

85 | 程序员练级攻略：容器化和自动化运维

2018-07-24 陈皓

左耳听风

进入课程 >



讲述：柴巍

时长 06:42 大小 3.07M



这篇文章我们来重点学习 Docker 和 Kubernetes，它们已经是分布式架构和自动化运维的必备工具了。对于这两个东西，你千万不要害怕，因为技术方面都不算复杂，只是它们的玩法和传统运维不一样，所以你不用担心，只要你花上一点时间，一定可以学好的。

Docker

你可以先看一下 Docker 的官方介绍 [Docker Overview](#)。

然后再去一个 Web 在线的 Playground 上体验一下，[Katacoda Docker Playground](#) 或者是 [Play With Docker](#)。

接下来，跟着 [Learn Docker](#) 这个文档中的教程自己安装一个 Docker 的环境，实操一把。

然后跟着 [Docker Curriculum](#) 这个超详细的教程玩一下 Docker。

有了上述的一些感性体会之后，你就可以阅读 Docker 官方文档 [Docker Documentation](#) 了，这是学习 Docker 最好的方式。

如果你想了解一下 Docker 的底层技术细节，你可以参看我的文章。

[Docker 基础技术：Linux Namespace \(上\)](#)

[Docker 基础技术：Linux Namespace \(下\)](#)

[Docker 基础技术：Cgroup](#)

[Docker 基础技术：AUFS](#)

[Docker 基础技术：DeviceMapper](#)

还有一些不错的与 Docker 网络有关的文章你需要阅读及实践一下。

[A container networking overview](#)

[Docker networking 101 - User defined networks](#)

[Understanding CNI \(Container Networking Interface\)](#)

[Using CNI with Docker](#)

Docker 有下面几种网络解决方案：[Calico](#)、[Flannel](#) 和 [Weave](#)，你需要学习一下。另外，还需要学习一下 [netshoot](#)，这是一个很不错的用来诊断 Docker 网络问题的工具集。

关于这几个容器网络解决方案的性能对比，你可以看一下下面这几篇文章或报告。

[Battlefield: Calico, Flannel, Weave and Docker Overlay Network](#)

[Comparison of Networking Solutions for Kubernetes](#)

[Docker Overlay Networks: Performance analysis in high-latency environments](#)

如果你对 Docker 的性能有什么问题的话，你可以看一下下面这些文章。

[IBM Research Report: An Updated Performance Comparison of Virtual Machines and Linux Containers](#)

[An Introduction to Docker and Analysis of its Performance](#)

下面是一些和存储相关的文章。

[Storage Concepts in Docker: Network and Cloud Storage](#)

[Storage Concepts in Docker: Persistent Storage](#)

[Storage Concepts in Docker: Shared Storage and the VOLUME directive](#)

然后是跟运维相关的文章。

[Docker Monitoring with the ELK Stack: A Step-by-Step Guide](#)

最后，推荐看看 [Valuable Docker Links](#)，其中收集并罗列了一系列非常不错的 Docker 文章。

最佳实践

下面分享一些与 Docker 相关的最佳实践。

[Best Practices for Dockerfile](#)，Docker 官方文档里的 Dockerfile 的最佳实践。

[Docker Best Practices](#)，这里收集汇总了存在于各个地方的使用 Docker 的建议和实践。

[Container Best Practices](#)，来自 Atomic 项目，是一个介绍容器化应用程序的架构、创建和管理的协作型文档项目。

[Eight Docker Development Patterns](#)，八个 Docker 的开发模式：共享基础容器、共享同一个卷的多个开发容器、开发工具专用容器、测试环境容器、编译构建容器、防手误的安装容器、默认服务容器、胶黏容器。

Kubernetes

Kubernetes 是 Google 开源的容器集群管理系统，是 Google 多年大规模容器管理技术 Borg 的开源版本，也是 CNCF 最重要的项目之一，主要功能包括：

基于容器的应用部署、维护和滚动升级；

负载均衡和服务发现；

跨机器和跨地区的集群调度；

自动伸缩；

无状态服务和有状态服务；

广泛的 Volume 支持；

插件机制保证扩展性。

Kubernetes 发展非常迅速，已经成为容器编排领域的领导者。

首先，我推荐你阅读 Kubernetes 前世今生的一篇论文。

[Borg, Omega, and Kubernetes](#)，看看 Google 这十几年来从这三个容器管理系统中得到的经验教训。

学习 Kubernetes，有两个免费的开源电子书。

《[Kubernetes Handbook](#)》，这本书记录了作者从零开始学习和使用 Kubernetes 的心路历程，着重于经验分享和总结，同时也会有相关的概念解析。希望能够帮助你少踩坑，少走弯路，还会指引你关注 kubernetes 生态周边，如微服务构建、DevOps、大数据应用、Service Mesh、Cloud Native 等领域。

《[Kubernetes 指南](#)》，这本书旨在整理平时在开发和使用 Kubernetes 时的参考指南和实践总结，形成一个系统化的参考指南以方便查阅。

这两本电子书都不错，前者更像是一本学习教程，而且面明显广一些，还包括 Cloud Natvie、Service Mesh 以及微服务相关的东西。而后者聚焦于 Kubernetes 本身，更像一本参考书。

另外，我这两天也读完了《Kubernetes in Action》一书，感觉写的非常好，一本很完美的教科书，抽丝剥茧，图文并茂。如果你只想读一本有关 Kubernetes 的书来学习 Kubernetes，那么我推荐你就选这本。

但是也别忘了 Kubernetes 的官方网站：[Kubernetes.io](https://kubernetes.io)，上面不但有[全面的文档](#)，也包括一个很不错的[官方教程](#)。

此外，还有一些交互式教程，帮助你理解掌握，以及一些很不错的文章推荐你阅读。

一些交互式教程

[Katacoda](#)

[Kubernetes Bootcamp](#)

一些文章

这里还有一些不错的文档，你应该去读一下。

[Kubernetes tips & tricks](#)

[Achieving CI/CD with Kubernetes](#)

[How to Set Up Scalable Jenkins on Top of a Kubernetes Cluster](#)

10 Most Common Reasons Kubernetes Deployments Fail [Part I](#) 和 [Part II](#)

[How to Monitor Kubernetes](#)，一共有 4 个篇章

[Logging in Kubernetes with Fluentd and Elasticsearch](#)

[Kubernetes Monitoring: Best Practices, Methods, and Existing Solutions](#)

网络相关的文章

要学习 Kubernetes，你只需要读一下，下面这个 Kubernetes 101 系列的文章。

[Kubernetes 101 - Networking](#)

[Kubernetes networking 101 - Pods](#)

[Kubernetes networking 101 - Services](#)

[Kubernetes networking 101 - \(Basic\) External access into the cluster](#)

[Kubernetes Networking 101 - Ingress resources](#)

[Getting started with Calico on Kubernetes](#)

CI/CD 相关的文章

[Automated Image Builds with Jenkins, Packer, and Kubernetes](#)

[Jenkins setups for Kubernetes and Docker Workflow](#)

[Lab: Build a Continuous Deployment Pipeline with Jenkins and Kubernetes](#)

最佳实践

[Kubernetes Best Practices](#) by [Sachin Arote](#) , AWS 工程师总结的最佳实践。

[Kubernetes Best Practices](#) by [Sandeep Dinesh](#) , Google 云平台工程师总结的最佳实践。

Docker 和 Kubernetes 资源汇总

下面是 GitHub 上和 Docker & Kubernetes 相关的 Awesome 系列。

[Awesome Docker](#)。

[Awesome Kubernetes](#)。

虽然上面的这些系列非常全的罗列了很多资源，但是我觉得很不系统。对于系统的说明 Docker 和 Kubernetes 生态圈，我非常推荐大家看一下 The New Stack 为 Kubernetes 出的一系列的电子书或报告。

[The New Stack eBook Series](#) , 非常完整和详实的 Docker 和 Kubernetes 生态圈的所有东西。

Book 01: [The Docker Container Ecosystem](#)

Book 02: [Applications & Microservices with Docker & Containers](#)

Book 03: [Automation & Orchestration with Docker & Containers](#)

Book 04: [Network, Security & Storage with Docker & Containers](#)

Book 05: [Monitoring & Management with Docker & Containers](#)

Book 06: [Use Cases for Kubernetes](#)

Book 07: [State of the Kubernetes Ecosystem](#)

Book 08: [Kubernetes Deployment & Security Patterns](#)

Book 09: [CI/CD with Kubernetes](#)

Book 10: [Kubernetes solutions Directory](#)

Book 11: [Guid to Cloud-Native Microservices](#)

小结

总结一下今天的内容。Docker 和 Kubernetes 已经成为分布式架构和自动化运维方面的不可或缺的两大基本构成，是你必需要学习的。虽然它们的玩法跟传统运维不一样，但技术方面并不算复杂，只要你花上一点时间，一定会学好的。

在这篇文章中，我推荐了 Docker 和 Kubernetes 基础技术方面的学习资料，并给出了存储、运维、网络、CI/CD 等多方面的资料，同时列出了与之相关的最佳实践。相信认真学习和消化这些知识，你一定可以掌握 Docker 和 Kubernetes 两大利器。

下篇文章，我们将学习机器学习和人工智能方面的内容。敬请期待。

下面是《程序员练级攻略》系列文章的目录。

[开篇词](#)

入门篇

[零基础启蒙](#)

[正式入门](#)

修养篇

[程序员修养](#)

专业基础篇

[编程语言](#)

[理论学科](#)

[系统知识](#)

软件设计篇

软件设计

高手成长篇

Linux 系统、内存和网络（系统底层知识）

异步 I/O 模型和 Lock-Free 编程（系统底层知识）

Java 底层知识

数据库

分布式架构入门（分布式架构）

分布式架构经典图书和论文（分布式架构）

分布式架构工程设计(分布式架构)

微服务

容器化和自动化运维

机器学习和人工智能

前端基础和底层原理（前端方向）

前端性能优化和框架（前端方向）

UI/UX 设计（前端方向）

技术资源集散地

左耳朵耗子

全年独家专栏《左耳听风》

20000 名程序员的练级攻略

陈皓 资深技术专家
骨灰级程序员



新版升级：点击「👤 请朋友读」，10位好友免费读，邀请订阅更有**现金**奖励。

© 版权归极客邦科技所有，未经许可不得传播售卖。页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。

上一篇 84 | 程序员练级攻略：微服务

下一篇 86 | 程序员练级攻略：机器学习和人工智能

精选留言 (27)

写留言



微leng

2018-07-24

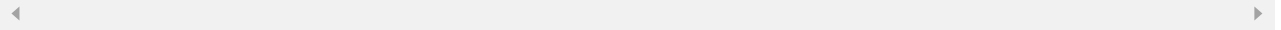
👍 29

耗子叔 这几期感觉像是在贴链接，能不能找一两个重点细细解读下，还是想听听你的东西。

作者回复: 我是想授人以渔，我给的这些文章，讲的比我讲的好。如果你想听我讲，我无非也就是把这些东西消化一遍然后接合自己的经历讲述一下。并于重点，我还是那句话，我已经过滤了七成以上的资料了，留下的已经是重点了。我这样做就是想培养一下学习者的自己的学习和总结能力。

这里真的没有速成，不要觉得通过我，你就可以不用思考，我是不会把饭喂到你嘴里的，我更希望你能自己总结知识，自己找人讨论，自己找到重点。

我这里只是给大家画了一张地图。



啦啦啦将比

2018-07-29

👍 11

这么多抱怨的，你知道我自学，为找这些东西费了多大得劲吗



刘強

2018-07-24

👍 10

在二叉树视频里，看到耗子的一头白发，顿生感动和佩服。没有人能随随便便成功，耗子的几百行代码值20万，背后付出了多少。

展开 ▾



Joran

2018-07-24

👍 6

耗子叔你的知识广度和深度，我十分敬佩，但是这么多我感觉够我学一辈子了



打奥特曼的...

2018-08-01

👍 3

最近在学习docker k8s 很需要指导方案。耗子叔的专栏就是巨大宝藏。网络上自己搜索的文档良莠不齐！



manx00

2018-07-24

👍 3

看到目录最后的三个点，感觉路还长...

展开 ▾



别人家的孩...

2018-07-25

👍 2

谢谢，正需要。

展开 ▾



机械师种太...

2018-07-24

👍 2

非常期待人下一期人工智能~~

展开 ▾



左耳朵

2019-03-20

👍 1

近期读完了《Kubernetes in Action》一书，5星好评。如果你只想读一本有关K8s的书来学习K8s，那么我推荐你就选这本。

展开 ▾



小金牛

2018-08-10

👍 1

感谢耗子叔的归纳，非常珍贵的资料

展开 ▾



MarksGui

2018-08-02

👍 1

学习编程没有捷径，这些资料自己要找全很不容易。路已经铺好了，还要人推着你走么？



章洁

2018-07-29

👍 1

一、世界上真没有太多的捷径，有些好的经验方法，但仍需要自己消化。

二、除了望洋兴叹外，不如跨步向前，每一步都让你与目标更近。

这是除了技术本身之外读到的两点。

展开 ▾



NonStatic

2018-07-24

👍 1

推荐新入坑的也顺手了解一下Moby，开源版的Docker。两者大部分地方都是同一套代码，不过Moby能够帮你绕过产品中的法律问题。

展开 ▾



小薛薛

2018-07-24

👍 1

做个标记，看几年后能不能走到这里。

展开 ▾



Abyssal

2018-07-24

👍 1

感谢皓叔！

展开 ▾



CN8818

2019-04-26

👍

唉，英语是一大障碍

展开 ▾



阿邱

2019-04-17

👍

现在的我们不缺资料，我们要的是一些挑选的东西，东西少点都没关系，不要坏了自己的名声，最起码得讲讲k8s与docker的区别吧！

展开 ▾



godtrue

2019-01-11

👍

阅后留痕

高山仰止，深刻体会到了英语是阶级的门槛，加油跨进去。

展开 ▾



简言

2018-12-10

👍

给出方向就够了，路还是要自己走的。

展开 ▾



老K

👍

2018-08-23

棒极了，开头的5点作为初学者的入门切入点，太合适了！贴链接比长篇大论的讲好多了（可以融汇自己的思考+每篇文章作者的思想为一体）！感谢！！

展开 ∨