



Inside Explorer OpenStack
Develop OpenStack Components with Python

深度实践 OpenStack

基于Python的OpenStack组件开发

喻涛 张伟 田亮 郝宁 等著

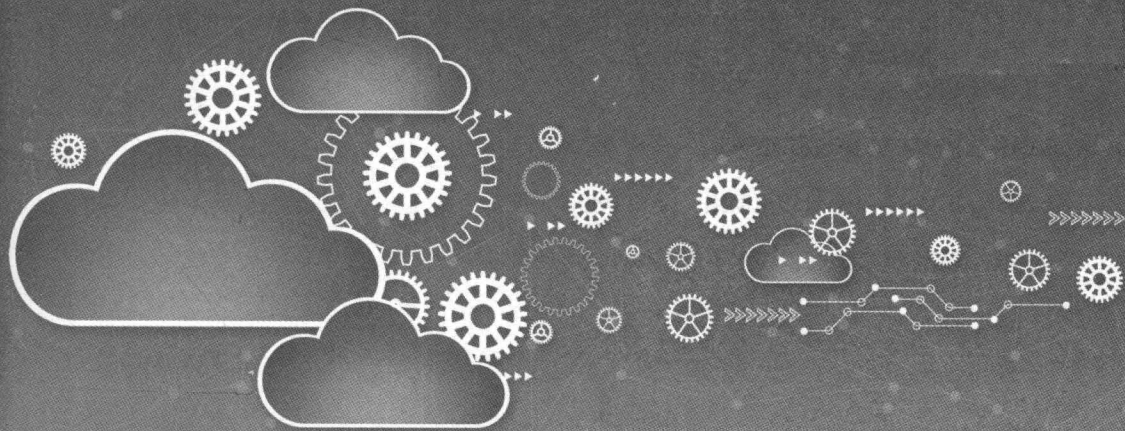
国内领先的企业级OpenStack云服务提供商海云捷迅多年经验总结，OpenStack基金会个人独立董事王庆推荐

从源码层面深入分析OpenStack核心组件重要功能的实现原理，详细讲解利用Python对OpenStack组件进行二次开发需要的全套技术



机械工业出版社
China Machine Press

云计算与虚拟化技术丛书



Inside Explorer OpenStack
Develop OpenStack Components with Python

常州大学图书馆
藏书章

深度实践 OpenStack

基于Python的OpenStack组件开发

喻涛 张伟 田亮 郝宁 张家龙 赵利军 李飞 著

 机械工业出版社
China Machine Press

图书在版编目 (CIP) 数据

深度实践 OpenStack : 基于 Python 的 OpenStack 组件开发 / 喻涛等著. —北京: 机械工业出版社, 2018.3

(云计算与虚拟化技术丛书)

ISBN 978-7-111-59077-4

I. 深… II. 喻… III. 计算机网络 IV. TP393

中国版本图书馆 CIP 数据核字 (2018) 第 025253 号

深度实践 OpenStack

基于 Python 的 OpenStack 组件开发

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: 李 艺

责任校对: 李秋荣

印 刷: 北京市兆成印刷有限责任公司

版 次: 2018 年 3 月第 1 版第 1 次印刷

开 本: 186mm × 240mm 1/16

印 张: 20.5

书 号: ISBN 978-7-111-59077-4

定 价: 79.00 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) 88379426 88361066

投稿热线: (010) 88379604

购书热线: (010) 68326294 88379649 68995259

读者信箱: hzit@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

Foreword 序

2014年年底，我在从事 OpenStack 工作两年多之后第一次在北京环球贸易中心接触到了李华和他们的海云团队，之后我们团队与海云也有过多次技术交流，我个人也聆听过李华对 OpenStack 现实困境和未来发展的各种看法。我被李华和他们创业团队的远见，以及在云计算事业里的决断气魄而折服，也为他们在 OpenStack 领域的低调实干精神感到钦佩。

后来，由于英特尔公司对海云捷迅公司的投资关系，我们英特尔研发团队与海云创业团队形成了稳定的合作模式，建立了稳固的合作关系，在工作中发展了深厚的友谊。我们主要在上游社区负责英特尔平台相关的功能开发，而海云除了开发 OpenStack 功能之外，更多是要在客户那边负责 OpenStack 部署和实施。也正是因为这样，我们也从海云那边得到了第一手客户需求信息和部署实施经验。

2015年年中，李华毫不吝啬地送给我一本内部交流资料用于学习。该内部交流资料是培训企业运维人员，如何一步一步在企业内部用 OpenStack 部署工具，从零开始搭建云计算系统的，内容基本覆盖了所有 OpenStack 关键项目的基础用法，非常接地气。对于我这样一名以 OpenStack 代码开发出身的开发人员来说，受益良多。而且我们研发团队围绕着固定几个关键项目进行开发，很难接触到 OpenStack 全局，也很少有机会接触到为支撑一个企业内部私有云所需的所有 OpenStack 组件的工作原理，这本内部交流资料使我拓宽了视野，增长了见识。

这次欣闻海云即将出版这本书，并将用于企业培训，我特别期待它能早日面世。从李华那里拿到电子版后，我快速浏览了一遍，它除了会讲述 OpenStack 那些子项目的用法之外，还有众多的背景介绍，比如 KVM、Open vSwitch 和 Ceph 等，甚至还有海云部署案例分析。这本书的内容比我当年拿到的内部交流资料的内容还要丰富和完整，在感谢海云对 OpenStack

社区慷慨分享的同时，我相信读者一定能从书中学到更全面、更深入的 OpenStack 知识，并运用到自己的生产部署和实践开发中。

王庆 博士

OpenStack 基金会个人独立董事

英特尔开源技术中心网络和存储部门开发经理

为什么写这本书

从2010年7月至今，OpenStack已经发布了15个版本，并成为云计算基础架构（IaaS）的事实标准。目前OpenStack也被国内企业接受，包括中国移动通信集团公司、中国电信集团公司、中国银联股份有限公司、中国国电集团公司等都在直接或间接使用OpenStack。

随着云的普及，越来越多的业务部署并运行在OpenStack云中，原生的OpenStack的界面及功能已经远远不能满足需求，有规模的企业寻求定制自己的OpenStack，其中包括界面的定制、计算和网络等功能的增强等。而与OpenStack相关的创业公司经过几年的发展，都已经有了自己成熟的产品，这些公司如今不再为温饱而四处奔波，也已经从“卖人”转向出售自有产品。随着OpenStack创业公司规模不断扩大，原有人员远远不能满足需求，熟悉OpenStack开发的人员更是难寻，除了在“云圈子”寻找人才外，很大一部分是从Java、C++等专业人才中进行招聘，这些新人进入公司后，经过一段时间的培训，才能逐渐进入OpenStack开发的角色。

缺少OpenStack开发人员的原因主要有以下几方面。

1) 开发人员首先要熟悉Linux操作系统，然后要熟悉OpenStack的相关概念，同时需要具有OpenStack相关安装部署经验。

2) 开发人员要有良好的英文基础，能够看懂官方的英文文档。

3) OpenStack的开发语言为Python，很多高校没有开设相关课程，需要开发人员自己学习。

4) 开源软件需要用到很多开源工具，包括Git、Curl、Jenkins等，这些工具都需要开发人员掌握。

这些条件和要求整合到一起，难度就陡然增加了，一名普通计算机专业的本科毕业生要

从事 OpenStack 开发相关工作，至少要在 OpenStack 的开发企业中培训学习 3 个月左右；而一个云计算的开发企业，每次招收的毕业生、培养的新人是有限的，加在一起不超过千人，这些人员远远不能满足市场的需求。由此看来，OpenStack 开发人员还是非常紧缺的，其待遇也是很不错的。

本书的主要内容和特色

本书是根据北京海云捷迅科技有限公司内部培训教程重新整理编辑而成，内容包括 Python 语言基础及开发优化的原则和 OpenStack 组件及开发方面的知识，是一本系统、全面、讲解深入的开发类书籍，值得所有 Python 和 OpenStack 从业人员学习参考。本书特色如下：

- 1) 有针对性地讲解了与 OpenStack 相关的 Python 专业知识。
- 2) 全方位涵盖了 OpenStack 的技术知识，包括 DevStack、开发基础、API 调用、源码及结构分析、功能扩展等。
- 3) 堪称零基础入门。本书包含详尽的操作步骤，还有详细图示操作指引及错误分析，带领读者步入 OpenStack 开发的殿堂。
- 4) 第一次公开 OpenStack 开发企业内部资料——工程师的成长资料及资深工程师的日常工作记录，极具参考价值。

本书的知识体系

全书共分八个部分：

第一部分（第 1、2 章）：首先以若干名企业内部 OpenStack 开发人员的成长历程为例，讲解 OpenStack 开发人员应具备的知识体系；然后讲解了基本开发环境的搭建。

第二部分（第 3、4 章）：不同于市面上的 Python 教材，本书以企业内部教程为主线，从企业应用角度讲解 Python 基础知识、优化原则和 OpenStack 中 Python 相关的模块知识。

第三部分（第 5～7 章）：讲述了消息队列、DevStack 开发环境及开发的相关知识，其中包含一些基本概念和技巧，是 OpenStack 开发人员必须掌握的入门知识。

第四部分（第 8、9 章）：本部分首先用很多实例讲解了 Horizon 的开发框架 Django，然后讲了 Horizon 界面结构，分析了 Horizon 的源代码，最后用示例完成 Horizon 的开发测试过程。

第五部分（第 10 章）：本部分首先详述了 Nova 启动实例的流程，接着分析了 Nova 的源码，最后用示例讲述了微模块的开发过程。

第六部分（第 11 章）：本部分首先讲述了 OpenStack 的部署拓扑和实例的数据流向分析，接着对 Neutron 的启动流程进行了跟踪分析，最后扩展了 Neutron 的 extensions 模块并测试其功能。

第七部分（第 12 章）：本部分主要讲述了 OpenStack 测试的相关内容，包含开发人员个人的单元测试和团队的模块集成测试技术，为代码整合打好基础。

第八部分（附录 A ~ D）：首先附录 A 讲述了国内企业对 OpenStack 社区的贡献；附录 B、C 分别对 OpenStack 的 Newton 和 Ocata 版本新特性进行了说明；附录 D 列举出 Git 代码管理命令供读者参考，希望可以为读者指明学习的方向。

本书面向的读者

本书适用于想从事 OpenStack 开发及深入学习 OpenStack 的相关人员。本书也可以作为 OpenStack 的培训教材。

如何阅读本书

OpenStack 强调实践，希望读者能够把书中的示例代码都手动输入一遍，在输入运行代码的过程中可能会出现输入错误、语法错误等，读者可以自己尝试独立解决相关问题，逐渐增加自己的编程经验。

一定要去阅读和分析官方的源码，然后自己尝试去修改源码，只有这样才能锻炼自己的开发能力，逐步成为优秀的 OpenStack 开发人员。

致谢

本书参考了部分官方文档以及北京海云捷迅科技有限公司的内部资料，得到了该公司周征晟、张加龙、叶东灿、郭姗、伍军、徐鼻等开发人员的大力支持；还要感谢北京海云捷迅科技有限公司的张征宇和李华，在他们的大力支持下本书才得以完成。

感谢云技术社区创始人肖力老师，是肖老师推荐了华章分社杨福川副主编，还要感谢出版社其他工作人员，有了他们的共同努力，本书才得以正式出版。

由于编撰时间有限，加上 OpenStack 博大精深，版本更迭较快，文中有失效或分析不对的地方还请读者批评指正。

目 录 *Contents*

序 前 言

第1章 OpenStack企业工程师的成长 及工作介绍····· 1

- 1.1 Horizon 界面工程师的工作内容····· 1
- 1.2 Neutron 网络工程师的工作内容····· 2
- 1.3 Nova 资深工程师的工作内容····· 3
- 1.4 本章小结····· 5

第2章 开发环境的搭建····· 6

- 2.1 Windows 开发环境的搭建····· 6
 - 2.1.1 Git 的安装····· 6
 - 2.1.2 JDK 的安装与配置····· 12
 - 2.1.3 Python 的安装与配置····· 15
 - 2.1.4 PyCharm 的安装与配置····· 19
- 2.2 Linux 开发环境的搭建····· 24
 - 2.2.1 Vim 编辑器····· 24
 - 2.2.2 PyCharm 编辑器····· 25
- 2.3 Eclipse 开发环境的搭建····· 26
 - 2.3.1 安装 Eclipse 的 PyDev 插件····· 26
 - 2.3.2 安装 Eclipse 的 EGit 插件····· 26

- 2.4 Launchpad 账号····· 26
 - 2.4.1 Launchpad 账号注册····· 26
 - 2.4.2 上传 SSH keys····· 27
 - 2.4.3 补充相关账号与信息····· 30
- 2.5 Git 的使用····· 32
 - 2.5.1 设置 Git 全局配置····· 32
 - 2.5.2 安装 git-review 工具并验证····· 33
 - 2.5.3 提交代码步骤····· 33
 - 2.5.4 Git 管理流程图····· 35
- 2.6 本章小结····· 35

第3章 Python语言基础····· 36

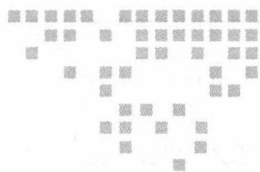
- 3.1 Python 概述····· 36
- 3.2 Python 基础····· 37
 - 3.2.1 编码····· 37
 - 3.2.2 缩进与分隔····· 37
 - 3.2.3 命名规范····· 38
 - 3.2.4 注释····· 39
 - 3.2.5 执行····· 39
 - 3.2.6 代码的调试····· 40
 - 3.2.7 帮助的使用····· 41
- 3.3 Python 数据类型····· 42

3.3.1	变量	42	4.2.1	logging 的使用	70
3.3.2	数字与表达式	44	4.2.2	若干重要的概念	71
3.3.3	字符串和列表	45	4.2.3	logging 工作流程	73
3.3.4	元组	49	4.2.4	日志配置	74
3.3.5	字典	50	4.3	Eventlet	77
3.3.6	其他类型	50	4.3.1	greenlet	77
3.4	流程控制	51	4.3.2	GreenThread	78
3.4.1	if 语句	51	4.4	WSGI	81
3.4.2	while 循环	52	4.4.1	RESTful API 介绍	81
3.4.3	for 循环	53	4.4.2	WSGI 简介	82
3.4.4	continue 和 break	53	4.4.3	简单的 WSGI	83
3.4.5	异常处理	53	4.4.4	运行 WSGI 服务	84
3.4.6	else 的特殊用法	54	4.4.5	Middleware	86
3.5	函数	55	4.5	PasteDeploy	87
3.5.1	函数基础	55	4.5.1	PasteDeploy 安装	87
3.5.2	函数参数	55	4.5.2	Paste 配置文件	87
3.5.3	函数体和返回值	57	4.5.3	改进 WSGI	89
3.5.4	再论类型和循环	58	4.6	WebOb	90
3.5.5	生成器	60	4.6.1	Python 装饰器说明	90
3.5.6	可调用的变量	62	4.6.2	简单示例	91
3.5.7	变量作用域	63	4.6.3	添加自己的中间件	91
3.6	本章小结	64	4.6.4	其他特性	92
			4.7	routes 模块	92
第4章	Python 模块	65	4.7.1	routes 模块概述	92
4.1	SQLAlchemy	65	4.7.2	使用示例	93
4.1.1	安装 SQLAlchemy	65	4.8	综合实例	94
4.1.2	使用 SQLAlchemy 对数据库 操作	65	4.9	本章小结	98
4.1.3	使用 ORM	67	第5章	RabbitMQ	99
4.1.4	完整示例	69	5.1	RabbitMQ 简介	99
4.2	logging 模块	70	5.2	RabbitMQ 流程	101

5.2.1	单向发送消息	101	7.1.1	screen 的进入	139
5.2.2	队列	103	7.1.2	screen 的使用	140
5.2.3	交换	106	7.1.3	screen 的退出	142
5.2.4	路由	109	7.2	API 调试	142
5.2.5	主题	111	7.2.1	API	142
5.2.6	RPC 调用	114	7.2.2	RESTClient 工具	145
5.3	RabbitMQ 持久化	119	7.2.3	Curl 实验	151
5.4	本章小结	120	7.2.4	错误分析	155
第6章	DevStack	121	7.3	OpenStack 界面主题的修改	157
6.1	DevStack 介绍	121	7.3.1	替换 logo	157
6.2	操作系统的安装与配置	121	7.3.2	修改样式	159
6.2.1	VMware 虚拟机的配置	121	7.4	本章小结	164
6.2.2	系统配置	123	第8章	Django 框架	165
6.3	环境准备与配置	126	8.1	安装 Django	165
6.3.1	DevStack 准备	126	8.2	Django 的架构	169
6.3.2	配置 local.conf 文件	127	8.3	创建项目	170
6.4	安装 DevStack	128	8.3.1	Django 管理工具	171
6.5	环境验证	128	8.3.2	测试站点	172
6.5.1	登录验证	128	8.3.3	增加一个 APP	174
6.5.2	创建网络	130	8.4	使用数据库	176
6.5.3	创建实例	132	8.4.1	连接数据库	176
6.5.4	验证实例网络	134	8.4.2	创立模型	177
6.5.5	OpenStack 版本	136	8.4.3	显示记录	179
6.6	错误分析	136	8.5	模板	180
6.6.1	日志目录	136	8.5.1	使用模板	180
6.6.2	使用 Git 下载代码出错	137	8.5.2	流程分析	182
6.6.3	网络配置错误	137	8.5.3	使用循环	182
6.6.4	重新部署	137	8.5.4	模板继承	184
6.7	本章小结	138	8.6	表单	185
第7章	开发小试	139	8.6.1	提交表单 (GET 方法)	185
7.1	screen 简介	139			

8.6.2 提交表单 (POST 方法)·····	186	9.5.3 JavaScript·····	224
8.7 存储数据·····	187	9.6 错误分析·····	225
8.8 表单对象·····	188	9.7 本章小结·····	227
8.9 admin 管理·····	190	第10章 Nova组件 ·····	228
8.9.1 默认管理·····	190	10.1 组件介绍·····	228
8.9.2 复杂的模型·····	191	10.1.1 OpenStack 的版本·····	228
8.9.3 自定义页面·····	192	10.1.2 组件的组成·····	228
8.9.4 Inline 显示·····	193	10.1.3 组件之间的关系·····	229
8.9.5 列表页的显示·····	194	10.1.4 API 的使用·····	230
8.10 错误分析·····	196	10.2 虚拟创建过程详解·····	230
8.11 本章小结·····	198	10.3 Nova 源码分析·····	233
第9章 Dashboard的开发 ·····	199	10.3.1 目录结构·····	233
9.1 Dashboard 概述·····	199	10.3.2 重要目录详解·····	236
9.2 源码解析·····	200	10.4 Nova 调用关系分析·····	240
9.2.1 UI 整体结构·····	200	10.4.1 创建虚拟机过程源码追溯··	240
9.2.2 初识 Dashboard·····	201	10.4.2 创建虚拟机过程调用流程	
9.2.3 项目结构·····	203	函数分析·····	241
9.3 自定义 Dashboard 和 Panel·····	207	10.4.3 Nova 和 AMQP 的关系··	243
9.3.1 手工启动 Dashboard 工程··	207	10.5 开始实践·····	246
9.3.2 创建 Dashboard·····	207	10.6 API V2.1 说明·····	248
9.3.3 创建 mypanel·····	208	10.6.1 compatibility·····	249
9.3.4 编写代码·····	209	10.6.2 Validation·····	249
9.3.5 验证代码·····	215	10.6.3 Microversion·····	249
9.4 复杂的 action table·····	217	10.7 编码·····	251
9.4.1 定义 view·····	217	10.7.1 示例编码·····	251
9.4.2 增加 URL·····	219	10.7.2 官方代码参考·····	252
9.4.3 定义 action·····	220	10.7.3 Nova 代码实现示例·····	258
9.4.4 测试·····	220	10.8 调试并修复 nova-compute 的	
9.5 Horizon 开发相关内容·····	222	所有协程卡住的问题·····	262
9.5.1 汉化·····	222	10.9 本章小结·····	269
9.5.2 workflows·····	222		

第11章 Neutron组件	270	11.4.1 编写代码	293
11.1 Neutron 基础	270	11.4.2 重启服务	295
11.1.1 服务及组件介绍	270	11.4.3 测试	296
11.1.2 Neutron 架构	271	11.5 本章小结	297
11.1.3 Neutron 抽象出的概念	272	第12章 测试技术	298
11.1.4 Linux 网络基础	272	12.1 OpenStack 测试基础	298
11.2 实例网络详解	273	12.2 单元测试	299
11.2.1 硬件配置及网络拓扑	273	12.3 集成测试	301
11.2.2 各个节点服务关系及 说明	274	12.4 本章小结	303
11.2.3 场景举例	277	附录A Mitaka国内代码贡献	304
11.3 Neutron 的源码分析	282	附录B Newton新特性	307
11.3.1 目录结构	283	附录C Ocata版本新特性	309
11.3.2 重要目录详解	285	附录D Git代码管理命令	314
11.3.3 Neutron 启动分析	288		
11.4 Neutron 组件扩展	293		



OpenStack 企业工程师的成长及工作介绍

本章以两名 OpenStack 开发工程师的成长历程及一名资深工程师的部分工作内容为例，通过了解不同 OpenStack 开发工程师的工作内容来分析工程师的成长历程。

1.1 Horizon 界面工程师的工作内容

该工程师大概 12 月份入职，入职之前没有 Python 语言的使用经历，现在主要从事 Horizon 界面的开发工作，他的工作内容可以参考表 1-1。

表 1-1 Horizon 界面工程师的工作内容

时间段	主要工作内容	小结
12.2 ~ 12.23	主要研读代码	第一个月主要是研读代码并开始学习 Python 语言
12.24 ~ 1.6	学习内部 Python 课程	
1.7 ~ 1.8	搭建 RDO 环境	第二个月已经学完 Python 语言，开始安装、部署 OpenStack
1.12 ~ 1.13	调度 VPN 页面代码	
	手动部署 L 版 OpenStack	
2.1 ~ 2.3	研读代码并总结	
2.4 ~ 2.15	2016 年中国新年假期	第三个月提高了研读代码的速度，已经学会使用 Github，开始查看 bug 并尝试修改
2.16 ~ 2.17	搭建 DevStack 环境，做了一些测试，研读代码	
2.22 ~ 2.29	调试 bug#1547998，参加 Ansible 培训	
3.1 ~ 3.7	研读代码，更新 patch，总结整理以前学习的笔记	第四个月可以看到该工程师开始修改 Horizon 界面。学习的渠道包括自学、内部培训等
3.9 ~ 3.10	研读代码，复现 bug1325397	
3.11 ~ 3.15	研究怎么在前端的 form 中增加 workflow	

(续)

时间段	主要工作内容	小结
3.16 ~ 3.23	调研 Horizon 样式改版	第四个月可以看到该工程师开始修改 Horizon 界面。学习的渠道包括自学、内部培训等
3.24 ~ 3.30	学习阅读关于前端测试的文档, 修改 base HTML 模板, 继续完善 Horizon 样式改版教程, 写 Horizon 开发步骤	

1.2 Neutron 网络工程师的工作内容

该工程师大概 2016 年 3 月份入职, 入职之前有相关的工作经验, 没有 Python 语言的使用经历, 现在主要从事 Neutron 网络相关的开发工作, 他的工作内容可以参考表 1-2。

表 1-2 Neutron 网络工程师的工作内容

时间段	主要工作内容	小结
3.14 ~ 3.18	办理入职手续; 安装操作系统及相关软件; 注册 OpenStack 社区开发相关账号; 在 Stackalytics 中提交个人信息的代码; 熟悉 Git 的使用方法; 通过 DevStack 单点安装 OpenStack	第一个月因为有关工作经验, 开始注册账号熟悉工具的使用, 开始搭建并使用 OpenStack
3.21 ~ 3.25	安装 ubuntu14.04.4 64bit 虚拟机; 安装各组件; 理解核心组件的功能	
3.28 ~ 3.31	熟悉 DevStack 中的相关脚本 (stack.sh/unstack.sh/rejoin-stack.sh/local.conf); 熟悉虚拟机的使用及排错方法	
4.1 ~ 4.6	熟悉 Linux 网络虚拟化相关技术 (network namespace、TAP/TUN、VETH); 参考 K 版本的安装部署文档, 在控制、网络、计算这 3 个节点上部署 L 版本的 OpenStack; 理解 OpenStack 中的 4 种网络类型 (管理、数据、外部、API) 和 Neutron 支持部署的 5 种网络拓扑; 理解 Neutron 二层网络服务实现原理	第二个月主要学习 Linux、OpenStack 网络的相关内容
4.7 ~ 4.12	写 Linux 网络虚拟化的 PPT (Network namespace 部分); 画 OpenStack 环境中 network 和 compute 节点的内部网络流程图; 熟悉 OVS 的使用方法; 熟悉 ovs-ofctl 的使用方法, 理解 compute 节点中 br-int、br-tun 的流表, 并画出其具体流程图; 写 Linux 网络虚拟化的 PPT (OVS 部分)	
4.13 ~ 4.16	理解 OpenFlow 协议 (v1.3); 写 Linux 网络虚拟化的 PPT (iptables、OpenFlow 部分); 熟悉 netfilter/iptables (基本用法、自定义链、Neutron 中自定义链的初始化); 写 Linux 网络虚拟化的 PPT (iptables 部分)	
4.17 ~ 4.25	熟悉 OpenStack Docs: OpenStack Networking Guide、Open vSwitch (ovs+vlan、ovs+gre);	

(续)

时间段	主要工作内容	小结
4.17 ~ 4.25	熟悉 Neutron 的配置文件 (ml2_conf.ini、openvswitch_agent.ini)、Linux Bridge (linux_bridge+vlan、linux_bridge+vxlan)、使用分布式虚拟路由实现高可用 (DVR); 熟悉策略路由 (IP rule)	第二个月主要学习 Linux、OpenStack 网络的相关内容
4.26 ~ 4.28	熟悉 DVR 中数据包的流程; 通过 DevStack 部署多节点 (controller 节点和 network 节点)	
5.3 ~ 5.4	Python 基础学习 (数据结构、类和对象)	第三个月开始学习 Python 语言, 阅读 OpenStack 源代码, 提交发现的 bug
5.5 ~ 5.10	理解 Neutron 代码; 理解 Neutron 的软件实现 (WSGI 和 RPC); 熟悉 Neutron 的代码实现	
5.11 ~ 5.14	结合 Neutron 的代码, 熟悉 OpenStack 的通用库 oslo.messaging、oslo.config; 熟悉 Neutron 中 ML2 plugin 与数据库的交互流程, 并理解 Type Manager、Mechanism Manager、Type Driver、Mechanism Driver 的实现机制; 理解 Neutron 中的 ovs_neutron_agent.py 代码 (OVS 网桥的初始化 (br-int、br-tun、br-ex) 以及各个网桥的流表的初始化); 学习 Neutron 相关的 Web 框架 (Paste+Paste Deploy+Routes+WebOb)	
5.16 ~ 5.24	学习 MySQL 数据库的基本操作 (MySQL 数据库入门); 学习 Python 数据库的编程 (MySQL DB 模块); 学习在 OpenStack 社区中修补 bug 的流程; OpenStack 社区 Neutron bug1580927 的验证与处理, 涉及 RFC3021 在 IPv4 点对点连接中使用 31 位前缀; OpenStack 社区 Neutron bug1580927 的修改与提交; 熟悉 Neutron 中 l3-agent 相关代码实现; 查找并整理 SDN 应用相关的资料 (SDN 技术、标准、产业发展现状, SDN 技术在电力光通信网络中的应用研究); 熟悉 Neutron 中相关代码的实现	
5.25 ~ 5.31	Neutron DVR 中公网 IP 地址浪费问题的调研 (fg 口); 服务器上搭建测试环境, 通过 Virt-Manager 安装虚拟机 (创建 controller、network、compute1 节点); 定位现场问题: 在现场某种应用场景下, 导致 LAN 间无法正常通信的原因是在 Neutron 中为了防止 IP 地址欺骗的过滤规则 (接口 IP&MAC 绑定) 丢掉相关报文; 理解 Neutron 中的 port_security、security_group; 分析 Neutron 中的 ovs+vlan 数据流程 (Neutron 网络实现: ovs+vlan); 理解 Neutron 中的 Linux 路由 (Neutron 网络实现: Linux 路由基础)	

1.3 Nova 资深工程师的工作内容

该工程师拥有 4 年多的 Nova 开发经验, 现在主要从事 Nova 相关的开发工作, 他的工作内容可以参考表 1-3。

表 1-3 2016 年 5 月份工作日报

时间段	主要工作内容
6.1 ~ 6.7	<ol style="list-style-type: none"> 1. 继续开发某公司三期项目的功能, 基本完成 snapshot 的相关架构和设计文档, 准备进行代码的编写。 2. 调查某公司环境指定 datastore 无法创建 VM 的问题。主要原因是该公司自己在 VMware 设置了相关的策略, 导致存储无法创建。 3. 调查某公司环境无法创建 image 的问题, 原因在于某公司用于创建 image 的 VM 太大, vsphere 超时, 导致创建失败。 4. 搭建 Mitaka 环境, 为 AWSTACK 2.0 底层架构梳理比较合适的配置项。 5. 调查某公司环境出现的问题, 确定该公司的环境存在部分 datastore 无法创建虚拟机。 6. 完成 Mitaka 版本环境的搭建, 完成相关配置文件的整理。 7. 处理某公司无法进行虚拟机迁移的问题, 问题的原因在于该公司进行迁移时, 缺少几个必需的参数, 导致迁移失败。 8. 处理某公司无法挂载 volume 的问题, 问题的原因在于该公司进行 volume 的挂载时, 本身只有 v1 的 endpoint, 但是, 在调用的时候使用的是 v2 的 endpoint, 导致无法访问, 从而挂载失败
6.8 ~ 6.14	<ol style="list-style-type: none"> 1. 解决某公司 VM resize 的问题, 结果发现并没有出现所说的情况。 2. 为同事讲解在 Nova 当中开发的新功能——显示 ebs 虚拟机 image。 3. 为同事讲解 Nova 当中开发的新功能——设置 VM 的 qos。 4. 开始分析 project_admin 的角色定位以及权限管理。 5. 完成 project_admin 的角色定位以及权限管理的基础 demo 版本。 6. 为同事讲解 Keystone 中关于权限管理的部分。 7. 完成 project_admin 的角色定位以及权限管理, 并且重写 policy 策略, 屏蔽对 admin 的 disable 操作。 8. 为同事讲解 Nova license 和其他相关功能的概念和基础逻辑。 9. 为同事讲解 Keystone 中关于权限管理的部分
6.15 ~ 6.17	<ol style="list-style-type: none"> 1. 重新部署 Mitaka 环境, 为 AWSTACK 2.0 配置符合业务需求的权限策略。 2. 为同事讲解 Nova 的代码, 包括 drs、定时任务等。 3. 为同事解决 Keystone 方面的问题, 发现其调用 Keystone 的 API 有误, 只调用了一部分 API。 4. 为同事讲解 Nova 的代码, 包括 live-resize 等新功能。 5. 重新规划权限系统, 开始实现 AWSTACK 2.0 所需要的 monitor 角色权限。 6. 完成 Keystone 的 monitor 角色权限配置, 开始设计 Nova 的 monitor 权限。 7. 实现 AWSTACK 2.0 所需要的 monitor 角色权限, 但是, 发现 monitor 权限在目前的 OpenStack 当中, 由于部分项目固化在代码当中, 而不是完全交给 policy 策略文件管理, 从而导致某些项目下 monitor 角色的权限管理失效。 8. 查找 AWSTACK 2.0 当中, admin 获取的 token 没有 catalog 存在的问题。结果发现是 kolla 在初始化 Keystone 之时, 并没有为使用 keystone-manage 创建的 admin 用户添加相关的 project, 导致无法获取到 catalog
6.18 ~ 6.22	<ol style="list-style-type: none"> 1. 完成所有权限策略文件的编写, 并提交公司代码仓库。 2. 修复社区 bug (https://launchpad.net/bugs/1534052), 并提交相关的 lite-spec (https://review.openstack.org/#/c/324195/)。 3. 处理某公司无法进行 VM 迁移的问题, 原因是他们的开发人员调用接口错误。 4. 某公司项目支持, 添加接口支持, 使之能够查询到 VM 所在 ESXi 和 datastore。 5. 创客问题排查, 最后发现是 Neutron 存在部分的问题。 6. 公司项目支持, 和同事讨论存储迁移的开发方案。 7. 调研某 stack 的功能, 整理我们所需要的功能点