

OpenStack in Action

OpenStack 实战指南

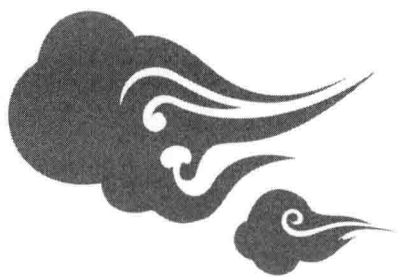
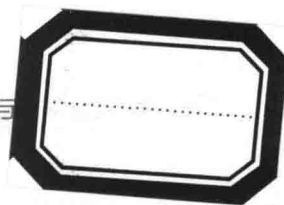
.....黄凯 毛伟杰 顾俊杰 著.....

结合作者在云计算领域的实战经验和最佳实践，深入剖析OpenStack体系结构的设计理念及实际应用，透析其中每个模块的工作原理



机械工业出版社
China Machine Press

云计算与



OpenStack in Action

OpenStack实战指南

黄凯 毛伟杰 顾俊杰 著



机械工业出版社
China Machine Press

图书在版编目 (CIP) 数据

OpenStack 实战指南 / 黄凯等著. —北京: 机械工业出版社, 2014.8
(云计算与虚拟化技术丛书)

ISBN 978-7-111-47632-0

I. O… II. 黄… III. 计算机网络—指南 IV. TP393-62

中国版本图书馆 CIP 数据核字 (2014) 第 185949 号

OpenStack 实战指南



出版发行: 机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码: 100037)

责任编辑: 余浩

印刷: 北京市荣盛彩色印刷有限公司

开本: 186mm × 240mm 1/16

书号: ISBN 978-7-111-47632-0

责任校对: 董纪丽

版次: 2014 年 9 月第 1 版第 1 次印刷

印张: 18

定价: 69.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

客服热线: (010) 88378991 88361066

购书热线: (010) 68326294 88379649 68995259

投稿热线: (010) 88379604

读者信箱: hzsj@hzbook.com

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问: 北京大成律师事务所 韩光 / 邹晓东

Preface 前言

在 2007 年前后，许多网站站长发现当时一部分主流机托管商开始逐步使用虚拟化技术提供云主机的托管服务，而当时的虚拟化程度还停留在半虚拟化或类似 OpenVZ 这样的伪虚拟化的水平。即便如此，一台服务器已经可以承载数十台甚至上百台虚拟主机的负荷，而且安全性更好，价格也更低。通过更加有效的隔离和封装，虚拟主机的创建和恢复也变得更加迅速和可靠。与此同时，像 Eucalyptus 和 OpenNebula 这样的云计算管理工具也开始逐步进入人们的视野。

次年，笔者机缘巧合地加入了 EMC（也就是 VMware 的母公司）在中国的研发实验室。此后几年，以 VMware ESX 为主的私有云在中国的各个数据中心遍地开花。与此同时，存储超配、内存去重、SDN 等技术也纷至沓来，使得云计算更加集约化，各种硬件资源的管理也更加合理。笔者正是在这种“天时地利”的条件下在实验室里开展了各项研究，让 P2P 计算、Hadoop 计算和 HPC 在虚拟化和云计算技术的辅助下更加灵活和高效。

后来笔者加入了 IBM 中国实验室，负责 x86 架构下的云计算解决方案的设计和开发。作为 OpenStack 的主要代码贡献者，IBM 公司的资源让笔者对 KVM 和 OpenStack 等开源的云计算方案有了零距离接触和本质了解，为了让每 GB 级别的存储和每 MB 级别的带宽给用户带来更多的价值，笔者也经常不断地将方案推翻重来。也就是在这段时间，OpenStack 逐渐步入了它的成熟期。

从 OpenStack 早期的 Essex 版本一直到最新的 IceHouse 版本，笔者在不同的环境下都部署过，深深体验到了由于早期 OpenStack 不成熟导致的各种问题和障碍，有时候两天或三天也找不到解决方案，只能硬着头皮“挖” StackOverflow 的帖子，甚至一行行地从 OpenStack 代码中“抠” Bug。虽然在这期间笔者增长了知识，也学会了苦中作乐，但笔者却不想读者也为同样的问题徒增几根白发或加班熬夜。幸运的是，随着 OpenStack 版本的不断升级，无论是代码质量还是文档的详尽度，都在一步步走向成熟，变得更加易用，而对于企业级应用

中亟待解决的问题和需求，OpenStack 都逐渐有了清晰的答案和参考解决方案。随之而来的是许多早期云计算用户的态度，也从最早的 FUD (Fear/Uncertainty/Doubt, 即害怕、不确定、怀疑的英文缩写) 到逐渐接受甚至主动申请往 OpenStack 上迁移。

时至今日，国内用户对 OpenStack 的兴趣与日俱增，2013 年的“OpenStack 大会”更是破格在中国香港举办。本书的出版对于帮助读者学习和了解 OpenStack 应该是非常及时的。

2014 年将是 OpenStack 加速进入生产环境和应用的元年，撇开 Mirantis、Rackspace 和 Softlayer 这些典型的云主机服务商，各大电信企业、电商也在跃跃欲试，使用 OpenStack 部署私有云或公有云环境。举个例子，笔者就见证了国内某金融服务商的数百台服务器、几千个虚拟机的 OpenStack 群集应用。据估计，到 2015 年，全球将会有约 50 亿美元投入到云计算建设或升级中，这其中以 OpenStack 为基础的云计算平台和应用肯定会大放异彩；而随着亚马逊云计算在中国的落地和国内大小云计算提供商的纷纷上马，好戏还在后头。

云计算并不只是虚拟化，实际是由虚拟化引出的所谓“软件定义的数据中心”。它不仅包括网络的虚拟化、计算的虚拟化和存储的虚拟化，还对传统数据中心的运维、管理，甚至硬件选型都产生了深刻的影响，因此，读者在学习和应用 OpenStack 云平台部署的时候，最好同时了解一下：哪些功能是在硬件层提供的，哪些功能是在协议层提供的，哪些功能是在虚拟层提供的，哪些功能是在 OS 层甚至应用层提供的。思考这些问题不仅可以帮助优化系统的整体性能，也会帮助系统管理员更加高效和简单地管理整体云计算环境。同时读者也需要对 Linux 操作系统有更深入的了解。限于篇幅，本书无法对上述问题一一给出解答，笔者推荐读者参看由机械工业出版社出版的 Kai Hwang 所著的《云计算与分布式系统：从并行处理到物联网》，这本书是开始着手学习 OpenStack 之前的一本非常好的基础及进阶读物。

OpenStack 社区和各大网站上关于 OpenStack 配置和运维的文档和讨论繁多，有时笔者觉得本书中某些内容无非是在做搬运工兼翻译的活计，但回头想想，当有人向自己虚心请教 OpenStack 的问题时若被回复一句“自己看文档去”，他心里会是何等失落。所以在写作中，笔者将体力和脑力活动一并进行，把良莠不齐的文档甄选一番，省去很多初涉 OpenStack 云计算读者的麻烦，让他们把时间花在更有意义的开发上，这也算是一件很有意义的事情。我希望本书的出版能让更多的人掌握 OpenStack 的原理和操作，广泛了解目前处于云计算前沿的技术。我还希望借此推动国内高校云计算的教学——不仅作为一门选修课，更要帮助学生将其作为 IT 领域的核心技能之一来掌握。

随着越来越多的开发者投入 OpenStack 这个生态圈，笔者也特别希望分享自己总结的成功或失败的经验，避免其他人走同样的弯路，帮助他们进入 OpenStack 开发的“快车道”。虽然笔者已经在企业级计算、存储及服务器领域辗转十余年，对于数据中心的建设与

运维稍有心得，但一人智短，三人智长，我特别要感谢 Frank、杨海明、Mark Zhu，他们在本书的创作过程中，为笔者解惑释疑，帮助扫除知识的死角，使笔者可以从系统的高度去阐述 OpenStack，避免局限在技术细节而忽视了全系统的协同。这里笔者也要感谢奋战在一线的同事，他们给予的许多宝贵的现场反馈让笔者重新思索如何将 OpenStack 与用户实际需求相结合，避免纸上谈兵和教条主义。他们其中很多人是 OpenStack 的核心代码开发者，对专业知识的深入理解让笔者钦佩，也深深感到设计 OpenStack 这样一个通用云计算框架的不易。最后，笔者也要感谢无私地将 OpenStack 学习笔记分享出来的 lvsaloon、nc_triin 等技术爱好者。

黄 凯

2014 年 4 月于上海

目 录 *Contents*

前 言

第一部分 基础篇

第 1 章 OpenStack 简介	2
1.1 OpenStack 概述	2
1.2 OpenStack 的结构	3
1.3 OpenStack 的功能与作用	4
1.4 OpenStack 与 CloudStack 的比较	6
1.5 OpenStack 应用现状和发展趋势	8
1.6 体验 OpenStack	10
1.6.1 初探 OpenStack	10
1.6.2 创建 OpenStack 虚拟机实例	12
1.6.3 创建虚拟机流程概述	14
1.6.4 创建 OpenStack 磁盘实例	16
1.6.5 创建块存储流程概述	20
1.7 OpenStack 体系结构	22
1.7.1 OpenStack 设计原则	22
1.7.2 OpenStack 架构	23
1.8 OpenStack 的开发资源	24
1.8.1 OpenStack 社区	24

1.8.2	OpenStack 基金会	25
1.8.3	OpenStack 项目资料	25
1.9	OpenStack 非核心项目介绍	29
1.9.1	Ironic 项目介绍	29
1.9.2	Tempest 项目介绍	33
第 2 章	OpenStack 的安装	35
2.1	在 Ubuntu 上使用二进制包安装	35
2.1.1	控制节点的安装	35
2.1.2	网络节点的安装	55
2.1.3	计算节点的安装	60
2.1.4	块存储节点的安装	64
2.2	在 Ubuntu 上使用源代码编译安装	66
2.2.1	控制节点的安装	67
2.2.2	计算节点的安装	79
2.2.3	网络节点的安装	81
2.2.4	块存储节点的安装	82

第二部分 进阶篇

第 3 章	OpenStack 组织结构一览	84
3.1	组件关系	84
3.1.1	Nova 组件	85
3.1.2	Swift 组件	87
3.1.3	Keystone 组件	87
3.1.4	Glance 组件	88
3.1.5	Neutron 组件	89
3.1.6	Cinder 组件	89
3.2	OpenStack 目录组织结构	90
3.2.1	Nova 目录结构	90
3.2.2	Swift 目录结构	91

3.2.3	Keystone 目录结构	92
3.2.4	Glance 目录结构	92
3.2.5	Neutron 目录结构	93
3.2.6	Cinder 目录结构	93
3.3	OpenStack 配置文件	94
3.3.1	Nova 配置文件及日志	94
3.3.2	Swift 配置文件及日志	95
3.3.3	Keystone 配置文件及日志	95
3.3.4	Glance 配置文件及日志	96
3.3.5	Neutron 配置文件及日志	96
3.3.6	Cinder 配置文件及日志	96
3.4	小结	98
第 4 章	OpenStack API 的介绍和开发	99
4.1	简述 Nova API 体系	99
4.1.1	WSGI 架构	99
4.1.2	API 响应流程	102
4.1.3	扩展 API 的加载	103
4.1.4	Nova API 列表	104
4.1.5	分页查询和查询优化	106
4.1.6	如何编写一个核心 API	108
4.1.7	如何编写一个扩展 API	108
4.1.8	通过 Filter 为 API 增加功能	109
4.2	理解 Eventlet	110
4.3	自定义 DashBoard 面板	112
4.3.1	理解 Django 框架	112
4.3.2	Django 界面国际化	112
4.3.3	玩转 Horizon	115
第 5 章	Keystone 认证组件	119
5.1	认证组件 Keystone 介绍	119

5.1.1	基本概念	119
5.1.2	用户管理	120
5.1.3	服务管理	123
5.2	配置文件及参数选项	123
5.2.1	keystone.conf 配置文件示例	124
5.2.2	keystone-paste.ini 配置文件示例	134
5.2.3	logging.conf 配置文件示例	136
5.2.4	连接 OpenStack 服务到 Keystone	137
5.3	原理分析	139
5.3.1	Keystone 认证原理	139
5.3.2	图解 Keystone	143
5.4	关键源代码阅读	146
5.4.1	源代码分析之服务启动	146
5.4.2	源代码分析之 CLI 调用	151
5.5	案例：配置 Keystone 使用 Active Directory 进行认证	152
第 6 章 Glance 镜像组件		154
6.1	镜像组件介绍	154
6.1.1	Glance 表结构	155
6.1.2	Glance 中镜像的概念	157
6.2	Glance 的配置文件	159
6.2.1	glance-api.conf 文件概述	159
6.2.2	配置镜像缓存	163
6.2.3	glance-registry.conf 文件概述	165
6.3	设置 Glance 的后端存储	166
6.4	制作镜像	167
6.4.1	使用 virt-install 创建 CentOS 镜像	167
6.4.2	在 Ubuntu 上使用 QEMU 相关命令制作 Windows 镜像	170
第 7 章 Nova 计算组件		174
7.1	Nova 组件介绍	174

7.2	虚拟化组件	175
7.3	配置文件及参数	177
7.3.1	数据库配置	178
7.3.2	Hypervisor 配置	179
7.3.3	RPC 配置	181
7.3.4	配额设置	182
7.3.5	日志配置	183
7.3.6	调度配置	184
7.3.7	VNC 配置	184
7.4	关键源代码阅读	185
7.4.1	Nova 服务启动	185
7.4.2	虚拟机状态转换	188
7.4.3	Nova Context	195
7.4.4	Keystone 认证处理	195
7.4.5	REST API 调用	196
7.4.6	组件间 RPC 调用	198
7.4.7	Hypervisor 驱动	198
7.5	虚拟机镜像格式	202
第 8 章 Neutron 网络组件		203
8.1	Neutron 概述	203
8.2	使用 Open vSwitch 的 plugin	208
第 9 章 Cinder 块存储组件		216
9.1	Cinder 交互流程	216
9.1.1	Nova 现有块设备操作 API 统计	216
9.1.2	Nova-Cinder 交互流程分析	217
9.1.3	相关代码源文件	226
9.1.4	使用 Cinder 实现云硬盘需要注意的问题	227
9.2	配置从 Volume 启动虚拟机	227
9.3	使用 Ceph 作为 Cinder 的后端	230

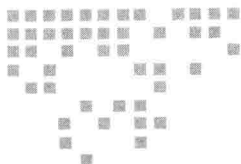
第 10 章 OpenStack 日常运维	233
10.1 维护与诊断.....	233
10.1.1 控制节点和 Swift 的维护与纠错.....	233
10.1.2 计算节点的维护与纠错.....	234
10.1.3 网络诊断.....	239
10.2 标准化修复与例行检查流程.....	246
10.2.1 标准化修复.....	246
10.2.2 例行检查.....	247
10.3 日志与监控.....	247
10.3.1 定位错误.....	247
10.3.2 错误日志.....	248
10.3.3 集中管理日志.....	252
10.3.4 监控.....	253
10.4 备份与恢复.....	257
10.4.1 需要备份的数据.....	257
10.4.2 数据库备份.....	257
10.4.3 文件系统备份.....	258
10.4.4 数据恢复.....	259
第 11 章 使用 Fuel 快速安装 OpenStack	260
11.1 Fuel 规划.....	261
11.2 Fuel 安装及使用.....	263
11.3 小结.....	273



第一部分 *Part 1*

基础篇

- 第 1 章 OpenStack 简介
 - 第 2 章 OpenStack 的安装
-



OpenStack 简介

第 1 章

1.1 OpenStack 概述

OpenStack 是当今最流行的开源云平台管理项目，可以控制整个数据中心计算、存储和网络资源的大型资源池。从 OpenStack 的名字可以看出它大致的含义，Open 顾名思义为开源软件，开放式的设计理念、开放式的开发模式、开放式的社区，Stack 意为堆，可以理解为云计算是靠每一块小瓦砾堆砌而成。OpenStack 并不是单独的一个软件，它由多个组件一起协作完成某些具体工作。OpenStack 本身就是一个巨大的开源软件集合，集各种开源软件之大成。当你想寻找 AWS EC2 的替代品时，OpenStack 将是一个不错的选择。

云计算的概念并不是很新。实际上，AWS EC2 已经出现有 7 年左右。虽然 OpenStack 是如今最为流行的一种可用的开源云计算解决方案之一，但它不是最早的一个。它是在公共和私有领域开发的两种旧解决方案的综合。

OpenStack 是一个非常年轻的开源项目，最初是由美国国家航空航天局（NASA）和 Rackspace 合作研发的项目，2010 年 7 月以 Apache 2.0 许可证授权开源，源代码来自于 NASA 的 Nebula 云平台 and Rackspace 的分布式云存储（Swift）项目。NASA 最初使用的是 Eucalyptus 云计算平台，当规模持续快速增长后，Eucalyptus 已经不能满足 NASA 的云计算规模，而 Eucalyptus 是不完全开放源代码的（“开放核”模式）。NASA 首席技术官 Chris Kemp 的研究小组为此专门建立了自己的计算引擎，新平台命名为 Nova，并将其开源。在 2010 年 NASA 和 Rackspace 分别将 Nova 和 Swift 项目代码开源时，已经获得了 25 个企业和组织的支持。

OpenStack 致力于一个开放式设计过程，每 6 个月开发社区就会举行一次设计峰会来收集需求并写入即将发布版本的规格中。设计峰会是完全对公众开放的，包括用户、开发者和上游项目。社区收集需求和制定经过批准的线路图，用于指导未来 6 个月的发展。

OpenStack 使用 Apache 2.0 许可证，兼容 GPLv3 以及 DFSG。

下面来了解一下 OpenStack 的优势和劣势。

OpenStack 的优势：

- 解除厂商绑定。
- 具有可扩展性及很好的弹性，可定制化 IaaS。
- 良好的社区氛围。

OpenStack 的劣势：

- 入手难、学习曲线较高，在对整体把握不足的情况下，很难快速上手。
- 偏底层，需要根据实际应用场景进行二次开发。
- 现阶段的厂商支持较弱，商业设备的 OpenStack 驱动相对不够全面。

1.2 OpenStack 的结构

OpenStack 包含了许多组件。有些组件会首先出现在孵化项目中，待成熟以后进入下一个 OpenStack 发行版的核心服务中。同时也有部分项目是为了更好地支持 OpenStack 社区和项目开发管理，不包含在发行版代码中。

OpenStack 的核心服务包括：

- Nova 计算服务 (Compute as a Service)
- Neutron 网络服务 (Networking as a Service)
- Swift 对象存储服务 (Object Storage as a Service)
- Cinder 块存储服务 (Block Storage as a Service)

OpenStack 的公共服务包括：

- Glance 镜像服务 (Image as a Service)
- Keystone 认证服务 (Identity as a Service)
- Horizon 仪表盘服务 (Dashboard as a Service)

OpenStack 的依赖库项目包括：Oslo 基础设施代码共享依赖库 (Common Lab as a Service)。

OpenStack 的孵化项目包括：

- Ceilometer 计费 & 监控服务
- Heat 编排服务
- Ironic 物理设备服务 (Bare Metal as a Service)

- ❑ Marconi 消息队列服务 (Message Queue as a Service)
- ❑ Savanna 大数据处理 (MapReduce as a Service)
- ❑ Trove 数据库服务 (DataBase as a Service)

OpenStack 的其他项目涉及：

- ❑ Infrastructure OpenStack 社区建设项目
- ❑ Documentation OpenStack 文档管理项目
- ❑ TripleO OpenStack 部署项目
- ❑ DevStack OpenStack 开发者项目
- ❑ QA OpenStack 质量管理项目
- ❑ Release Cycle Management 版本控制项目

这些 OpenStack 项目有一些共同点，比如：

- ❑ OpenStack 项目组件由多个子组件组成，子组件有各自的模块。
- ❑ 每个项目都会选举 PTL (Project Technical Leader)。
- ❑ 每个项目都有单独的开发人员和设计团队。
- ❑ 每个项目都有具有优良设计的公共 API，API 基于 RESTful，同时支持 JSON 和 XML。
- ❑ 每个项目都有单独的数据库和隔离的持久层。
- ❑ 每个项目都可以单独部署，对外提供服务，也可以在一起协同完成某项工作。
- ❑ 每个项目都有各自的后端驱动，所有的驱动都可以以 plugin 方式加载。
- ❑ 每个项目都有各自的 client 项目，如 Nova 有 nova-client 作为其命令行调用 RESTful 的实现。

除了以上项目，OpenStack 的其他项目或多或少也会需要 Database (数据库)、Message Queue (消息队列) 进行数据持久化、通信。

1.3 OpenStack 的功能与作用

当今的数据中心，许多服务器都遇到过同样的问题，即计算、电源、网络带宽等资源利用率不足。例如，某个项目可能会需要大量计算资源来完成计算，而一旦完成了计算任务，将不再需要那么多的计算资源。当用户想要一种灵活的、按需供给计算资源的服务，通过自动化或很少人工干预就能使用时，那么云计算就是最好的选择之一。“云计算”通常包含了一个服务责任 (Service Level Agreement, SLA)，表示云计算服务提供商承诺的性能、规格、可用率等。云计算服务让用户通过一个共享的计算资源、网络带宽、存储池，运行应用程序或服务来完成计算工作，并按资源的使用量来计费。

这些关于云计算服务的主要特点如下。

- 按需自助服务：用户可以提供自己的需要订购所需的计算、存储和网络资源，而几乎

不需要人工干预。

- 网络访问：可以通过网络使用任意类型的（异构）计算能力。通过标准化的机制调用计算资源而不受限于具体的访问设备。
- 资源池：多个用户可以同时访问和使用云计算提供的计算服务，服务提供商根据消费者的计算要求或实际使用量汇集和分配实际的计算资源。
- 弹性：可根据需要在不停机或短暂停机后迅速垂直或横向扩展。
- 计量或测量服务：按照使用的时间、传输或存储的字节数支付云计算服务，并提供消费者具体的资源消费图表。同时，它也可以根据消费者的不同需求提供定制化的计费模式。

下面了解一下当今的 IaaS/Cloud 与 OpenStack 的对比情况，从而进一步了解 OpenStack 的特点，见表 1-1。

表 1-1 IaaS/Cloud 与 OpenStack 组件的对比

当今的 IaaS/Cloud	作用	OpenStack	作用
vApps	新的或传统操作系统	Apps	提供云应用或 Web 应用
vCloud	管理云	Nova Compute	管理 x86 服务器、集群
vCenter	管理虚拟机		
网络	高可用 (HA)	网络	Neutron 提供扁平网络
存储	高可用 (HA)	存储	Swift、Cinder 提供云存储

当今的云计算概念是由 Google 公司提出的，狭义的云计算是指 IT 基础设施的交付和使用模式，按需取用所需的 IT 资源；广义的云计算是指服务交付和使用模式，通过网络按需取用所需的服务，这种服务可以是 IT、软件、互联网相关的，也可以是其他服务。它具有超大规模、虚拟化、可靠安全、弹性等特性。通过 SaaS (Software as a Service)、PaaS (Platform as a Service)、IaaS (Infrastructure as a Service) 提供从上到下不同层面的云计算服务。

云计算 (Cloud Computing) 是网格计算 (Grid Computing)、分布式计算 (Distributed Computing)、并行计算 (Parallel Computing)、效用计算 (Utility Computing)、联机存储技术 (Network Storage Technology)、虚拟化 (Virtualization)、负载均衡 (Load Balance) 等一系列传统计算机技术和网络技术发展融合的产物。它旨在通过网络将多个成本低廉的计算实体整合成一个大计算资源池，并借助 SaaS、PaaS、IaaS 等服务模式，将强大的计算能力分发到终端用户手中。云计算的核心理念就是通过不断提高“云”端处理能力，减轻用户负担，将一系列的 IT 能力以服务形式提供给用户，简化用户终端的处理负担，最终使用户成为一个单纯的输入/输出设备，享受“云”提供的强大计算处理及服务能力。

OpenStack 具有建设这样资源池的能力，通过 OpenStack 的各种组件多种模式的排列组合，可以搭建成各种规模的“云”，这些云可以是私有云、公有云、混合云。

OpenStack 具有三大核心功能，即计算、存储、网络，分别对应相应的项目 Nova、Cinder。其中 Neutron。其中 Nova 提供了计算资源的管理，可以管理跨服务器网络的 VM 实例。同时，Nova 还提供对多种 Hypervisor 的支持，如 KVM、QEMU、Xen、LXC、