26 所示。

如果“VMnet8”后面的子网地址不是“192.168.11.0”，需要单击“更改设置 (C)”按钮，将图 1-26 中的“子网 IP (I)”修改为“192.168.11.0”，然后单击“VMnet 信息”中的“NAT 设置 (S) ...”按钮，打开“NAT 设置”窗口，如图 1-27 所示。



图 1-26 “虚拟网络编辑器”窗口



图 1-27 “NAT 设置”窗口

在图 1-27 中，将“网关 IP”值修改为“192.168.11.2”，单击“确定”按钮，返回“虚拟网络编辑器”窗口。

网卡 ens34 采用仅主机模式，所以需要修改 VMnet1 的配置，将其子网地址修改为“192.168.33.0”，如图 1-28 所示。

然后单击“虚拟网络编辑器”窗口中的“确定”按钮，关闭窗口即可。完成上述设置后，可以启动服务器主机，参考表 1-2 服务器 IP 地址设置的内容进行设置。



图 1-28 修改 VMnet1 的配置

下面以服务器 controller 为例，分别完成两张网卡的设置。

首先，完成网卡 1（ens33）的设置，使用 vi 工具，修改配置文件 “/etc/sysconfig/network-scripts/ifcfg-ens33”，修改内容如下，修改完成后，使用命令 “x” 保存退出。

```
TYPE=Ethernet
BOOTPROTO=static
IPADDR=192.168.11.10
PREFIX=24
GATEWAY=192.168.11.2
DNS1=192.168.11.2
NAME=ens33
DEVICE=ens33
ONBOOT=yes
```

此时，使用命令 “systemctl restart NetworkManager” 重启网络，使用 “ip a” 命令可查看网卡设置，结果如图 1-29 所示。

```
[root@controller ~]# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:96:27:1c brd ff:ff:ff:ff:ff:ff
    inet 192.168.11.10/24 brd 192.168.11.255 scope global noprefixroute ens33
        valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe96:271c/64 scope link
        valid_lft forever preferred_lft forever
3: ens34: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:96:27:26 brd ff:ff:ff:ff:ff:ff
    inet6 fe80::20c:29ff:fe96:2726/64 scope link
        valid_lft forever preferred_lft forever
[root@controller ~]#
```

图 1-29 查看网卡 1 设置

然后，进行网卡 2（ens34）的设置，同样的方法，使用 vi 工具，修改配置文件“/etc/sysconfig/network-scripts/ifcfg-ens34”，修改内容如下，修改完成后，使用命令“x”保存退出。

```
TYPE=Ethernet
BOOTPROTO=static
NAME=ens34
IPADDR=192.168.33.10
PREFIX=24
DEVICE=ens34
ONBOOT=yes
```

使用命令“systemctl restart NetworkManager”重启网络，使用“ip a”命令查看网卡设置，结果如图 1-30 所示。

```
[root@controller ~]# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:96:27:1c brd ff:ff:ff:ff:ff:ff
    inet 192.168.11.10/24 brd 192.168.11.255 scope global noprefixroute ens33
        valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe96:271c/64 scope link
        valid_lft forever preferred_lft forever
3: ens34: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:96:27:26 brd ff:ff:ff:ff:ff:ff
    inet 192.168.33.10/24 brd 192.168.33.255 scope global noprefixroute ens34
        valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe96:2726/64 scope link
        valid_lft forever preferred_lft forever
[root@controller ~]#
```

图 1-30 查看网卡 2 设置

至此，controller 服务器网络配置完成，注意在设置 ens33 网卡时，需要配置上 DNS，因为在项目二中，安装基础服务时，需要从互联网安装一个工具。读者可以使用同样的方法，参照表 1-2 完成 compute 和 repository 服务器的网络配置，在此不再赘述。

网络配置完成后，我们就可以在 Windows 中使用远程工具来访问服务器了，本教材中统一使用工具“MobaXterm”完成远程操作。读者可以自行下载该工具，或者使用其他工具。

3) 配置域名解析

使用 vi 命令在 controller 节点与 compute 节点的 /etc/hosts 文件添加如下内容，添加完成后使用“x”命令保存退出。

```
192.168.33.10 controller
192.168.33.20 compute
```

4) 配置防火墙和 Selinux

为了学习的顺利进行，需要关闭有可能影响 OpenStack 配置的防火墙和 Selinux 等内容，同样以 controller 为例来完成。

首先，关闭服务器的防火墙，执行如下命令关闭防火墙并禁止防火墙自动启动。

```
systemctl stop firewalld
systemctl disable firewalld
```

然后，关闭 Selinux，使用 vi 命令打开如下 Selinux 的配置文件 “/etc/selinux/config”，将第 6 行的 “SELINUX=enforcing” 修改为 “SELINUX=disabled”，最后使用 “x” 命令保存退出。

```
1 # This file controls the state of SELinux on the system.
2 # SELINUX= can take one of these three values:
3 #   enforcing – SELinux security policy is enforced.
4 #   permissive – SELinux prints warnings instead of enforcing.
5 #   disabled – No SELinux policy is loaded.
6 SELINUX=enforcing
7 # SELINUXTYPE= can take one of these three values:
8 #   targeted – Targeted processes are protected,
9 #   minimum – Modification of targeted policy. Only selected processes are protected.
10 #   mls – Multi Level Security protection.
11 SELINUXTYPE=targeted
```

完成上述操作后，controller 的基础配置就完成了，我们可以使用同样的方法，完成 compute 和 repository 的基础配置，在此不再赘述。

任务三 配置 OpenStack 私有仓库

本任务将 repository 服务器配置为私有的 yum 源服务器，供 controller 及 compute 服务器使用，将文件 “openeuler_iaas_controller_compute.tar.gz” 和 openEuler 的镜像文件 “openEuler-22.03-LTS-SP2-x86_64-dvd.iso” 复制到 repository 服务器，然后进一步配置三台服务器的 yum 源信息。

一、配置 repository 服务器的 OpenStack 私有仓库

为了能够顺利安装 OpenStack，本教材要求搭建自己的 repository 服务器作为私有仓库，后续安装 OpenStack 时使用该服务器作为 yum 源，这样能够有效地避免网络问题造成的安装失败。下面开始私有仓库的配置。

1. 上传 yum 源所需的文件

通过拖拽两个文件到 MobaXterm 的左边栏面板 Scp 中，上传 openEuler 的 iso 镜像文件 “openEuler-22.03-LTS-SP2-x86_64-dvd.iso” 和安装 OpenStack 所需内容的压缩包文件 “openeuler_iaas_controller_compute.tar.gz” 到 repository 服务器，如图 1-31 所示。

等待上传完毕，两个文件上传到 root 用户的家目录 “/root/” 中，后续的步骤中，将从该目录中找到上传的文件并使用。使用命令 “ll” 查看上传的文件，如图 1-32 所示。

openEuler 下默认没有安装 tar 包，我们可以直接从互联网安装 tar 包，也可以在配置 yum 仓库后，从私有 yum 仓库安装。安装完成后，需要将压缩文件“openeuler_iaas_controller_compute.tar.gz”解压。

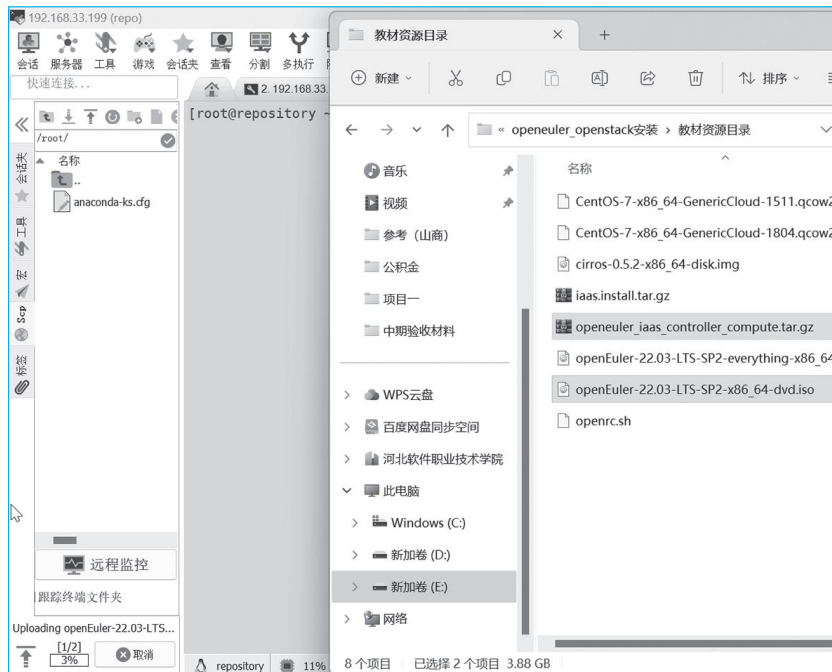


图 1-31 完成 MobaXterm 的 Scp 客户端上传

```
[root@repository ~]# ll
总用量 4072088
-rw----- 1 root root      825 12月  2 04:06 anaconda-ks.cfg
-rw-r--r-- 1 root root 3713007616 12月  2 22:08 openEuler-22.03-LTS-SP2-x86_64-dvd.iso
-rw-r--r-- 1 root root 456804795 12月  2 22:08 openeuler_iaas_controller_compute.tar.gz
[root@repository ~]#
```

图 1-32 查看上传的文件

2. 配置私有 yum 仓库

执行如下命令，将“openEuler-22.03-LTS-SP2-x86_64-dvd.iso”挂载到目录“/mnt”，并在“/opt”下创建目录“openeuler”，将“/mnt”下的内容复制到创建的“/opt/openeuler”文件夹中。

```
mount /root/openEuler-22.03-LTS-SP2-x86_64-dvd.iso /mnt
mkdir /opt/openeuler
cp -rf /mnt/* /opt/openeuler
```

命令执行完毕后，使用 ls 命令查看目录 /opt/openeuler 中的内容，结果如图 1-33 所示。

```
[root@repository ~]# mount /root/openEuler-22.03-LTS-SP2-x86_64-dvd.iso /mnt
mount: /mnt: WARNING: source write-protected, mounted read-only.
[root@repository ~]# mkdir /opt/openeuler
[root@repository ~]# cp -rf /mnt/* /opt/openeuler
[root@repository ~]# ls /opt/openeuler
docs EFI images isolinux ks Packages repodata RPM-GPG-KEY-openEuler TRANS.TBL
[root@repository ~]#
```

图 1-33 执行命令并查看结果

下面开始配置 yum 源，将文件夹 /etc/yum.repos.d/ 中的文件备份（移动）到文件夹 /media/ 中，如图 1-34 所示。

```
[root@repository ~]# mv /etc/yum.repos.d/* /media/
[root@repository ~]# ls /etc/yum.repos.d/
[root@repository ~]# ls /media/
openEuler.repo
[root@repository ~]#
```

图 1-34 备份 yum 源配置文件

执行 vi 命令，创建 /etc/yum.repos.d/openEuler.repo 文件。在文件中输入如下内容。

```
[openEuler]
name=openEuler
baseurl=file:///opt/openeuler
gpgcheck=0
enabled=1
```

在 vi 中保存退出。至此，在私有仓库服务器上，已经配置好 yum 源信息，接着可以清空 yum 缓存，验证 yum 源，如图 1-35 所示。

```
[root@repository ~]# yum clean all
6 files removed
[root@repository ~]# yum makecache
openEuler                               217 MB/s | 3.4 MB      00:00
Metadata cache created.
```

图 1-35 清空 yum 缓存并验证新的 yum 源

看到上面的结果，说明 yum 源配置正常，我们在该服务器上就可以使用 yum 进行安装了。

说明

读者可能会产生疑问，为什么上传的压缩文件“openeuler_iaas_controller_compute.tar.gz”，在 yum 源的配置中并没有使用呢？因为该文件解压后是为 controller（控制节点）和 compute（计算节点）提供安装内容的，所以在 repository（仓库节点）yum 源配置中，不需要设置，但在另外两台服务器上需要配置。

下面来处理压缩文件，先使用命令“yum -y install tar”安装 tar 包。安装完成后，即可使用相应命令进行解压。执行如下命令，将“openeuler_iaas_controller_compute.tar.gz”文件解压到 /opt/ 目录下。

```
tar -zxvf openeuler_iaas_controller_compute.tar.gz -C /opt/
```

解压后，使用 ls 命令查看解压后的内容，执行结果如图 1-36 所示。

```
[root@repository ~]# tar -zxvf openeuler_iaas_controller_compute.tar.gz -C /opt/
[root@repository ~]# ls /opt
openeuler_iaas_controller_compute
[root@repository ~]# ls /opt/openeuler_iaas_controller_compute/
Packages repodata
[root@repository ~]#
```

图 1-36 查看解压后的内容

压缩文件处理完毕后，文件夹“/opt”中，将有两个文件夹“openeuler”和“openeuler_iaas_controller_compute”。在“二、配置 controller 和 compute 的 yum 源”中，将安装和设置 ftp 服务，指定文件夹“/opt”作为 ftp 的根路径，从而使用 ftp 服务，为 controller 和 compute 服务器提供 yum 源。

3. 私有仓库安装 vsftpd 服务

执行命令“yum -y install vsftpd”，在私有仓库（repository 节点）安装 vsftpd 服务，为另外两台服务器提供 yum 源服务，如图 1-37 所示。

```
[root@repository ~]# yum -y install vsftpd
Last metadata expiration check: 0:03:53 ago on 2023年12月02日 星期六 22时20分19秒.
Dependencies resolved.
=====
Package             Architecture Version                               Repository      Size
=====
Installing:
vsftpd              x86_64      3.0.3-33.oe2203sp2                  openEuler       94 k
=====
Transaction Summary
=====
Install 1 Package

Total size: 94 k
Installed size: 215 k
Downloading Packages:
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing                : 1/1
  Installing               : vsftpd-3.0.3-33.oe2203sp2.x86_64 1/1
  Running scriptlet        : vsftpd-3.0.3-33.oe2203sp2.x86_64 1/1
  Verifying                : vsftpd-3.0.3-33.oe2203sp2.x86_64 1/1

Installed:
vsftpd-3.0.3-33.oe2203sp2.x86_64

Complete!
```

图 1-37 私有仓库安装 vsftpd 服务

为了 controller 服务器和 compute 服务器能够匿名访问，使用 vi 命令修改 vsftpd 服务的配置文件“/etc/vsftpd/vsftpd.conf”，将“anonymous_enable=no”中的“no”改为“yes”，并添加一行“anon_root=/opt/”，修改后的内容如下。

```
anonymous_enable=yes
anon_root=/opt/
```

“anon_root=/opt/”配置项指定 ftp 服务匿名访问时，访问到的是“/opt/”目录下的内容，保存退出后，设置 vsftpd 服务器自启动，并启动服务，执行命令如下。

```
systemctl enable vsftpd
systemctl start vsftpd
systemctl status vsftpd
```

运行结果如图 1-38 所示。

```
[root@repository ~]# systemctl enable vsftpd
Created symlink /etc/systemd/system/multi-user.target.wants/vsftpd.service -> /usr/lib/systemd/system/vsftpd.service.
[root@repository ~]# systemctl start vsftpd
[root@repository ~]# systemctl status vsftpd
● vsftpd.service - Vsftpd ftp daemon
   Loaded: loaded (/usr/lib/systemd/system/vsftpd.service; enabled; vendor p
   Active: active (running) since Sat 2023-12-02 22:26:45 CST; 5s ago
     Process: 11755 ExecStart=/usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf (code=ex
   Main PID: 11756 (vsftpd)
      Tasks: 1 (limit: 8924)
     Memory: 408.0K
    CGroup: /system.slice/vsftpd.service
            └─ 11756 /usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf

12月 02 22:26:45 repository systemd[1]: Starting Vsftpd ftp daemon...
12月 02 22:26:45 repository systemd[1]: Started Vsftpd ftp daemon.
```

图 1-38 启动 vsftpd 服务

至此，私有仓库配置就完成了。

二、配置 controller 和 compute 的 yum 源

controller 和 compute 的 yum 源配置方法相同，本教材以 controller 服务器为例进行讲解，读者需要分别在两台服务器上配置。

首先，将文件夹 “/etc/yum.repos.d/” 中的文件备份（移动）到文件夹 “/media/” 中，如图 1-39 所示。

```
[root@controller ~]# mv /etc/yum.repos.d/* /media/  
[root@controller ~]# ls /etc/yum.repos.d/  
[root@controller ~]# ls /media/  
openEuler.repo  
[root@controller ~]# █
```

图 1-39 移动 yum 仓库配置文件

执行 vi 命令，创建文件 “/etc/yum.repos.d/openEuler.repo”，在文件中输入如下内容，并保存退出。

```
[openEuler]  
name=openeuler  
baseurl=ftp://192.168.33.199/openeuler  
gpgcheck=0  
enabled=1  
priority=20  
[openeuler-openstack]  
name=openeuler-openstack  
baseurl=ftp://192.168.33.199/openeuler_iaas_controller_compute  
gpgcheck=0  
enabled=1  
priority=10
```

配置文件中，通过设置 priority 属性，保证安装时优先使用 openeuler-openstack 源，iaas 源中没有的包再去查找 openEuler 源。priority 的取值范围为 1 ~ 99，值越大优先级越低。

至此，在 controller 上 yum 源已经配置完成，接着可以清空 yum 缓存，验证 yum 源，如图 1-40 所示。

```
[root@controller ~]# yum clean all  
50 files removed  
[root@controller ~]# yum makecache  
openeuler-openstack          12 MB/s | 741 kB      00:00  
openeuler                    29 MB/s | 3.4 MB      00:00  
Metadata cache created.  
[root@controller ~]# █
```

图 1-40 清空 yum 缓存并验证新的 yum 源

现在，我们可以安装一些基本的软件，如 vim、net-tools、tree 等，如图 1-41 所示。

```
[root@controller ~]# yum -y install vim net-tools tree
Last metadata expiration check: 0:01:53 ago on 2023年12月02日 星期六 22时34分59秒.
Dependencies resolved.
=====
Package           Arch      Version              Repository            Size
=====
Installing:
net-tools          x86_64    2.10-3.oe2203sp2     openeuler-openstack  199 k
tree               x86_64    2.0.4-2.oe2203sp2    openEuler             53 k
vim-enhanced       x86_64    2:9.0-15.oe2203sp2   openeuler-openstack  1.6 M
Installing dependencies:
gpm-libs           x86_64    1.20.7-26.oe2203sp2  openeuler-openstack   18 k
vim-common         x86_64    2:9.0-15.oe2203sp2   openeuler-openstack  7.7 M
vim-filessystem     noarch    2:9.0-15.oe2203sp2   openeuler-openstack   19 k
Transaction Summary
=====
Install 6 Packages

Total download size: 9.6 M
Installed size: 38 M
Downloading Packages:
(1/6): gpm-libs-1.20.7-26.oe2203sp2.x86_64.rpm 669 kB/s | 18 kB    00:00
```

图 1-41 安装 vim、net-tools 和 tree

配置 compute 的 yum 源，读者可以自行进行。

项目总结

本项目中，通过三个任务了解了 OpenStack 配置时要使用的三台服务器：controller、compute 和 repository，并将三台服务器配置完成，其中 repository 服务器为另外两台服务器提供私有 yum 源服务。我们在后续的项目中，将使用这些服务器完成 OpenStack 各个服务的安装和配置，安装时会自动使用 repository 服务器进行下载安装。

拓展阅读

什么是 openEuler ?

openEuler(也称“欧拉”)是一个开源操作系统，由开放原子开源基金会主导，华为深度参与，面向服务器、桌面和嵌入式等领域，为用户提供安全、稳定、易用的操作系统平台。

当前有多个国产 OSV (operating system vendor, 操作系统供应商) 基于 openEuler 发布了商用版本(如麒麟软件、统信软件、麒麟信安、凝思、超聚变等)。openEuler 是面向数字基础设施的开源操作系统，支持 CPU (中央处理器，包括 ARM、x86、RISC-V 等多种指令集)、GPU (图形处理器)、NPU (神经网络处理器) 等多样性算力，并支持服务器、云计算、边缘计算、嵌入式等应用场景，支持 OT (operational technology, 运营技术) 领域应用及 OT 与 ICT (information and communication technology, 信息与通信技术) 的融合。

openEuler 的主要特点如下。

- (1) 开源：openEuler 是完全开源的操作系统，所有代码都可以在网上公开获取，任何人可以自由使用、修改和发布。
- (2) 安全：openEuler 采用多种安全技术，如内核加固、安全补丁、安全认证等，确保系统的安全可靠。
- (3) 稳定：openEuler 经过了严格的测试和验证，具有良好的稳定性和可靠性。

(4) 易用: openEuler 提供了友好的用户界面和丰富的操作工具, 方便用户使用。

openEuler 的应用场景非常广泛, 包括如下几种。

(1) 服务器: openEuler 可以用于构建各种类型的服务器, 如 Web 服务器、数据库服务器、应用服务器等。

(2) 桌面: openEuler 可以用于构建桌面操作系统, 为用户提供办公、娱乐等应用。

(3) 嵌入式: openEuler 可以用于构建嵌入式系统, 如路由器、交换机、防火墙等。

自 1956 年第一个操作系统诞生以来, 操作系统一直推动着商业计算、个人计算、互联网、移动互联网等产业变革, 也成为构建产业与技术生态的关键。在当前国际环境下, 对我国来说, 拥有自主可控的操作系统尤为重要。随着 5G 移动通信技术、人工智能、云计算等技术的持续演进, 我们正在进入一个万物智联的时代, 泛在化、场景化与生态化等新的诉求给操作系统带来新的挑战, 也给我国操作系统的发展提供了换道超车的机会。

openEuler 是一个覆盖全场景的创新平台, 致力于提供安全、稳定、易用的操作系统。它支持多样性计算, 鼓励开发者提出新想法、开拓新思路、实践新方案。

openEuler 的成功实践证明了中国开发者协同起来有能力创建世界一流的开源社区, 推动了中国操作系统的发展。openEuler 操作系统在中国占据服务器操作系统市场份额的第一名, 用户主动选择 openEuler 的技术路线, 说明了其在国内市场的认可度和应用行业的影响力。openEuler 在政府、工业互联网等领域有着广泛的应用, 特别是在政府领域, 因其高安全性、软件隔离能力强和漏洞修复能力强而受到青睐。

拓/展/训/练

1. repository 服务器配置了 vsftpd 服务后, 在其他两台服务器和 Windows 宿主机中都可以通过 ftp 访问该服务。要求在 Windows 的资源管理器中, 访问 repository 服务器上的 ftp 服务。

2. 在 controller、compute 和 repository 三台服务器与 Windows 宿主机之间, 使用 ping 命令, 测试两个不同的网段 (192.168.11.0/24, 192.168.33.0/24) 的连通性。



在线测试 1

项目二

基础服务

知识目标 >

- (1) 了解 MariaDB 和 RabbitMQ。
- (2) 了解 MariaDB 和 RabbitMQ 在 OpenStack 中的作用。

能力目标 >

- (1) 学会 OpenStack 基础环境的安装和配置。
- (2) 学会安装和配置 MariaDB 数据库。

素质目标 >

- (1) 培养协作意识。
- (2) 培养阅读文档的能力。
- (3) 培养数据库安全意识。

项目描述 >

在本项目中，读者将学习 OpenStack 私有云部署时各基础服务的作用，以及它们的安装和配置方法，包括 MariaDB 数据库、RabbitMQ 消息队列软件等内容。掌握这些基础服务内容，后续可以学习更高级的 OpenStack 服务。

任务一 安装和配置OpenStack基础环境

OpenStack 的正常工作需要很多服务的配合，其中包括很多基础服务，它们为 OpenStack 的其他服务提供基础的技术支撑。如同国家的科技发展，基础科学扎实，科学技术的应用才能更好地服务社会、服务国家。在任务一中，我们安装和配置 OpenStack 基础环境，学会基础环境的安装和配置。

一、安装 OpenStack 基础环境

安装 OpenStack，需要配置服务器 controller 和 compute 的基础环境变量，并部署 OpenStack 云平台基础环境。OpenStack 云平台的基础环境为 OpenStack 各种组件提供底层的数据库支持和消息服务支持。两台服务器均需要进行一些操作，本教材以 controller 为例进行讲解。

先做准备工作，从教材的资源网站下载文件“iaas-install.tar.gz”，将其复制到服务器的“/root/”目录下，上传后使用如下命令进行查看。

```
ll
```

结果如图 2-1 所示。

```
[root@controller ~]# ll
总用量 16
-rw-----. 1 root root 825 12月 2 04:06 anaconda-ks.cfg
-rw-r--r-- 1 root root 11055 12月 3 20:53 iaas-install.tar.gz
[root@controller ~]#
```

图 2-1 查看上传文件结果

执行如下命令，将“iaas-install.tar.gz”文件中的内容解压到“/usr/local/bin”目录中。

```
tar -xf /root/iaas-install.tar.gz -C /usr/local/bin
```

执行完成后，压缩包中的 shell 文件将被放入文件夹“/usr/local/bin”，继续使用“ll”命令查看文件夹“/usr/local/bin”中的内容，如图 2-2 所示。

```
[root@controller ~]# tar -zxf iaas-install.tar.gz -C /usr/local/bin
[root@controller ~]# ll /usr/local/bin
总用量 80
-rwxr-xr-x 1 root root 9429 12月 3 21:04 iaas-install-cinder-compute.sh
-rwxr-xr-x 1 root root 3019 11月 30 10:41 iaas-install-cinder-controller.sh
-rwxr-xr-x 1 root root 2136 11月 30 10:41 iaas-install-dashboard.sh
-rwxr-xr-x 1 root root 3541 11月 30 10:41 iaas-install-glance.sh
-rwxr-xr-x 1 root root 3586 11月 30 10:41 iaas-install-keystone.sh
-rwxr-xr-x 1 root root 2884 11月 30 10:41 iaas-install-mysql.sh
-rwxr-xr-x 1 root root 3204 11月 30 10:41 iaas-install-neutron-compute.sh
-rwxr-xr-x 1 root root 7106 11月 30 10:41 iaas-install-neutron-controller.sh
-rwxr-xr-x 1 root root 3326 11月 30 10:41 iaas-install-nova-compute.sh
-rwxr-xr-x 1 root root 4999 12月 10 22:46 iaas-install-nova-controller.sh
-rwxr-xr-x 1 root root 2382 11月 30 10:41 iaas-install-placement.sh
-rwxr-xr-x 1 root root 4538 11月 30 10:41 iaas-install-swift-compute.sh
-rwxr-xr-x 1 root root 4509 11月 30 10:41 iaas-install-swift-controller.sh
-rwxr-xr-x 1 root root 2693 11月 30 10:41 iaas-pre-host.sh
```

图 2-2 查看文件夹“/usr/local/bin”中的内容

如果使用“ll”命令查看到的文件没有可执行权限，执行如下命令可以为这些文件添加可执行权限。

```
chmod a+x /usr/local/bin/iaas*
```

在 compute 服务器做相同的操作。

二、配置基础环境变量

在 controller 和 compute 两个服务器节点完成安装后，需要在两个节点分别配置环境变量。本教材中，使用的 OpenStack 的环境变量文件为“/etc/openstack/openrc.sh”，首先在“/etc”目录下创建目录“openstack”，命令为“mkdir /etc/openstack”，然后使用 vi 命令“vi /etc/openstack/openrc.sh”创建该环境变量文件，两个节点的配置内容相同，下面列出了需要写入文件的配置项。

```
HOST_IP=192.168.33.10
HOST_PASS=123456
HOST_NAME=controller
HOST_IP_NODE=192.168.33.20
HOST_PASS_NODE=123456
HOST_NAME_NODE=compute
network_segment_IP=192.168.33.0/24
RABBIT_USER=openstack
RABBIT_PASS=000000
DB_PASS=000000
DOMAIN_NAME=demo
ADMIN_PASS=000000
DEMO_PASS=000000
KEYSTONE_DBPASS=000000
GLANCE_DBPASS=000000
GLANCE_PASS=000000
PLACEMENT_DBPASS=000000
PLACEMENT_PASS=000000
NOVA_DBPASS=000000
NOVA_PASS=000000
NEUTRON_DBPASS=000000
NEUTRON_PASS=000000
METADATA_SECRET=000000
INTERFACE_NAME=ens33
Physical_NAME=provider
minvlan=1
maxvlan=1000
```

输入完成后保存退出。上面配置的参数在此不详细讲解，后续项目中各个服务的安装中会使用这些参数，对照后续项目中服务的安装，可以更加深入地理解这些参数的意义。

三、部署 OpenStack 云平台基础环境

在 controller 节点和 compute 节点分别执行脚本“iaas-pre-host.sh”，以部署（也称“安装”）OpenStack 云平台基础环境，完成后使用“reboot”命令重启虚拟机，部署生效。由于脚本中没有错误处理，因此安装的时候需要格外注意，若出现错误提示，先排查问题，避免后期不好排查。另外，由于该脚本不能重复执行，因此虚拟机执行脚本前，需要先在 VMware 中做一下快照，方便后期的测试和恢复。下面分别在 controller 和 compute 两个节点进行安装，分别执行如下脚本。

```
iaas-pre-host.sh
```

出现如图 2-3 所示提示，说明安装完成，需要执行“reboot”命令进行重新启动。

```
Package chrony-4.1-5.oe2203sp2.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
Please Reboot or Reconnect the terminal
```

图 2-3 安装完成提示

脚本“iaas-pre-host.sh”的内容较多，读者可以使用 vi 或 cat 命令查看。在该脚本中对服务器节点做一些初始化的配置，如主机名、防火墙、Selinux、时间同步服务器、DNS 等。下面分析一些重要配置的内容，其他内容在此不做详细讲解。

```
1 #/bin/bash
2 source /etc/openstack/openrc.sh
3
4 #Welcome page
5 cat > /etc/motd <<EOF
6 #####
7 # Welcome to OpenStack #
8 #####
9 EOF
10
11 #selinux
12 sed -i 's/SELINUX=.*/SELINUX=disabled/g' /etc/selinux/config
13 setenforce 0
14
15 #firewalld
16 systemctl stop firewalld
17 systemctl disable firewalld >> /dev/null 2>&1
18
```

```

19 yum remove -y firewalld
20
.....
85
86 #chrony
87 yum install -y chrony
88 sed -i '3, 6s/^/#/g' /etc/chrony.conf
89 sed -i '7G' /etc/chrony.conf
90 sed -i '7s/^/server local-ntp1 iburst/g' /etc/chrony.conf
91
92 systemctl restart chronyd
93 systemctl enable chronyd
94
95 printf "\033[35mPlease Reboot or Reconnect the terminal\n\033[0m"

```

第 2 行，加载了 OpenStack 的环境变量。

第 11 行到第 19 行，处理 Selinux 和防火墙，确保安装时不受其影响。

第 87 行到第 93 行，安装并配置了时间同步服务，指定了时间同步服务器名称为“local-ntp1”。后续的配置中，需要配置一个时间同步服务器，本教材使用 repository 服务器作为时间同步服务器。

四、配置 repository 服务器作为时间同步服务器

在 OpenStack 的工作过程中，controller 节点和 compute 节点的时间需要同步，否则可能造成 OpenStack 的工作不正常，因此需要配置 repository 服务器作为时间同步服务器，在 controller 节点和 compute 节点指定 repository 服务器作为自己的时间同步服务器。

1. 在 repository 服务器上安装和配置时间同步服务

使用命令“rpm -qa chrony”检查系统是否已安装时间同步服务，如果没有安装，需要使用命令“yum -y install chrony”来安装 chrony（时间同步）服务。若已经安装，结果如图 2-4 所示。

```

[root@repository ~]# rpm -qa chrony
chrony-4.1-5.oe2203sp2.x86_64

```

图 2-4 已安装时间同步服务

经过检查，系统已经安装该服务，接着来修改服务的配置，使用 vi 命令，配置文件为“/etc/chrony.conf”。文件内容如下。

```

1 # Use public servers from the pool.ntp.org project.
2 # Please consider joining the pool (https://www.pool.ntp.org/join.html).
3 pool pool.ntp.org iburst
4
5 # Record the rate at which the system clock gains/losses time.
6 driftfile /var/lib/chrony/drift

```

```

7
.....
22 # Allow NTP client access from local network.
23 allow 192.168.33.0/24
24
.....

```

其中第 3 行指定了互联网时间同步服务器为“pool.ntp.org”。

第 23 行已修改为“allow 192.168.33.0/24”，允许“192.168.33.0/24”网段的节点进行时间同步。

修改完成后，执行如下命令重新启动 chrony 服务。

```
systemctl restart chronyd
```

2. 在 controller 和 compute 节点配置 hosts 文件

在 controller 和 compute 节点修改“/etc/hosts”文件，在文件最后添加如下命令。

```
192.168.33.199 local-ntp1
```

在两个节点重新启动 chrony 服务，后续 controller 和 compute 节点就会自动与 repository 节点同步时间。

任务二 安装和配置数据库和消息队列软件

OpenStack 项目中使用的服务、用户及用户的数据均存储在 MariaDB 数据库中，因此需要安装 MariaDB 数据库服务。RabbitMQ 为目前广泛使用的消息队列软件之一，OpenStack 项目内部的不同组件间通过消息总线进行通信，采用 RabbitMQ 消息队列软件完成通信。注意：这部分的服务只需要在 controller 节点上安装。

一、了解 MariaDB

MySQL 是知名的开源数据库，后来其开发者以 10 亿美元的价格，将自己创建的 MySQL AB 公司卖给了 SUN 公司，此后，随着 SUN 被甲骨文公司收购，MySQL 正式成为甲骨文公司的产品之一。

由于甲骨文是典型的商业公司，其模式与开源格格不入，MySQL 的创始人担心 MySQL 有闭源的风险，他主导开发了 MariaDB 数据库，读者可以将其理解为 MySQL 的一个分支。MariaDB 主要由开源社区维护，采用 GPL 授权许可。MariaDB 的开发目的是完全兼容 MySQL（包括 API 和命令行），使之能轻松成为 MySQL 的替代品。在存储引擎方面，MariaDB 使用 XtraDB 来代替 MySQL 的 InnoDB。

MariaDB 虽然被视为 MySQL 数据库的替代品，但它在扩展功能、存储引擎及一些新的功能改进方面都强过 MySQL，因此在很多开源项目都使用了 MariaDB 替代 MySQL，OpenStack 平台就是其中之一。

二、了解 RabbitMQ

RabbitMQ 是一套开源（遵守 MPL 许可证）的消息队列软件，是由 LShift 提供的 AMQP（advanced message queuing protocol，高级消息队列协议）的开源实现，由以高性能、健壮及可伸缩性出名的 Erlang 语言写成。RabbitMQ 支持主流的操作系统，如 Windows、Linux、macOS、FreeBSD 等，支持主流的开发语言，如 Java、Python、C#、JavaScript、Go、Ruby 等。

那么，什么是消息队列呢？消息队列（message queue，后文简称 MQ）是一种应用间的通信方式，消息发送后可以立即返回，由消息系统来确保消息的可靠传递。消息发布者（producer）只管把消息发布到 MQ 中而不用管谁来取，消息使用者（consumer）只管从 MQ 中取消息而不管是谁发布的。发布者和使用者都不用知道对方的存在。

RabbitMQ 就是一个特殊的消息队列软件，非常好地实现了消息队列中消息的发布与使用的管理。

三、安装和配置 MariaDB 和 RabbitMQ

在 controller 节点执行如下脚本开始安装。

```
iaas-install-mysql.sh
```

shell 文件中编写了安装 MariaDB 数据库和 RabbitMQ 消息队列软件的脚本，可以自动进行安装。如果见到如图 2-5 所示的界面，说明安装完成。安装完成的同时，安装文件会根据前面的配置信息进行自动配置。

```
Installing      : etcd-3.4.14-5.oe2203sp2.x86_64      1/1
Running scriptlet: etcd-3.4.14-5.oe2203sp2.x86_64      1/1
Verifying       : etcd-3.4.14-5.oe2203sp2.x86_64      1/1

Installed:
  etcd-3.4.14-5.oe2203sp2.x86_64

Complete!
Created symlink /etc/systemd/system/multi-user.target.wants/etcd.service → /usr/lib/systemd/system/etcd.service.
[root@controller ~]#
```

图 2-5 安装完成界面

下面来分析一下安装文件中的内容。

```
1 #!/bin/bash
2 source /etc/openstack/openrc.sh
.....
10 # MariaDB
11 yum install -y mariadb mariadb-server python-PyMySQL
.....
16 systemctl enable mariadb.service
17 systemctl restart mariadb.service
18
19 expect -c "
20 spawn /usr/bin/mysql_secure_installation
```

```
.....
37 expect \"Reload privilege tables now?\"
38 send \"y\\r\\n\"
39 expect eof
40 \"
41
42 # RabbitMQ
43 yum install rabbitmq-server -y
44 systemctl start rabbitmq-server.service
45 systemctl enable rabbitmq-server.service
46
47 rabbitmqctl add_user $RABBIT_USER $RABBIT_PASS
48 rabbitmqctl set_permissions $RABBIT_USER \".*\" \".*\" \".*\"
49
50 # Memcache
51 yum install memcached python3-memcached -y
.....
57 systemctl start memcached.service
58 systemctl enable memcached.service
59
60 # ETCD
61 yum install etcd -y
.....
71 systemctl start etcd
72 systemctl enable etcd
```

第 2 行，装载了 OpenStack 的环境变量。

第 11 行到第 17 行，完成了 MariaDB 的安装和启动。

第 19 行到第 39 行，使用 expect 交互命令，完成了 MariaDB 数据库设置。

第 43 行到第 48 行，完成了 RabbitMQ 的安装和配置，并启动 RabbitMQ。RabbitMQ 是开源的消息队列软件，在 OpenStack 中完成各个组件之间的通信功能。

第 51 行到第 58 行，完成了 Memcached 服务的安装和配置，并启动 Memcached（Memcache 是项目名称，Memcached 是主程序文件）。Memcached 是一款开源、高性能、分布式内存对象缓存系统，可应用于各种需要缓存的场景，其主要目的是通过减少对数据库的访问来加速 Web 应用程序。它是基于内存的“键值对”存储，用于存储数据库调用、API（应用程序接口）调用或页面引用结果的直接数据，如字符串、对象等。

第 61 行到第 72 行，完成了 etcd 服务的安装和配置，并启动 etcd。etcd 是一个高可用性的分布式键值存储系统，它主要用于共享配置和服务发现。etcd 以其高度一致性、高可用性、可扩展性和开放 API 等特点而闻名业界。

项目总结

OpenStack 平台提供的服务，包含多个组件（应用），这些组件的数据需要存储，且组件和组件之间需要通信，因此 OpenStack 平台使用了 MariaDB 数据库和 RabbitMQ 消息队列软件分别完成数据的存储和组件之间的通信。在后续的项目中，将依次安装 OpenStack 平台需要使用的各个组件。

拓展阅读

科技创新为人民，“云”力量赋能数字中国建设

2023 年年初，中共中央和国务院印发的《数字中国建设整体布局规划》（以下简称《规划》）指出，要充分发挥数字中国建设的驱动引领作用，统筹推进数字技术在经济、政治、文化、社会、生态文明建设各领域全过程中的融合应用，更好支撑经济社会高质量发展，为推进中国式现代化提供不竭动力。

当前，数字技术日益成为创新驱动发展的先导力量，如何顺应信息革命时代浪潮，抢抓数字化发展历史机遇，推动经济社会变革成为当今时代决定大国兴衰的重要因素。

在此背景下，移动云砥砺前行，一直以来不断强化核心技术，自主创新，拥有完全的自主知识产权和产品体系。在 2023 年的移动云大会上，移动云重磅发布面向算力网络打造的新型技术架构体系——“移动云技术内核 3.0”，进一步实现了技术领先、安全可靠的发展目标；展示了业界首个算网大脑，基于对资源的实时感知、智能编排、灵活调度等能力，生成算网服务的最优解……

践行“为国建云”使命，移动云持续关注国家的战略需求、社会的发展需求、人民的生活需求，积极发挥云计算技术创新优势，为解决社会数字化转型难题、赋能社会民生发展、建设数字中国贡献自己的“云”力量。

（《科技创新为人民，“云”力量赋能数字中国建设》，中国日报，2023 年 6 月 30 日，有修改）

拓/展/训/练

使用用户账号 root、密码 000000 登录 MariaDB 数据库，查看已经安装的数据库。



在线测试 2