

云计算时代不可多得的运维好书，详述大规模容器与大数据平台运维实践，
业内数十名专家一致好评

Broadview[®]
www.broadview.com.cn

*PaaS implementation
and Operation
management*



PaaS实现与运维管理

基于Mesos+Docker+ELK的实战指南

余何 编著



中国工信出版集团



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn



PaaS实现与运维管理

基于Mesos+Docker+ELK的实战指南

余何 编著

电子工业出版社

Publishing House of Electronics Industry

北京•BEIJING

内 容 简 介

本书主要阐述了云计算中 PaaS 平台的实现与运维管理，分为四大部分，分别是概念模型、基础资源、平台实现与运维管理，共十五章。第一部分阐述了运维与开发之间的关系、这层关系存在的矛盾，以及 PaaS 是如何有效缓解其矛盾的；第二部分勾勒出了数据中心的计算、网络、存储三大资源的主干，避免让人陷入上层的种种产品中；第三部分通过开源产品来构建一个完整的 PaaS 平台，包括资源管理、任务调度、计算单元打包、分布式协调、日志集中等；第四部分对运维管理进行了实践。

本书适合运维工程师、运维管理人员，以及希望在 PaaS 上运行分布式、可伸缩、高可用的后端开发工程师阅读。

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有，侵权必究。

图书在版编目（CIP）数据

PaaS 实现与运维管理：基于 Mesos + Docker + ELK 的实战指南 / 余何编著. — 北京：电子工业出版社，2016.2

ISBN 978-7-121-27502-9

I. ①P… II. ①余… III. ①程序语言—程序设计 IV. ①TP312

中国版本图书馆 CIP 数据核字（2015）第 262151 号

责任编辑：孙学瑛

印 刷：北京嘉恒彩色印刷有限公司

装 订：北京嘉恒彩色印刷有限公司

出版发行：电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本：787×1092 1/16 印张：27.75 字数：620 千字

版 次：2016 年 2 月第 1 版

印 次：2016 年 2 月第 1 次印刷

印 数：3000 册 定价：79.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888。

质量投诉请发邮件至 zltts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线：（010）88258888。

推荐语

本书最大的特点是理论联系实际，深入浅出地讲解了 PaaS 的实现方法，涉及当下非常热门的 Docker、Mesos 技术，更难能可贵的是，作者同时分享了珍贵的运维实践经验，我为读者能看到本书的精彩内容而感到高兴。

“KVM 虚拟化实践”公众号维护者、珠海金山西山居系统运维经理 肖力

每一名 IT 工程师都有自己的 PaaS 平台梦，这才有近年来风行的 GAE、CloudFoundry、Docker。然而 PaaS 要在企业中应用推广，绝不是改个容器或虚拟机那么简单。本书以 PaaS 为题，内容却覆盖了平台建设的理论基础、技术实现、配套系统和流程管理，是作者多年实践经验的精华所在，堪称大型企业应用运维平台化的指导用书。

微博运维团队系统架构师 饶琛琳

PaaS 越来越受到重视，它是未来真正的高效运维利器。本书作者有着丰富的一线运维经验，在本书中从 PaaS 平台的多个角度阐述了其核心原理与技术实现。值得称赞的是，作者还给出了应用系统如何适应 PaaS 及对 PaaS 应用的运维管理，让我们对 PaaS 平台有了系统而全面的认识。

优维科技创始人、互联网运维杂谈 王津银

国内有不少运维同行在自己的业务领域尝试引入私有 IaaS 技术，因此催生了不少分享 IaaS 体系理念及实践的资料和书籍。伴随着愈演愈烈的竞争，互联网业务对运营及运维能力的要求也会不断攀升，大家会慢慢意识到私有云的概念停留在 IaaS 层面是远远不够的。国内分享 PaaS 体系的理念及实践的资料和书籍并不多，本书不但对 PaaS 理论及基础概念介绍得比较清楚，而且以平安科技的实践经验为背景，对广大运维及运营开发同行会有很大的帮助。

腾讯游戏蓝鲸产品中心总监 咖啡党

云服务是当前炙手可热的话题，现在业务使用云服务是趋势，百度内部的业务线也很早就推广 Matrix、PaaS 技术。本书内容通俗易懂，讲解由浅至深，读者既能从书

中了解到云服务的现状，又能从细节上了解 PaaS 的实现原理。作者本身对 PaaS 有着丰富的实战经验，结合书中丰富的源代码，能使读者快速踏入 PaaS 领域。

百度国际化首席架构师、广告变现技术团队负责人 谢朴锐

相较 IaaS 而言，PaaS 更为丰富、可操作性更强，作者精选多种流行、前沿的 PaaS 工具、产品及运维自动化工具，再辅以基础理论知识和个人多年运维行业技术经验写成本书。本书文字朴实易懂、图文并茂，适合运维同仁们细细阅读。

开放运维联盟联合主席、高效运维技术社区创始人 萧田国

推荐序一

第一次见作者是在 2007 年年中，他刚到平安科技入职。初次交谈，他给我最大的印象是虚心好学，积极上进。当时为了提升团队整体的计算机专业水平，我们进行了约一年的内部相互学习和培训，重点放在计算机最底层的计算机组成和程序运行原理上。记得第一次安排他做的培训题目是从软件破解的角度去了解计算机软件。他用了两周来学习和准备就可以大致讲明白编译运行类程序的内存布局和进行破解的方法。这种自学的激情和效率不是所有人都能做到的，正是这种好学上进的精神成就了今天的他。其实，要把计算机方面的工作做到极致，必须要有这种精神。

作者与我在一个 IT 团队共事五年，我们一起经历过系统运维一线的种种复杂而烦琐的工作：应用环境构建、问题应对、重大事故处理、运维管理……我们曾经通宵达旦地解决事故背后的疑难问题，曾经为解决运维中的资源管理、监控、自动化、工作协同等问题一起设计并开发运维工作平台。多年前，我们有一个想法：我们的团队亲身经历所获取的经验和教训，整理下来就是一本很好的书。一年前，作者提到自己准备写一本反映 IT 运维工作本源和真谛的书。当时觉得，IT 运维工作者确实需要这样一本书。市面上的运维书籍多在讲解如何去使用某个软件或者系统，却很少提及如何面对多种多样的系统、软件、硬件，更谈不上从本质和道理层面上去讲解运维的本真。对于任何一个事物，如果我们掌握了它的本质，则应对起来会更加游刃有余。好比程序员读懂了 *Programming from ground up*，OS 管理员读懂了《计算机的心智——操作系统之哲学原理》，创业者读懂了《创业维艰》。虽然，现在本书把重点放在了 PaaS，但多从计算机技术本质角度提出问题和解决问题，也算是回归了运维本真的思路。

IT 运维有两种复杂度：一是应用规模大，一个应用要应对海量访问，例如上了规模的互联网运维；二是应用数量多，碎片化小应用特别多，例如大企业中大量的异构小应用和复杂的网络拓扑。这两种复杂度带来的问题总是让人头疼。面对这些复杂的问题，如何选择合适的技术和方法进行应用的快速构建、资源配置、信息管理、监控、操作自动化等，并没有一致的答案。要有答案，需要你充分考虑和分析所在环境的团队素质、外部资源支持、应用特点等因素。作者以自己的亲身工作经验为背景，对这些重要内容进行了讲解。

引用爱因斯坦的一句话：Any intelligent fool can make things bigger, more complex, and

more violent. It takes a touch of genius and a lot of courage to move in the opposite direction, 意思是自命不凡的学者总会把事物变得更大、更复杂和更剧烈，而在相反方向上努力是需要不凡的天赋和极大的勇气的。

IT 运维工作者，你懂的！

TechSure 创始人兼 CEO 温海波

2015 年 10 月

推荐序二

云计算正在中国普及，在 IaaS、PaaS、SaaS 的三层服务里：IaaS 的标准相对成熟；SaaS 百花齐放、新应用层出不穷；PaaS 作为衔接 IaaS 与 SaaS 的平台服务层，现在越来越受到人们的重视，专门提供 PaaS 的创业公司也越来越多的了。

本书介绍了比较常见和重要的 PaaS 系统，包括基于容器的操作系统虚拟化技术 Docker、分布式协调系统 ZooKeeper、资源管理系统 Mesos、服务调度框架 Marathon、大数据处理架构 Spark、日志搜索分析系统 ELK 等。Docker、Mesos、Spark、ELK 等系统在美国都有专门的创业公司如 Docker、Mesosphere、Databricks、Elastic 等知名公司在开发并提供技术支持服务，这些公司受到了风险投资界的追捧。更有专门的公司开发和维护这些系统，这是对其生命力和前途的背书。这些系统已经在业界得到广泛应用，每一位对云计算感兴趣的技术人员都应该了解这些系统。

以日志管理为例，一家公司的服务器、网络设备、应用系统每时每刻都在产生日志，大公司的 IT 系统可能每秒产生超过 10 万条日志，每天产生的日志量达到 TB 级。这些日志包含了极其重要的运维信息和业务信息，但分散在服务器和网络设备上，需要登录每台设备上查看，存储空间满了就被覆盖、删除，管理非常不方便。一些大公司建立了日志管理系统，把分散在各台设备上的日志采集上来，集中管理，并提供查询、分析、统计等功能。因为日志是非结构化数据，传统的使用数据库的处理方式不适合日志处理，于是出现了 ELK 这种采用实时搜索引擎处理日志的开源系统。本书详细介绍了 ELK。

另外，本书也涵盖了数据中心运维技术和管理如配置、监控、变更等，对 GAE、Cloud Foundry、Heroku 等国外热门的 PaaS 也有介绍，对运维工程师大有裨益。

本书作者在平安科技从事运维管理工作多年，经历了平安科技从金融 IT 到互联网金融的转变。互联网强调快速迭代，金融 IT 强调稳定合规，能把这对矛盾体结合在一起，实属不易。作者在这方面积累了丰富的经验，现在把这些宝贵经验分享出来，是对 IT 运维界的贡献，希望本书能够对云平台运维工程师有所帮助。

日志易创始人兼 CEO 陈军

2015 年 9 月

古往今来随风去，书剑相伴两袖空

笔者在运维领域耕耘十余载。2007 年加入平安集团旗下的科技公司，2011 年主导了业内最大的应用迁徙与架构变更，2012 年开展 IT 运维管理变革，打通横向条线，实现了技能融合。光阴荏苒，日月如梭，运维往事历历在目，我们流过汗，熬过夜，摊过事，也拿过奖，运维是一个从无到有、日积月累、不断提升的过程，也是一个需要耐得住寂寞，顶得住压力的行当，在此与正奋斗在运维一线的伙伴们共勉。

平安 IT 经历了企业信息管理部、国际领先 IT 服务公司、互联网金融三个历史阶段，每一次蜕变都紧随时代步伐，拥抱技术革新，创造业务价值。从企业 ERP、PC 互联、移动互联、云计算到大数据，运维人一直在奔袭，从未停歇。企业信息管理时期的运维人飞行于各大城市的上空，每一次发版、变更就是一次长途远行，分散的管理模式简单直接，但随着业务发展、规模扩大，企业对系统稳定性、运营成本控制的要求越来越高，随之而来的是 IT 集中管理时期，这个时期追求一流的国际 IT 服务管理水平，构建独立数据中心，实现开发架构、安全标准与运营规范的全面统一，该时期形成的 IT 管理实践堪称行业标杆。2013 年全面进入互联网金融时期，移动互联、云计算、大数据的火热发展，业务渠道、流量入口、用户终端的改变，促使我们又一次站在了历史的新起点。

互联网金融在运维方面上演了一场“双城记”。一边是充满贵族气质、为荣誉而战的“英伦君主立宪”，在这个领域内严格执行 IT 管理规范，稳若磐石，滴水不漏，保证绝对稳定与安全；而另一边是自由、民主喧哗下的“法国大革命”，为了满足互联网下业务系统的高并发、高吞吐、版本多变的需求，应用不再拘泥于一致框架、规范与准则，多样选择，以快为先。本书的核心内容就是在这样的背景下诞生的，我们无法“一刀切”地构建承载所有应用类型的基础环境，只有准确定义应用标准，以一种兼容并存的方式在应用架构上迭代地朝轻量级、微模块方向发展，通过技术、管理双层标准来调和“君主立宪”与“民主共和”之间的矛盾，转化成优势互补、经验传承、全面共赢，最终完成新时期的历史使命。这个过程是无法一步到位、一蹴而就的，而是以一种螺旋式形态探索前进的，除了在原有管理规范上引入新技术探究，还需要开发同步配合在应用架构上进行改造，基础架构将从环境运维逐渐转变成平台研发，以提供功能粒度细、服务自助化的运维平台来满足上层应用需求。

通明大道去浮华，千辛历尽回本真

PaaS 并不能解决 IT 运维的所有问题，它对应了“民主共和”的部分，提供了一个用户自助的开放形式平台来满足部分应用需求。PaaS 也不是从一开始就从解决异地容灾、流量切换等数据中心级别功能上设计的，而是在兼容当前企业应用架构、满足资源分配、计算单元打包、版本发布控制等基础功能后，再做进一步功能延伸的。PaaS 并不是一种新兴技术，而是对以往运维经验的总结，利用容器等技术对开发、运维工作流的重新编排。

IT 技术更新发展快，新词汇层出不穷，特别是在云计算、大数据、移动互联下衍生了大量的产品，让运维人应接不暇，刚刚掌握 Hadoop，又出 Spark，才标准化 ActiveMQ，又有 Kafka，OpenStack 的 IaaS 体系才得以了解，又有各种平台下的 PaaS……如果要熟练掌握每一门技艺，则将是一个疲于奔命的过程，而这个过程大部分都是表面的产品架构与配置，我们会发现一旦深入其中，所有技术却看似一致。对于运维人员来说，亘古不变的始终是底层的三大基础资源计算、存储、网络，这些知识包括：程序的链接、加载与运行；操作系统下的 CPU、内存、I/O 资源管理；基本的 TCP/IP 协议栈等。随着技术层级越多、复杂性越高，运维人员只有透过眼花缭乱的“新技术”看到其内在一致的本质，在底层长期沉淀，理解好程序是如何调用三大基础资源，找到运维技术的本真，才可以做到最终的不变应万变，这也印证了运维是一个要耐得住寂寞的行当。

最后让我们回到运维管理上，今年国内互联网上发生了几起影响比较严重的运维故障，一时间在圈内引起关于“再流弊的技术，也抵不过一次事故”等的科技杂谈，可见管理在运维上的重要性。管理是一门艺术，而这门艺术在运维上并不是形而上学、趋于神秘主义的，也不是绝对的“封”“杀”“控”，以怀疑主义、不信任方式来管控的。运维管理是人、流程与工具三方面的有机结合。“民主共和”型运维管理往往依赖于人的自觉性，而“君主立宪”则重在流程管控，朝两个极端偏倚而忽视它们与工具的结合都将不能从根本上解决问题。我们会发现越是技术实力强、自信的人越容易犯运维错误，而再完善的流程制度，如果烦琐到让人感觉“无意义”，那也就无从执行了。工具是人与流程之间的桥梁，但一定要注意这个工具不是管控的“枷锁”，而是在满足运维管理需求下关注一线运维人员用户体验的“帮手”。对于管理层，它是上层意志的体现，但对于一线运维人员，它是一个效率工具，除了包含精准配置信息、标准变更步骤，还要囊括人性的知识分享、社交互动等功能，它是为一线运维人员服务、以人为本的。

内容大纲

本书分为四大部分，分别是概念模型、基础资源、平台实现与运维管理，共十五章。各部分之间没有必然联系，读者可依据关注点和个人兴趣来阅读。对于需要系统化管理运维及 PaaS 的读者，建议遵循本书的章节顺序阅读。

第一部分——概念模型：阐述了运维与开发之间的关系、这层关系存在的矛盾，以及 PaaS 是如何有效缓解其矛盾的。这部分介绍了公有 PaaS 平台的特征，以及其开放性与约束

性，列出 12-Factor 规范来说明应用系统应当遵循的规则，这样才能适应于在 PaaS 上运行。

第二部分——基础资源：勾勒出了数据中心的计算、网络、存储三大资源的主干，避免让人陷入上层的种种产品中。对于已熟悉数据中心三大资源的运维人员来说，本部分是资源的总体概述，让你重拾内在本质。若你是一名运维新兵，则请以此为纲要寻找外部资源来继续深入学习；开发人员可通过本部分了解到日常运维工作所管理对象的基本内容。

第三部分——平台实现：通过开源产品来构建一个完整的 PaaS 平台，包括资源管理、任务调度、计算单元打包、分布式协调、日志集中等。通过学习本部分的内容，读者可以实现一个可扩展、自定义的开放 PaaS 平台，这个扩展部分包括了各自企业内部的集成部署流程、应用灰度发布、平台门户管理等方方面面的内容。

第四部分——运维管理：对运维管理进行了实践。运维管理的核心是配置管理，一个好用、易用的配置管理系统将直接影响上层监控管理、变更管理及事件管理，决定一个企业运维品质的好坏。

感谢

首先特别感谢我的太太李嘉，在过去一年里，写作几乎占用了我所有的周末及其他休息时间，你承担起了家务，并对我与余多多悉心照顾，没有你的理解、宽容与支持，本书无法完成。感谢姐夫喻立新、姐姐何碧，你们在本书插图上给予了我很大的帮助。特别感谢饶琛琳、杨永帮利用周末进行审订稿件，感谢梁山在 12-Factor 上对我的帮助，感谢策划编辑孙学瑛老师对我的鼓励，感谢责任编辑虾米（张国霞）的校对、排版与指导。

感谢我的公司平安科技，给予了我一个更大的平台，让我驰骋在 IT 运维大草原上得以一览全貌。感谢我的上级胡玮、朱永忠、李毅对我工作的支持，感谢我的同事王欣、于泳、宋楹柯、蓝景全、江锐、常明、黄文建、唐文祥、陈顺星、彭俊清、陈春润对我的帮助。感谢事件处理组的小伙伴们：王耀武、莫广华、陈秋浩、郑司营、赵宝磊、吴磊、林国峰、张浙栋、夏永燕、孟佩佩、罗颖胜、倪沛榆、丁江，感谢基础架构篮球队。感谢所有在互联网上帮助过我的朋友们。

余何

2015 年 11 月

目 录

第一部分 概念模型

第 1 章 分布式 PaaS 平台介绍	2
1.1 什么是 PaaS	2
1.1.1 开发与运维之间的困局	2
1.1.2 DevOps 的自动化	3
1.1.3 云计算的 IaaS	4
1.1.4 PaaS 的到来	4
1.1.5 PaaS 的约束与开放	4
1.1.6 PaaS 解决的具体问题	5
1.2 什么是分布式计算	6
1.2.1 分布式计算与 PaaS	6
1.2.2 分布式平台的挑战	7
第 2 章 PaaS 模型与特征	10
2.1 主流 PaaS 平台架构	10
2.1.1 谷歌 GAE	10
2.1.2 AEB	11
2.1.3 Cloud Foundry	13
2.1.4 Heroku	14
2.2 PaaS 与 12-Factor	15
2.2.1 基准代码 (Codebase)	15
2.2.2 依赖 (Dependency)	16
2.2.3 配置 (Config)	17
2.2.4 后端服务 (Backing Services)	18

- 2.2.5 构建 (Build)、发布 (Release)、运行 (Run)19
- 2.2.6 进程 (Process)20
- 2.2.7 端口绑定 (Port Binding)21
- 2.2.8 并发 (Concurrency)21
- 2.2.9 快捷性 (Disposable)22
- 2.2.10 开发/生产环境等价 (Dev/Prod Parity)23
- 2.2.11 日志 (Log)24
- 2.2.12 管理进程 (Admin Process)25
- 2.3 PaaS 与 Reaction 宣言26
 - 2.3.1 响应 (Responsive)26
 - 2.3.2 韧性 (Resilient)26
 - 2.3.3 弹性 (Elastic)27
 - 2.3.4 消息驱动 (Message Driven)28

第二部分 基础原理

- 第3章 计算资源.....30
 - 3.1 图灵机与冯·诺伊曼模型.....30
 - 3.2 服务器的种类.....34
 - 3.3 一切都是二进制37
 - 3.3.1 整数表示法.....38
 - 3.3.2 文本表示法.....39
 - 3.3.3 音频信息表示法.....41
 - 3.4 操作系统——计算机系统的指挥官42
 - 3.4.1 操作系统解决的问题.....42
 - 3.4.2 企业级操作系统.....43
 - 3.4.3 服务器虚拟化.....47
 - 3.5 进程——资源聚合的抽象体.....49
 - 3.5.1 计算单元的构建.....49
 - 3.5.2 计算请求的拆解.....51

第 4 章 网络资源	53
4.1 协议分层	53
4.1.1 OSI 网络体系模型	54
4.1.2 OSI 与 TCP/IP 协议簇	55
4.1.3 交换、选路与传输	56
4.2 网络物理设备	58
4.2.1 连线与接口	59
4.2.2 二层交换机	62
4.2.3 路由及三层交换	63
4.2.4 四~七层网络设备	64
4.2.5 现实网络构成	65
4.3 网络逻辑拓扑	65
4.4 对网络拓扑的考虑	66
4.5 对物理布线的考虑	67
4.6 网络虚拟化与 SDN	70
第 5 章 存储资源	73
5.1 俯瞰存储系统	73
5.1.1 数据存储功能分类	73
5.1.2 文件存储的三个层级	74
5.2 磁盘与磁盘阵列	77
5.2.1 硬盘的物理构造	77
5.2.2 磁盘阵列	79
5.2.3 SCSI 协议	81
5.3 存储、计算分离	82
5.3.1 磁盘柜与盘阵	82
5.3.2 FC 存储网络	83
5.3.3 FC 协议栈	86
5.3.4 FC 寻址过程	87
5.3.5 FC 交换机与适配器	88
5.3.6 FCoE 与 iSCSI	89

5.4	存储访问类型	90
5.4.1	NAS 与 SAN	90
5.4.2	分布式存储	92
第三部分 平台实现		
第 6 章	平台功能与架构	96
6.1	平台运维需求	96
6.1.1	软件配置	96
6.1.2	服务部署	97
6.1.3	服务发现	97
6.1.4	监控恢复	97
6.2	平台功能划分	97
6.3	平台高阶架构	100
6.4	企业应用迁移	102
6.4.1	企业应用很“厚重”	102
6.4.2	应用部署架构	102
6.4.3	企业应用调整	104
第 7 章	计算单元 Docker	108
7.1	Docker 介绍	108
7.1.1	Docker 是什么	108
7.1.2	Docker 术语	109
7.1.3	Docker 安装	111
7.2	Docker 容器命令	112
7.2.1	run 命令	112
7.2.2	start 命令	115
7.2.3	stop 命令	116
7.2.4	restart 命令	116
7.2.5	attach 命令	116

7.2.6	ps 命令	116
7.2.7	inspect 命令	117
7.3	Docker 镜像命令	119
7.3.1	search、pull、push 命令	120
7.3.2	commit 命令	120
7.3.3	image、diff、rmi 命令	121
7.3.4	save、load、export、import 命令	121
7.4	Docker 网络与链接	122
7.4.1	Docker 网络模式	122
7.4.2	pipework 管理网络	125
7.4.3	容器链接与数据卷	127
7.5	Dockerfile	129
7.5.1	基本指令集	130
7.5.2	环境指令集	131
7.5.3	数据指令集	132
7.5.4	ENTRYPOINT 指令	132
第 8 章	分布式协调 ZooKeeper	134
8.1	ZooKeeper 介绍	134
8.1.1	ZooKeeper 是什么	134
8.1.2	ZooKeeper 架构	135
8.1.3	数据模型	136
8.1.4	监听与通知	139
8.1.5	API 集合	139
8.1.6	会话	140
8.1.7	观察者	141
8.2	ZooKeeper 使用	141
8.2.1	快速安装	141
8.2.2	基本操作	143
8.2.3	配置参数	145
8.2.4	动态重配置	149

8.2.5	监控	152
8.3	ZooKeeper 进阶	157
8.3.1	分组与权重	158
8.3.2	Paxos 算法	159
8.3.3	ZAB 协议	163
8.3.4	分布式协调场景	165
第 9 章	资源管理 Mesos	167
9.1	Mesos 介绍	167
9.1.1	资源管理需求	167
9.1.2	Mesos 的起源	169
9.2	Mesos 架构与工作流	169
9.2.1	Mesos 架构组件	169
9.2.2	Mesos 资源管理的工作流程	170
9.3	Mesos 安装配置	172
9.3.1	安装预先准备	172
9.3.2	构建 Mesos	173
9.3.3	启动 Mesos	174
9.3.4	高可用 Mesos	178
9.3.5	Slave 移除限速	182
9.4	Mesos 运维	183
9.4.1	认证管理	183
9.4.2	监控管理	186
9.4.3	容器网络限速	192
9.4.4	Framework API 限速	194
9.4.5	Restful 接口	195
9.4.6	配置参数	196
9.5	Mesos 资源分配	197
9.5.1	DRF 算法	197
9.5.2	DRF 权重	199