TURING 图灵原创

Docker

容器与容器云 第2版

Docker and Kubernetes under the Hood

浙江大学SEL实验室◎著

- 基于Docker 1.10和Kubernetes 1.2全面更新
- 从源码层面深度解析Docker核心原理
- 广泛涵盖Docker高级实践技巧
- 一本书讲透Docker和Kubernetes
- Kubernetes源码完全解读+最佳实践
- 全面梳理主流容器云技术架构方法

中国工信出版集团

人民邮电出版社
POSTS & TELECOM PRESS

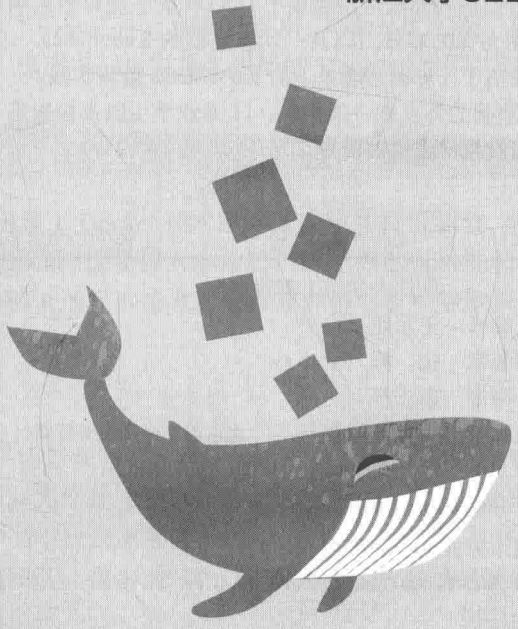
TURING 图灵原创

Docker

容器与容器云 第2版

Docker and Kubernetes under the Hood

浙江大学SEL实验室◎著



人民邮电出版社
北京

图书在版编目（CIP）数据

Docker——容器与容器云 / 浙江大学SEL实验室著. -- 2版. -- 北京 : 人民邮电出版社, 2016.10
（图灵原创）
ISBN 978-7-115-43504-0

Ⅰ. ①D… Ⅱ. ①浙… Ⅲ. ①Linux操作系统－程序设计 Ⅳ. ①TP316.85

中国版本图书馆CIP数据核字(2016)第216691号

内 容 提 要

本书根据 Docker 1.10 版和 Kubernetes 1.2 版对第 1 版进行了全面更新，从实践者的角度出发，以 Docker 和 Kubernetes 为重点，沿着“基本用法介绍”到“核心原理解读”到“高级实践技巧”的思路，一本书讲透当前主流的容器和容器云技术，有助于读者在实际场景中利用 Docker 容器和容器云解决问题并启发新的思考。全书包括两部分，第一部分深入解读 Docker 容器技术，包括 Docker 架构与设计、核心源码解读和高级实践技巧；第二部分归纳和比较了三类基于 Docker 的主流容器云项目，包括专注 Docker 容器编排与部署的容器云、专注应用支撑的容器云以及一切皆容器的 Kubernetes，进而详细解读了 Kubernetes 核心源码的设计与实现，最后介绍了几种典型场景下的 Kubernetes 最佳实践。

本书适用于有一定 Docker 基础的开发者、架构师、IT 专业学生以及探索基于 Docker 构建云计算平台的技术人员，也非常适合作为高校教材或培训资料。

◆ 著　　浙江大学SEL实验室
责任编辑　王军花
策划编辑　张　霞
责任印制　彭志环

◆ 人民邮电出版社出版发行　　北京市丰台区成寿寺路11号
邮编　100164　　电子邮件　315@ptpress.com.cn
网址　http://www.ptpress.com.cn
三河市海波印务有限公司印刷

◆ 开本：800×1000　1/16
印张：29.5
字数：697千字　　2016年 10 月第 2 版
印数：10 001 - 14 000册　　2016年 10 月河北第 1 次印刷

定价：89.00元

读者服务热线：(010)51095186转600　印装质量热线：(010)81055316
反盗版热线：(010)81055315
广告经营许可证：京东工商广字第 8052 号

站在巨人的肩上

Standing on Shoulders of Giants

iTuring.cn

站在巨人的肩上
Standing on Shoulders of Giants

iTuring.cn

推荐语

“虽然在此之前已经有了由 Docker 团队出的第一本 Docker 书，但是这是国内第一本深入解读 Docker 与 Kubernetes 原理的原创图书，这一点意义重大。本书比较完整地介绍了 Docker 与 Kubernetes 的工作原理和生态，非常有借鉴意义。”

——许式伟，七牛云存储 CEO

“Docker 容器技术已经在国内如火如荼地流行起来，浙江大学 SEL 实验室目前是国内掌握 Docker 技术最熟练的技术团队之一，他们在国内 Docker 技术界一直产生着重要影响。这次他们把 Docker 的实战经验汇编成书，可以帮助更多的 Docker 爱好者学习到一手的实战经验。”

——肖德时，数人科技 CTO

“本书非常细致地讲解了 Docker 技术的来龙去脉和技术细节，更为难得的是还加入了 Docker 生态当中的其他技术。Docker 这项技术本身就是将多种思想和技术融合的产物，从生态的视角去解读技术的来龙去脉将极大地促进读者对云计算和容器技术的重新思考。”

——程显峰，火币网 CTO

“本书宏观上描绘了容器和容器云技术发展的浪潮和生态系统，微观上以 Docker 和 Kubernetes 为典型进行了深度分析。无论是 Docker 技术爱好者，还是系统架构师、云端开发者、系统管理和运维人员，都能在本书中找到适合自己阅读的要点。浙江大学 SEL 实验室云计算团队是一支非常优秀的云计算研究团队，很多 85 后、90 后人才活跃在顶级社区前沿，感谢他们能将多年的知识和智慧积累分享出来！”

——刘俊，百度运维部高级架构师，百度最高奖获得者

“本书是浙江大学 SEL 实验室云计算团队多年深耕 Docker 及背后的容器技术的结晶。最大的特点就是深入，并且有各种实用案例和细致讲解。另外，这本书在怎样真正地把 Docker 及周边

产品落地以构建灵活多变的云平台方面也进行了生动的阐释。”

——郝林，微影时代架构师，《Go 并发编程实战》作者

“Docker 颠覆了容器技术，也将容器技术带到了新的高度。InfoQ 从 2014 年初就开始密切关注容器技术，见证并切身参与了容器技术的发展。作为我们的优秀作者，浙江大学 SEL 实验室在 InfoQ 撰写了很多与 Docker、Kubernetes 相关的技术文章，得到了广大读者的肯定。希望这本书能推动容器技术在中国的落地。”

——郭蕾，InfoQ 主编

“浙江大学 SEL 实验室属于国内较早接触并研究开源 PaaS 技术的团队之一，从传统 PaaS 的开源代表 CloudFoundry、OpenShift，到新一代基于 Docker 的 PaaS 平台如 DEIS、Flynn 等，他们均有深入的研究和实践经验。更为难得的是，他们不仅参与开源贡献，而且笔耕不辍，通过博客、论坛等方式积极分享有深度、有内涵的技术文章，并广泛参与国内 PaaS 届各种技术交流会议。华为 PaaS 团队也在与之交流中汲取了不少营养。此次，他们将近年来对 Docker 容器和 Kubernetes、DEIS、Flynn 等 PaaS 开源平台的研究成果结集成册，内容详尽且深入浅出。我相信，无论是入门者还是老手，都能够从中获益。”

——刘赫伟，华为中央软件院高级软件架构师

“容器技术在大型互联网企业中已广泛应用，而 Docker 是容器技术中的杰出代表。本书不仅介绍了 Docker 基础知识，而且进行了代码级的深入分析，并通过对 Kubernetes 等技术的讲解延伸至集群操作系统以及对 Docker 生态领域的思考，同时结合了大量实践，内容丰富，值得拥有。”

——王炜煜，百度运维部高级架构师，JPaaS 项目负责人

“Docker 作为操作系统层面轻量级的虚拟化技术，凭借简易的使用、快速的部署以及灵活敏捷的集成支持等优势，奠定了 Docker 如今在 PaaS 领域的江湖地位。浙江大学 SEL 实验室在云计算和 PaaS 领域耕耘多年，积累了丰富的经验。本书既有对 Docker 源代码层面的深度解读，也有实战经验的分享，希望这本书能够帮助 Docker 开发者在技术上更上一层楼。”

——李三红，蚂蚁金服基础技术部 JVM Architect

序

我已从事软件工程研究工作二十余年，在这期间，软件开发方式发生了巨大的变化。瞬息万变是这个时代的特征，固守经典、一成不变已无法应对，当代的软件工程拥有快速迭代的生命周期，越来越多的开发组织投入巨大精力关注软件开发的敏捷性。

云计算有了明确定义，浙江大学就组织研究力量投入这滚滚浪潮之中。云计算定义了一种按需索取、实时供应的特性，它是敏捷的。云平台提供的资源是计算能力，人们获得计算能力资源一如获取自来水和管道煤气一样方便。这为软件工程注入了新的活力，如果软件开发者可以快速、自由地获取开发过程中所需的各种资源，那么软件开发必将迎来一次飞跃式的发展。

然而，我们似乎并没有获得想要的飞跃。

众所周知，云计算拥有一个圣经般的三层模型，界限明确，职责分明。当下，依照圣经“戒律”，众多业内巨头率先建立起一批重型云平台，然而问题却慢慢浮现——按照传统定义设计的“云”对应用不够友好，要么做得不够，要么管得太死。

是时候打破“戒律”了吗？我认为是。

Docker 让所有人眼前一亮，它模糊了 IaaS 与 PaaS 之间的界限，为云计算的服务形式带来了无限的可能，Docker 带着它的容器理念破而后立，是云计算运动中一项了不起的创举。

丁轶群老师带领他的团队写作的这本《Docker——容器与容器云》，在很大程度上填补了国内容器与容器云技术领域深度分析的空白。本书浓缩了浙大 SEL 实验室多年来在 PaaS 以及容器技术领域的研究成果与开发实践经验，深入浅出地分析了云计算领域容器应用现状，是一部值得业内人士和容器技术爱好者长置案头的好书。

杨小虎

浙江大学软件学院院长

前　言

本书的写作目的不仅是在技术层面深入分析 Docker 背后的技术原理和设计思想，更在于从我们团队自 2011 年以来在云计算方面的积累出发，理清当前以 Docker、Kubernetes 为代表的“容器云”技术的发展脉络，以期对 IT 企业的开发运维人员、容器云服务提供商以及 Docker 技术爱好者在技术选型、技术路线规划上有所帮助。

2013 年是 Docker 正式开源发布的年份，也是我们团队开始使用 Docker 的时间。当时 Docker 作为一个单机版轻量级虚拟化工具，并没有像当前这样活跃的生态圈。我们使用 Docker 处理 Cloud Foundry 这类复杂分布式系统的快速部署和迁移，结果我们体验到了惊喜，但也有遗憾。确实，那时候 Docker 1.0 尚未发布，作为最先吃螃蟹的人之一，我们除了能感受到 Docker 相比虚拟机在资源利用率和性能上的巨大优势以及在使用方式上的高效便捷之外，还不得不忍受当时的 Docker 与一个完整的数据中心运维系统之间的差距。比如网络，跨宿主机间的通信在很长一段时间都困扰着我们；比如容器内部不能单独配置内核参数，一旦应用对性能有特殊要求的时候，就无法单独进行优化定制；再比如维护，时常需要手动清理僵尸容器、镜像等。

在随后的一整年里，我们真真切切地感受到了 Docker 是如何从一个开发运维人员略有耳闻的工具成长为一个技术圈里家喻户晓的名词。基于 Docker 的公有云、私有云项目也如雨后春笋般涌现；各大知名技术社区都为 Docker 开辟专栏，甚至出现了专为讨论 Docker 而生的技术社区。基于 Docker 的中国本土化也开始萌芽，各类国内镜像托管和加速服务层出不穷。Docker 官方也没有闲着，前不久，Docker 的各类邮件列表中都出现了招聘中国区执行官的消息。Docker 生态系统的建立已经是不争的事实，我们团队也从 Docker 的使用者，成为了 Docker、Kubernetes、libcontainer 等开源项目的特性维护者（maintainer）和代码贡献者（contributor）。

当前 Docker 已绝不仅仅是一项轻量级虚拟化技术，官方的 Docker 运维三件套、来自第三方的 Kubernetes、OpenShift v3、Flynn、Deis 等项目已经基于 Docker 这种容器技术构建出各种各样的容器云服务平台，关于 Docker 等容器技术的讨论重心也已经从“容器”转变为“容器云”。Docker 对于 IT 行业的价值也从节省资源这一方面扩展到对整个软件开发运维生命周期的改造。

作为软件行业多年的实践者和教育者，我们一直试图探索这样一些问题：云计算除了当前被广为接受的基础设施云平台（IaaS）的形态，是否还有更加贴近开发人员和运维人员的形态？云计算如何以更好的形态服务于互联网+这样一个以软件连接人与人、人与企业、企业与企业的时

代？正是 Docker 这类容器技术的出现，使得这样的探索成为了可能。

本书结构

本书共分两部分，沿着从容器到容器云的发展脉络，从“概念用法解析”到“核心原理分析”，然后到“高级实践技巧”，层层推进，全面介绍了 Docker 以及围绕 Docker 构建的各类容器云平台技术，深入分析了 Kubernetes 背后的技术原理和设计思想。

第一部分讲解了 Docker 容器的核心原理和实践技巧。其中第 1 章和第 2 章能够让读者在短时间内体验这场 IT 界的风暴，并且初步了解 Docker 的使用方法，为后续的源码解析做铺垫。第 3 章是本书第一部分的核心，这一章以 Docker 1.10 版本源码为基础，深入分析了容器的 namespace 和 cgroups 原理，紧接着我们以 `docker run` 命令为线索，一路贯穿 Docker 的容器创建、镜像组织、联合文件系统以及容器网络初始化的源码，深入透彻地向读者展示了从一条指令到最终 Linux 容器生成的整个过程中，Docker 源码的设计原理和执行路线。第 4 章则介绍了当前时髦的“容器化思维”以及 Docker 相关的几类实践技巧，包括网络、监控、服务发现等。值得一提的是，在上述代码走读的过程中，本书几乎没有贴出任何一部分 Docker 源码或者函数，而是力图使用平实的语言和生动的图示来展示代码背后的执行逻辑和设计思想。Docker 的源码字字珠玑，我们希望能够使用这样的解读方式使读者真正理解 Docker 和容器背后的设计方法和技术本质，而不是变成一本单纯的技术手册。

第二部分深入分析基于 Docker 的各类“容器云”平台的架构细节和背后的设计理念，这些容器云虽然在底层技术上都基于 Docker 这样的容器技术，但在背后的设计思想上却存在很大的差异。我们将看到一个因颠覆了原有 IaaS、PaaS 云计算生硬的分类方式而精彩纷呈的容器云世界。其中第 5 章介绍了一个最简单的容器云解决方案作为引子；第 6 章和第 7 章分析和比较了几类典型的容器云开源项目，包括了 Docker 官方的“三剑客”项目、Fleet 以及更类似经典 PaaS 的 Flynn 和 Deis；第 8 章是本书第二部分的重点，我们以 Kubernetes 1.2 版本源码为基础，从核心概念到架构梳理，再到深入到组件级别的 Kubernetes 源码解析，从多个维度详细讲解了 Kubernetes 容器云平台的各种技术细节，这在国内尚属首次。我们希望通过容器云平台的源码解读，能够带领读者从纷繁复杂的容器云项目中梳理出一个细致的脉络，让读者在选型和二次开发的过程中减少迷茫和试错成本。而作为 Kubernetes 项目的贡献者和特性维护者，我们希望有更多的技术人员能够从源码层面对 Kubernetes 有更深刻的理解和认识，并且同我们一起来推动这个优秀的开源项目在国内的进步和落地。在第二部分的结尾，我们试图回答之前的提问，即容器云应该以何种形态来更好地支撑当今时代。

第 2 版的改进

自本书第 1 版出版以来，容器生态圈已经发生了翻天覆地的变化。新的开源项目层出不穷，

各个开源项目都在快速迭代演进。Docker 已经从本书第 1 版里的 1.6.2 发展为当前的 1.10。Kubernetes 也从本书第 1 版里的 0.16 发展到了现在的 1.2，并且在 1.0.1 版本时宣布其已经正式进入可投入生产环境（production ready）的状态。

第 3 章是本书第一部分的重点。Docker 1.10 版相对于本书第 1 版中的 1.6.2 版，主要的更新包括如下几个方面。

- Docker 在架构方面不断将自身解耦，逐步发展成容器运行时（runtime）、镜像构建（builder）、镜像分发（distribution）、网络（networking）、数据卷（volume）等独立的功能组件，提供 daemon 来管理，并通过 Engine 暴露一组标准的 API 来操作这些组件（详见本书 3.2 节）。
- 将网络和数据卷提升为“一等公民”，提供了独立子命令进行操作，网络和数据卷具备独立的生命周期，不再依赖容器的生命周期（详见本书 3.7 节、3.8 节）。
- 网络实现方面，Docker 将网络相关的实现解耦为独立的组件 libnetwork，抽象出一个通用的容器网络模型（CNM），功能上也终于原生支持了跨主机通信（详见本书 3.8 节）。
- 在扩展性方面，在 1.7.0 版本后就开始支持网络、volume 和存储驱动（仍处于实验阶段）的插件化，开发者可以通过实现 Docker 提供的插件标准来定制自己的插件（详见本书 3.6 节、3.7 节、3.8 节）。
- 在 Docker 安全方面，Docker 支持了 user namespace 和 seccomp 来提高容器运行时的安全，在全新的镜像分发组件中引入可信赖的分发和基于内容存储的机制，从而提高镜像的安全性（详见本书 3.5 节、3.6 节、3.9 节）。

需要特别指出的一点是，随着容器如火如荼的发展，为了推动容器生态的健康发展，促进生态系统内各组织间的协同合作，容器的标准化也显得越来越重要。Linux 基金会于 2015 年 6 月成立 OCI（Open Container Initiative）组织，并针对容器格式和运行时制定了一个开放的工业化标准，即 OCI 标准。Docker 公司率先贡献出满足 OCI 标准的容器运行时 runC，HyperHQ 公司也开源了自己的 OCI 容器运行时 runV，相信业界会有越来越多的公司加入这个标准化浪潮中。Docker 公司虽然没有在 Docker 1.10 版本中直接使用 runC 作为容器的运行时，但是已经将“修改 Docker engine 来直接调用 runC 的二进制文件为 Docker 提供容器引擎”写入到了 1.10 版本的 roadmap 中。本书在 3.4.3 节中对 runC 的构建和使用进行了介绍。

第 8 章是本书第二部分的重点。由于 Kubernetes 的代码始终处于积极更新之中，自本书第 1 版截稿以来，Kubernetes 又相继发布了 0.17、0.18、0.19、0.20、0.21、1.0、1.1 与 1.2 等几个版本。主要的更新包括如下几个方面。

- 大大丰富了支撑的应用运行场景。从全面重构的 long-running service 的 replicaSet，到呼声渐高的支持 batch job 的 Job、可类比为守护进程的 DaemonSet、负责进行应用更新的 Deployment、具备自动扩展能力的 HPA（Horizontal Pod Autoscaler），乃至于有状态服务的 petSet，都已经或者即将涵盖在 Kubernetes 的支撑场景中（详见本书 8.2 节）。

- 加强各个组件的功能扩展或者性能调优。apiserver 和 controller manager 为应对全新的 resource 和 API 有显著的扩展；scheduler 也在丰富调度策略和多调度器协同调度上有积极的动作；kubelet 在性能上也有长足的进步，使得目前单个节点上支持的 pod 从原来的 30 个增长到了 110 个，集群工作节点的规模也从 100 个跃升为 1000 个；多为人诟病的 kube-proxy 如今也鸟枪换炮，默认升级为 iptables 模式，在吞吐量上也更为乐观；在可以预期的未来，rescheduler 将成为 Kubernetes 家庭中的新成员，使得重调度成为可能（详见本书 8.3 节）。
- 兼容更多的容器后端模型、网络及存储方案。从 Docker 到 rkt，Kubernetes 展示了对容器后端开放姿态，同时它还准备以 C/S 模式实现对其他容器的支撑。在网络方面，Kubernetes 引入了网络插件，其中最为瞩目的当属 CNI；存储上的解决方案更是层出不穷，flocker、Cinder、CephFS 不一而足，还增加了许多特殊用途的 volume，如 `secret`、`configmap` 等（详见本书 8.4 节、8.5 节）。
- 增加了 OpenID、Keystone 等认证机制、Webhook 等授权机制，以及更为丰富的多维资源管理机制 admission controller（详见本书 8.6 节）。
- 另外，作为 Kubernetes 社区的积极参与者，我们还专门增加了 8.8 节，讨论当前社区正在酝酿中的一些新特性，如 Ubernetes、petSet、rescheduler。我们还讨论了 Kubernetes 性能优化，以及 Kubernetes 与 OCI 的关系等话题。

除了全面更新这两个重点章节之外，我们还在第 1 章中更新了 Docker 近期的“大事记”并重新整理了容器生态圈，加入了许多重要的容器云技术开源项目，以及 OCI、CNCF 等国际标准化组织；在第 2 章中，我们将 Docker 命令行工具的基础用法更新到了 Docker 1.10 版；在第 4 章中完善了对时下火热的“容器化思维”和“微服务”的讨论；在第 6 章中更新了对 Docker“三剑客”——Compose、Swarm 和 Machine 的讨论；在附录中以 Docker 1.10 版为标准更新了附录 A 的 Docker 安装指南，以 Kubernetes 1.2 为标准，更新了附录 F 中 Kubernetes 的安装指南。

致谢

对于能够编写国内第一本在源代码层面深度解析 Docker 和 Kubernetes，并揭秘基于 Docker 容器的云计算生态圈底层技术的图书，我们感到非常荣幸。浙江大学 SEL 实验室云计算团队在此向所有支持帮助我们的朋友表达最诚挚的谢意，没有大家的支持，我们很可能无法顺利地完成这项工作。

感谢浙江大学软件学院杨小虎院长对云计算团队一直以来的关怀和支持，杨院长的远见卓识和诲人不倦令人钦佩。

感谢以极大热情参与到本书写作中的浙江大学计算机学院、软件学院的各位博士、硕士研究生：张磊、何思玫、高相林、张浩、孙健波、王哲、冯明振、乔刚、杜军、仇臣、周宇哲、叶瑞浩、赖春彬、孙宏亮、陈星宇。他们的热情是我们团队活力的源泉，他们使那些分散在各个领域

的技术得以整合。在本书编写过程中，他们不计个人得失地精诚合作，这是本书得以成书的基石。

特别要感谢不辞辛劳为本书出谋划策、日以继夜不断审阅修改的图灵公司的编辑们。在整个写作过程中，我们团队得到了出版方的大力支持。他们认真负责的态度是本书顺利出版的保证。

感谢 InfoQ 主编郭蕾一直以来对浙江大学 SEL 实验室技术分享工作所做出的支持和推广，他和 InfoQ 同事们的鼓励是推动本书发起的一大动力。

感谢《第一本 Docker 书》的译者刘斌为本书进行了细致的审读，并为我们提出了宝贵的修订建议。

感谢浙江大学 SEL 实验室云计算团队的其他所有人，他们认真负责的工作态度和令人满意的工作成果是本书不可或缺的支持力量。

感谢大家的共同努力，让我们的成果得以面世，在 Docker 布道之路上贡献出了自己的光和热，传播惠及当下的云计算前沿技术。

丁轶群

于浙江大学玉泉校区

站在巨人的肩上

Standing on Shoulders of Giants

iTuring.cn

站在巨人的肩上

Standing on Shoulders of Giants

iTuring.cn

目　录

第一部分　Docker深入解读