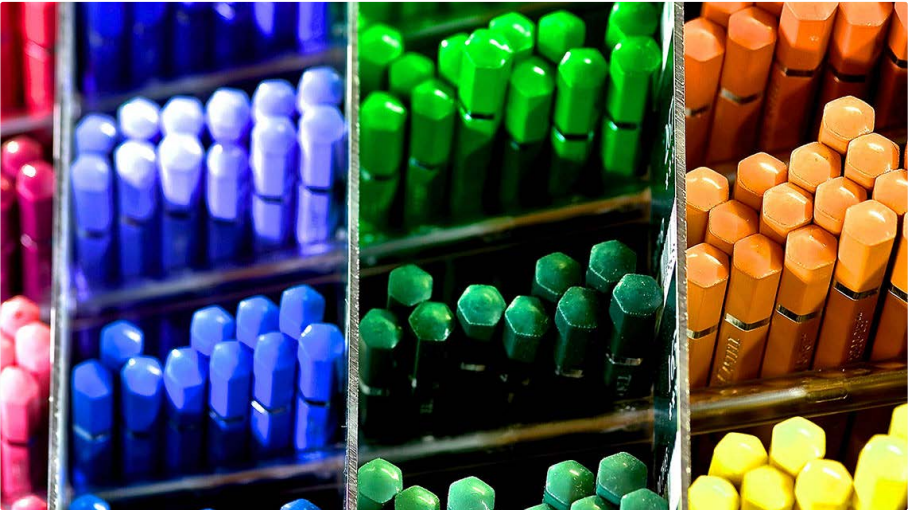


程序员练级攻略（2018）：容器化和自动化运维

2018-07-24 陈皓





程序员练级攻略（2018）：容器化和自动化运维

陈皓

- 00:00 / 06:42

这篇文章重点学习 Docker 和 Kubernetes，它们已经是分布式架构和自动化运维的必需品了，也是你必需要学习的。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，所以不用担心，只要你花上一点时间，一定会学好的。

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。

但是也别忘了Kubernetes的官方网站：[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	
<ul style="list-style-type: none">• 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。	
<ul style="list-style-type: none">• 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出的一系列电子书或报告。	
<ul style="list-style-type: none">• The New Stack eBook Series , 非常完整和详实的 Docker 和 Kubernetes 生态圈的所有东西。<ul style="list-style-type: none">◦ 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，是你所需要学习的。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（2018）》系列文章的目录（持续更新中）。	
<ul style="list-style-type: none">• 开篇词• 入门篇<ul style="list-style-type: none">◦ 零基础启蒙◦ 正式入门• 修养篇<ul style="list-style-type: none">◦ 程序员修养• 专业基础篇<ul style="list-style-type: none">◦ 编程语言◦ 理论学科◦ 系统知识• 软件设计篇<ul style="list-style-type: none">◦ 软件设计• 高手成长篇<ul style="list-style-type: none">◦ 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◦ 分布式架构工程设计◦ 容器化和自动化运维◦	



dilei	2018-07-24
<p>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。</p> <p>作者回复</p> <p>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。</p> <p>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。</p> <p>我这里只是给大家画了一张地图。</p>	
Joran	2018-07-24
<p>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</p> <p>刘强</p> <p>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。</p>	
manx00	2018-07-24
<p>看到目录最后的三个点，感觉路还长...</p> <p>小薛薛</p> <p>做个标记，看几年后能不能走到这里。</p>	
Abyssal	2018-07-24
<p>感谢皓叔！</p> <p>机械师种太阳</p> <p>非常期待人下一期人工智能~~</p>	
KingPoker	2018-07-26
<p>公司在推k8s，小组上了几个小项目，确实好用</p> <p>别人家的孩子</p> <p>谢谢，正需要。</p>	
NonStatic	2018-07-25
<p>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。</p> <p>JaneLee</p> <p>都是链接，还不如拿出一个点做一下儿深入的讲解，这么多链接怎么看？</p>	
李志博	2018-07-24
<p>非常喜欢，太感谢分享这些宝贵的资源</p> <p>_CountingStars</p> <p>kubernetes in action 不错</p>	

