内 容 简 介

Kubernetes 是由谷歌开源的 Docker 容器集群管理系统，为容器化的应用提供了资源调度、部署运行、服务发现、扩容及缩容等一整套功能。本书从一个开发者的角度去理解、分析和解决问题，囊括了 Kubernetes 入门、核心原理、实践指南、开发指导、高级案例、运维指南及源码分析等方面的内容，图文并茂、内容丰富、由浅入深、讲解全面；并围绕着生产环境中可能出现的问题，给出了大量的典型案例，比如安全问题、网络方案的选择、高可用性方案及 Trouble Shooting 技巧等，有很强的可借鉴性。

无论是对于软件工程师、测试工程师、运维工程师、软件架构师、技术经理，还是对于资深 IT 人士来说，本书都极具参考价值。

图书在版编目（CIP）数据

Kubernetes 权威指南：从 Docker 到 Kubernetes 实践全接触 / 龚正等编著. —2 版. —北京：电子工业出版社，2016.10

ISBN 978-7-121-29941-4

Ⅰ. ①K… Ⅱ. ①龚… Ⅲ. ①Linux 操作系统—程序设计—指南 Ⅳ. ①TP316.85-62

中国版本图书馆 CIP 数据核字（2016）第 225554 号

策划编辑：张国霞
责任编辑：徐津平
印　　刷：北京天宇星印刷厂
装　　订：北京天宇星印刷厂
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：787×980　1/16　印张：32.5　字数：730 千字
版　　次：2016 年 10 月第 1 版
印　　次：2016 年 10 月第 1 次印刷
印　　数：4000 册　　定价：99.00 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）88254888，88258888。

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

本书咨询联系方式：010-51260888-819　faq@phei.com.cn。

第 1 版推荐序

经过作者们多年的实践经验积累及近一年的精心准备，本书终于与我们大家见面了。我有幸作为首批读者，提前见证和学习了在云时代引领业界技术方向的 Kubernetes 和 Docker 的最新动态。

从内容上讲，本书从一个开发者的角度去理解、分析和解决问题：从基础入门到架构原理，从运行机制到开发源码，再从系统运维到应用实践，讲解全面。本书图文并茂，内容丰富，由浅入深，对基本原理阐述清晰，对程序源码分析透彻，对实践经验体会深刻。

我认为本书值得推荐的原因有以下几点。

首先，作者的所有观点和经验，均是在多年建设、维护大型应用系统的过程中积累形成的。例如，读者通过学习书中的 Kubernetes 运维指南和高级应用实践案例章节的内容，不仅可以直接提高开发技能，还可以解决在实践过程中经常遇到的各种关键问题。书中的这些内容具有很高的借鉴和推广意义。

其次，通过大量的实例操作和详尽的源码解析，本书可以帮助读者进一步深刻理解 Kubernetes 的各种概念。例如书中“Java 访问 Kubernetes API”的几种方法，读者参照其中的案例，只要稍做修改，再结合实际的应用需求，就可以用于正在开发的项目中，达到事半功倍的效果，有利于有一定 Java 基础的专业人士快速学习 Kubernetes 的各种细节和实践操作。

再次，为了让初学者快速入门，本书配备了即时在线交流工具和专业后台技术支持团队。如果你在开发和应用的过程中遇到各类相关问题，均可直接联系该团队的开发支持专家。

最后，我们可以看到，容器化技术已经成为计算模型演化的一个开端，Kubernetes 作为谷歌开源的 Docker 容器集群管理技术，在这场新的技术革命中扮演着重要的角色。Kubernetes 正在被众多知名企业所采用，例如 RedHat、VMware、CoreOS 及腾讯等，因此，Kubernetes 站在了容器新技术变革的浪潮之巅，将具有不可预估的发展前景和商业价值。

如果你是初级程序员，那么你有必要好好学习本书；如果你正在 IT 领域进行高级进阶修炼，那你也有必要阅读本书。无论是架构师、开发者、运维人员，还是对容器技术比较好奇的读者，本书都是一本不可多得的带你从入门向高级进阶的精品书，值得大家选择！

初瑞

中国移动业务支撑中心高级经理

自 序

我不知道你是如何获得这本书的，可能是在百度头条、网络广告、朋友圈中听说本书后购买的，也可能是某一天逛书店时，这本书恰好神奇地翻落书架，出现在你面前，让你想起一千多年前那个意外得到《太公兵法》的传奇少年，你觉得这是冥冥之中上天的恩赐，于是果断带走。不管怎样，我相信多年以后，这本书仍然值得你回忆。

Kubernetes 这个名字起源于古希腊，是舵手的意思，所以它的 Logo 既像一张渔网，又像一个罗盘。谷歌采用这个名字的一层深意就是：既然 Docker 把自己定位为驮着集装箱在大海上自在遨游的鲸鱼，那么谷歌就要以 Kubernetes 掌舵大航海时代的话语权，“捕获”和“指引”这条鲸鱼按照“主人”设定的路线巡游，确保谷歌倾力打造的新一代容器世界的宏伟蓝图顺利实现。

虽然 Kubernetes 自诞生至今才 1 年多，其第一个正式版本 Kubernetes 1.0 于 2015 年 7 月才发布，完全是个新生事物，但其影响力巨大，已经吸引了包括 IBM、惠普、微软、红帽、Intel、VMware、CoreOS、Docker、Mesosphere、Mirantis 等在内的众多业界巨头纷纷加入。红帽这个软件虚拟化领域的领导者之一，在容器技术方面已经完全“跟从”谷歌了，不仅把自家的第三代 OpenShift 产品的架构底层换成了 Docker+Kubernetes，还直接在其新一代容器操作系统 Atomic 内原生集成了 Kubernetes。

Kubernetes 是第一个将“一切以服务（Service）为中心，一切围绕服务运转”作为指导思想的创新型产品，它的功能和架构设计自始至终都遵循了这一指导思想，构建在 Kubernetes 上的系统不仅可以独立运行在物理机、虚拟机集群或者企业私有云上，也可以被托管在公有云中。Kubernetes 方案的另一个亮点是自动化，在 Kubernetes 的解决方案中，一个服务可以自我扩展、自我诊断，并且容易升级，在收到服务扩容的请求后，Kubernetes 会触发调度流程，最终在选定的目标节点上启动相应数量的服务实例副本，这些副本在启动成功后会自动加入负载均衡器中并生效，整个过程无须额外的人工操作。另外，Kubernetes 会定时巡查每个服务的所有实例的可用性，确保服务实例的数量始终保持为预期的数量，当它发现某个实例不可用时，会自动

重启该实例或者在其他节点重新调度、运行一个新实例，这样，一个复杂的过程无须人工干预即可全部自动化完成。试想一下，如果一个包括几十个节点且运行着几万个容器的复杂系统，其负载均衡、故障检测和故障修复等都需要人工介入进行处理，那将是多么难以想象。

通常我们会把 Kubernetes 看作 Docker 的上层架构，就好像 Java 与 J2EE 的关系一样：J2EE 是以 Java 为基础的企业级软件架构，而 Kubernetes 则以 Docker 为基础打造了一个云计算时代的全新分布式系统架构。但 Kubernetes 与 Docker 之间还存在着更为复杂的关系，从表面上看，似乎 Kubernetes 离不开 Docker，但实际上在 Kubernetes 的架构里，Docker 只是其目前支持的两种底层容器技术之一，另一个容器技术则是 Rocket，后者来源于 CoreOS 这个 Docker 昔日的“恋人”所推出的竞争产品。

Kubernetes 同时支持这两种互相竞争的容器技术，这是有深刻的历史原因的。快速发展的 Docker 打败了谷歌曾经名噪一时的开源容器技术 lmctfy，并迅速风靡世界。但是，作为一个已经对全球 IT 公司产生重要影响的技术，Docker 背后的容器标准的制定注定不可能被任何一个公司私有控制，于是就有了后来引发危机的 CoreOS 与 Docker 分手事件，其导火索是 CoreOS 撇开了 Docker，推出了与 Docker 相对抗的开源容器项目——Rocket，并动员一些知名 IT 公司成立委员会来试图主导容器技术的标准化，该分手事件愈演愈烈，最终导致 CoreOS“傍上”谷歌一起宣布“叛逃”Docker 阵营，共同发起了基于 CoreOS+Rocket+Kubernetes 的新项目 Tectonic。这让当时的 Docker 阵营和 Docker 粉丝们无比担心 Docker 的命运，不管最终鹿死谁手，容器技术分裂态势的加剧对所有牵涉其中的人来说都没有好处，于是 Linux 基金会出面调和矛盾，双方都退让一步，最终的结果是 Linux 基金会于 2015 年 6 月宣布成立开放容器技术项目（Open Container Project），谷歌、CoreOS 及 Docker 都加入了 OCP 项目。但通过查看 OCP 项目的成员名单，你会发现 Docker 在这个名单中只能算一个小角色了。OCP 的成立最终结束了这场让无数人揪心的“战争”，Docker 公司被迫放弃了自己的独家控制权。作为回报，Docker 的容器格式被 OCP 采纳为新标准的基础，并且由 Docker 负责起草 OCP 草案规范的初稿文档，当然这个“标准起草者”的角色也不是那么容易担当的，Docker 要提交自己的容器执行引擎的源码作为 OCP 项目的启动资源。

事到如今，我们再来回顾当初 CoreOS 与谷歌的叛逃事件，从表面上看，谷歌貌似是被诱拐“出柜”的，但局里人都明白，谷歌才是这一系列事件背后的主谋，其不仅为当年失败的 lmctfy 报了一箭之仇，还重新掌控了容器技术的未来。容器标准之战大捷之后，谷歌进一步扩大了联盟并提高了自身影响力。2015 年 7 月，谷歌正式宣布加入 OpenStack 阵营，其目标是确保 Linux 容器及关联的容器管理技术 Kubernetes 能够被 OpenStack 生态圈所容纳，并且成为 OpenStack 平台上与 KVM 虚机一样的一等公民。谷歌加入 OpenStack 意味着对数据中心控制平面的争夺已经结束，以容器为代表的应用形态与以虚拟化为代表的系统形态将会完美融合于 OpenStack 之上，并与软件定义网络和软件定义存储一起统治下一代数据中心。

谷歌凭借着几十年大规模容器使用的丰富经验，步步为营，先是祭出 Kubernetes 这个神器，然后又掌控了容器技术的制定标准，最后又入驻 OpenStack 阵营全力将 Kubernetes 扶上位，谷歌这个 IT 界的领导者和创新者再次王者归来。我们都明白，在 IT 世界里只有那些被大公司掌控和推广的，同时被业界众多巨头都认可和支持的新技术才能生存和壮大下去。Kubernetes 就是当今 IT 界里符合要求且为数不多的热门技术之一，它的影响力可能长达十年，所以，我们每个 IT 人都有理由重视这门新技术。

谁能比别人领先一步掌握新技术，谁就在竞争中赢得了先机。惠普中国电信解决方案领域的资深专家团一起分工协作，并行研究，废寝忘食地合力撰写，在短短的 5 个月内完成了这部厚达 500 多页的 Kubernetes 权威指南。经过一年的高速发展，Kubernetes 先后发布了 1.1、1.2 和 1.3 版本，每个版本都带来了大量的新特性，能够处理的应用场景也越来越丰富。本书遵循从入门到精通的学习路线，全书共分为六大章节，涵盖了入门、实践指南、架构原理、开发指南、高级案例、运维指南和源码分析等内容，内容详实、图文并茂，几乎囊括了 Kubernetes 1.3 版本的方方面面，无论是对于软件工程师、测试工程师、运维工程师、软件架构师、技术经理，还是对于资深 IT 人士来说，本书都极具参考价值。

吴治辉

惠普公司系统架构师

目 录