

第一本Docker书（修订版）

作者：[澳] 詹姆斯·特恩布尔（James Turnbull）

目 录

[版权信息](#)

[作者简介](#)

[译者简介](#)

[本书特色](#)

[版权声明](#)

[内容提要](#)

[对本书的赞誉](#)

[序](#)

[我们走在容器化的大道上](#)

[前言](#)

[本书面向的读者](#)

[致谢](#)

[技术审稿人团队](#)

[Scott Collier](#)

[John Ferlito](#)

[Paul Nasrat](#)

[技术插图作家](#)

[校对者](#)

[排版约定](#)

[代码及示例](#)

[说明](#)

[勘误](#)

[版本](#)

[第1章 简介](#)

[1.1 Docker简介](#)

[1.1.1 提供一个简单、轻量的建模方式](#)

[1.1.2 职责的逻辑分离](#)

[1.1.3 快速、高效的开发生命周期](#)

[1.1.4 鼓励使用面向服务的架构](#)

[1.2 Docker组件](#)

[1.2.1 Docker客户端和服务端](#)

[1.2.2 Docker镜像](#)

[1.2.3 Registry](#)

[1.2.4 容器](#)

[1.3 能用Docker做什么](#)

[1.4 Docker与配置管理](#)

[1.5 Docker的技术组件](#)

[1.6 本书的内容](#)

[1.7 Docker资源](#)

[第2章 安装Docker](#)

[2.1 安装Docker的先决条件](#)

[2.2 在Ubuntu和Debian中安装Docker](#)

[2.2.1 检查前提条件](#)

[2.2.2 安装Docker](#)

[2.2.3 Docker与UFW](#)

[2.3 在Red Hat和Red Hat系发行版中安装Docker](#)

[2.3.1 检查前提条件](#)

[2.3.2 安装Docker](#)

[2.3.3 在Red Hat系发行版中启动Docker守护进程](#)

[2.4 在OS X中安装Docker Toolbox](#)

[2.4.1 在OS X中安装Docker Toolbox](#)

[2.4.2 在OS X中启动Docker Toolbox](#)

[2.4.3 测试Docker Toolbox](#)

[2.5 在Windows中安装Docker Toolbox](#)

- [2.5.1 在Windows中安装Docker Toolbox](#)
- [2.5.2 在Windows中启动Docker Toolbox](#)
- [2.5.3 测试Docker Toolbox](#)
- [2.6 使用本书的Docker Toolbox示例](#)
- [2.7 Docker安装脚本](#)
- [2.8 二进制安装](#)
- [2.9 Docker守护进程](#)
 - [2.9.1 配置Docker守护进程](#)
 - [2.9.2 检查Docker守护进程是否正在运行](#)
- [2.10 升级Docker](#)
- [2.11 Docker用户界面](#)
- [2.12 小结](#)
- [第3章 Docker入门](#)
 - [3.1 确保Docker已经就绪](#)
 - [3.2 运行我们的第一个容器](#)
 - [3.3 使用第一个容器](#)
 - [3.4 容器命名](#)
 - [3.5 重新启动已经停止的容器](#)
 - [3.6 附着到容器上](#)
 - [3.7 创建守护式容器](#)
 - [3.8 容器内部都在干些什么](#)
 - [3.9 Docker日志驱动](#)
 - [3.10 查看容器内的进程](#)
 - [3.11 Docker统计信息](#)
 - [3.12 在容器内部运行进程](#)
 - [3.13 停止守护式容器](#)
 - [3.14 自动重启容器](#)
 - [3.15 深入容器](#)
 - [3.16 删除容器](#)
 - [3.17 小结](#)
- [第4章 使用Docker镜像和仓库](#)
 - [4.1 什么是Docker镜像](#)
 - [4.2 列出镜像](#)
 - [4.3 拉取镜像](#)
 - [4.4 查找镜像](#)
 - [4.5 构建镜像](#)
 - [4.5.1 创建Docker Hub账号](#)
 - [4.5.2 用Docker的commit命令创建镜像](#)
 - [4.5.3 用Dockerfile构建镜像](#)
 - [4.5.4 基于Dockerfile构建新镜像](#)
 - [4.5.5 指令失败时会怎样](#)
 - [4.5.6 Dockerfile和构建缓存](#)
 - [4.5.7 基于构建缓存的Dockerfile模板](#)
 - [4.5.8 查看新镜像](#)
 - [4.5.9 从新镜像启动容器](#)
 - [4.5.10 Dockerfile指令](#)
 - [4.6 将镜像推送到Docker Hub](#)
 - [自动构建](#)
 - [4.7 删除镜像](#)
 - [4.8 运行自己的Docker Registry](#)
 - [4.8.1 从容器运行Registry](#)
 - [4.8.2 测试新Registry](#)
 - [4.9 其他可选Registry服务](#)
 - [Quay](#)
 - [4.10 小结](#)
- [第5章 在测试中使用Docker](#)
 - [5.1 使用Docker测试静态网站](#)
 - [5.1.1 Sample网站的初始Dockerfile](#)
 - [5.1.2 构建Sample网站和Nginx镜像](#)
 - [5.1.3 从Sample网站和Nginx镜像构建容器](#)

- [5.1.4 修改网站](#)
- [5.2 使用Docker构建并测试Web应用程序](#)
 - [5.2.1 构建Sinatra应用程序](#)
 - [5.2.2 创建Sinatra容器](#)
 - [5.2.3 扩展Sinatra应用程序来使用Redis](#)
 - [5.2.4 将Sinatra应用程序连接到Redis容器](#)
 - [5.2.5 Docker内部连网](#)
 - [5.2.6 Docker Networking](#)
 - [5.2.7 使用容器连接来通信](#)
 - [5.2.8 连接容器小结](#)
- [5.3 Docker用于持续集成](#)
 - [5.3.1 构建Jenkins和Docker服务器](#)
 - [5.3.2 创建新的Jenkins作业](#)
 - [5.3.3 运行Jenkins作业](#)
 - [5.3.4 与Jenkins作业有关的下一步](#)
 - [5.3.5 Jenkins设置小结](#)
- [5.4 多配置的Jenkins](#)
 - [5.4.1 创建多配置作业](#)
 - [5.4.2 测试多配置作业](#)
 - [5.4.3 Jenkins多配置作业小结](#)
- [5.5 其他选择](#)
 - [5.5.1 Drone](#)
 - [5.5.2 Shippable](#)
- [5.6 小结](#)
- [第6章 使用Docker构建服务](#)
 - [6.1 构建第一个应用](#)
 - [6.1.1 Jekyll基础镜像](#)
 - [6.1.2 构建Jekyll基础镜像](#)
 - [6.1.3 Apache镜像](#)
 - [6.1.4 构建Jekyll Apache镜像](#)
 - [6.1.5 启动Jekyll网站](#)
 - [6.1.6 更新Jekyll网站](#)
 - [6.1.7 备份Jekyll卷](#)
 - [6.1.8 扩展Jekyll示例网站](#)
 - [6.2 使用Docker构建一个Java应用服务](#)
 - [6.2.1 WAR文件的获取程序](#)
 - [6.2.2 获取WAR文件](#)
 - [6.2.3 Tomcat7应用服务器](#)
 - [6.2.4 运行WAR文件](#)
 - [6.2.5 基于Tomcat应用服务器的构建服务](#)
 - [6.3 多容器的应用栈](#)
 - [6.3.1 Node.js镜像](#)
 - [6.3.2 Redis基础镜像](#)
 - [6.3.3 Redis主镜像](#)
 - [6.3.4 Redis副本镜像](#)
 - [6.3.5 创建Redis后端集群](#)
 - [6.3.6 创建Node容器](#)
 - [6.3.7 捕获应用日志](#)
 - [6.3.8 Node程序栈的小结](#)
 - [6.4 不使用SSH管理Docker容器](#)
 - [6.5 小结](#)
- [第7章 Docker编配和服务发现](#)
 - [7.1 Docker Compose](#)
 - [7.1.1 安装Docker Compose](#)
 - [7.1.2 获取示例应用](#)
 - [7.1.3 docker-compose.yml文件](#)
 - [7.1.4 运行Compose](#)
 - [7.1.5 使用Compose](#)
 - [7.1.6 Compose小结](#)
 - [7.2 Consul、服务发现和Docker](#)

- [7.2.1 构建Consul镜像](#)
- [7.2.2 在本地测试Consul容器](#)
- [7.2.3 使用Docker运行Consul集群](#)
- [7.2.4 启动具有自启动功能的Consul节点](#)
- [7.2.5 启动其余节点](#)
- [7.2.6 配合Consul，在Docker里运行一个分布式服务](#)
- [7.3 Docker Swarm](#)
 - [7.3.1 安装Swarm](#)
 - [7.3.2 创建Swarm集群](#)
 - [7.3.3 创建容器](#)
 - [7.3.4 过滤器](#)
 - [7.3.5 策略](#)
 - [7.3.6 小结](#)
- [7.4 其他编排工具和组件](#)
 - [7.4.1 Fleet和etcd](#)
 - [7.4.2 Kubernetes](#)
 - [7.4.3 Apache Mesos](#)
 - [7.4.4 Helios](#)
 - [7.4.5 Centurion](#)
- [7.5 小结](#)
- [第8章 使用Docker API](#)
 - [8.1 Docker API](#)
 - [8.2 初识Remote API](#)
 - [8.3 测试Docker Remote API](#)
 - [8.3.1 通过API来管理Docker镜像](#)
 - [8.3.2 通过API管理Docker容器](#)
 - [8.4 改进TProv应用](#)
 - [8.5 对Docker Remote API进行认证](#)
 - [8.5.1 建立证书授权中心](#)
 - [8.5.2 创建服务器的证书签名请求和密钥](#)
 - [8.5.3 配置Docker守护进程](#)
 - [8.5.4 创建客户端证书和密钥](#)
 - [8.5.5 配置Docker客户端开启认证功能](#)
 - [8.6 小结](#)
- [第9章 获得帮助和对Docker进行改进](#)
 - [9.1 获得帮助](#)
 - [9.1.1 Docker用户、开发邮件列表及论坛](#)
 - [9.1.2 IRC上的Docker](#)
 - [9.1.3 GitHub上的Docker](#)
 - [9.2 报告Docker的问题](#)
 - [9.3 搭建构建环境](#)
 - [9.3.1 安装Docker](#)
 - [9.3.2 安装源代码和构建工具](#)
 - [9.3.3 检出源代码](#)
 - [9.3.4 贡献文档](#)
 - [9.3.5 构建开发环境](#)
 - [9.3.6 运行测试](#)
 - [9.3.7 在开发环境中使用Docker](#)
 - [9.3.8 发起pull request](#)
 - [9.3.9 批准合并和维护者](#)
 - [9.4 小结](#)
- [看完了](#)

本书由布克书屋整理，www.bukebook.cn 提供最新最全的优质电子书下载！！

版权信息

书名：第一本Docker书（修订版）

ISBN：978-7-115-41933-0

本书由人民邮电出版社发行数字版。版权所有，侵权必究。

您购买的人民邮电出版社电子书仅供您个人使用，未经授权，不得以任何方式复制和传播本书内容。

我们愿意相信读者具有这样的良知和觉悟，与我们共同保护知识产权。

如果购买者有侵权行为，我们可能对该用户实施包括但不限于关闭该帐号等维权措施，并可能追究法律责任。

• 著 [澳] James Turnbull

译 李兆海 刘 斌 巨 震

责任编辑 杨海玲

• 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>

• 读者服务热线：(010)81055410

反盗版热线：(010)81055315

作者简介

James Turnbull是一位技术作家，还是一名开源极客。他最近的大作是一本讲述流行开源日志工具的书——*The LogStash Book*。James还写了两本关于Puppet的书，一本是*Pro Puppet*，另一本是较早的*Pulling Strings with Puppet: Configuration Management Made Easy*。此外，James还写了*Pro Linux System Administration*、*Pro Nagios 2.0*和*Hardening Linux*这三本书。

James是Kickstarter公司的CTO。之前，James曾担任Docker公司服务与支持副总裁、Venmo公司工程副总裁和Puppet Labs的技术运维副总裁。

本修订版聚集于Docker 1.9及更高版本。

译者简介

李兆海 网名Googol Lee。使用Googol这个名字真的是因为“10的100次方”这个意思，和后来的Google公司没有一点儿关系。多年后端程序员，早期以C、C++为主，后来转向Python，现在以Go为生。曾写过《Golang初探》发表于2011年2月号《程序员》。Docker早期使用者。平时喜欢乱翻书，遇到感兴趣的都会研究一番。Twitter账户@googollee。

刘斌 具有10余年软件开发经验，关注后台开发技术和各种编程语言。做过电子商务、金融、企业系统以及Android手机开发；写过Delphi，也兼做系统管理员和DBA（现在都改叫DevOps了）；既做后台应用，也要调用前台CSS和JavaScript，可还是不敢自称“全栈”（Full Stack）；今又舶来“增长黑客”（Growth Hacker），我想我要做一个“增长工程师”（Growth Engineer）。个人主页<http://liubin.org>。

巨震 北京大学软件工程硕士，服务器端开发者。目前就职于创业公司，使用Node.js、Golang进行服务器端开发。2013年底开始研究Docker，是Docker中文社区的活跃贡献者，负责Docker技术文章和视频的翻译、校对工作。生活中喜欢美食、骑行，热衷于PC硬件，喜爱折腾，热爱一切计算机相关的技术，坚信技术改变世界。最崇拜的技术传奇人物是前id Software首席程序员、现Oculus VR首席技术官John Carmack。

本书特色

- 全球第一本Docker技术图书中文版。
- Docker核心团队成员权威著作。
- 非常适合作为学习Docker的第一本入门书。
- 本书基于Docker 1.9及以上版本。

版权声明

Simplified Chinese translation copyright © 2015

by Posts and Telecommunications Press

Published by arrangement with James Turnbull

欢迎访问：电子书学习和下载网站 (<https://www.shgis.cn>)

文档名称：《第一本Docker书（修订版）》[澳] 詹姆斯·特恩布尔 (James Turnbull) 著.epu

请登录 <https://shgis.cn/post/320.html> 下载完整文档。

手机端请扫码查看：

