

0 | 开篇词 | 打通“容器技术”的任督二脉

开篇词 | 打通“容器技术”的任督二脉

张磊 2018-08-27



□

08:45

讲述：张磊 大小：4.02M

你好，我是张磊，Kubernetes 社区的一位资深成员和项目维护者。

2012 年，我还在浙大读书的时候，就有幸组建了一个云计算与 PaaS 基础设施相关的科研团队，就这样，我从早期的 Cloud Foundry 社区开始，正式与容器结缘。

这几年来，我大多数时间都在 Kubernetes 项目里从事上游技术工作，也得以作为一名从业者和社区成员的身份，参与和亲历了容器技术从“初出茅庐”到“尘埃落定”的全过程。

而即使从 2013 年 Docker 项目发布开始算起，这次变革也不过短短 5 年时间，可在现如今的技术圈儿里，不懂容器，没听过 Kubernetes，你还真不好意思跟人打招呼。

容器技术这样一个新生事物，完全重塑了整个云计算市场的形态。它不仅催生出了一批年轻有为的容器技术人，更培育出了一个具有相当规模的开源基础设施技术市场。

在这个市场里，不仅有 Google、Microsoft 等技术巨擘们厮杀至今，更有无数的国内外创业公司前仆后继。而在国内，甚至连以前对开源基础设施领域涉足不多的 BAT、蚂蚁、滴滴这样的巨头们，也都从 AI、云计算、微服务、基础设施等维度多管齐下，争相把容器和 Kubernetes 项目树立为战略重心之一。

就在这场因“容器”而起的技术变革中，Kubernetes 项目已然成为容器技术的事实标准，重新定义了基础设施领域对应用编排与管理的种种可能。

2014 年后，我开始以远程的方式，全职在 Kubernetes 和 Kata Containers 社区从事上游开发工作，先后发起了容器镜像亲密性调度、基于等价类的调度优化等多个核心特性，参与了容器运行时接口、安全容器沙盒等多个基础特性的设计和研发。还有幸作为主要的研发人员和维护者之一，亲历了 Serverless Container 概念的诞生与崛起。

在 2015 年，我发起和组织撰写了《Docker 容器与容器云》一书，希望帮助更多的人利用容器解决实际场景中的问题。时至今日，这本书的第 2 版也已经出版快 2 年了，受到了广大容器技术读者们的好评。

2018 年，我又赴西雅图，在微软研究院（MSR）云计算与存储研究组，专门从事基于 Kubernetes 的深度学习基础设施相关的研究工作。

我与容器打交道的这些年，一直在与关注容器生态的工程师们交流，并经常探讨容器在落地过程中遇到的问题。从这些交流中，我发现总有很多相似的问题被反复提及，比如：

1. 为什么容器里只能跑“一个进程”？
2. 为什么我原先一直在用的某个 JVM 参数，在容器里就不好使了？
3. 为什么 Kubernetes 就不能固定 IP 地址？容器网络连不通又该如何去 Debug？
4. Kubernetes 中 StatefulSet 和 Operator 到底什么区别？PV 和 PVC 这些概念又该怎么用？

这些问题乍一看与我们平常的认知非常矛盾，但它们的答案和原理却并不复杂。不过很遗憾，对于刚刚开始学习容器的技术人员来说，它们却很难用一两句话就能解释清楚。

究其原因在于，从过去以物理机和虚拟机为主体的开发运维环境，向以容器为核心的基础设施的转变过程，并不是一次温和的改革，而是涵盖了对网络、存储、调度、操作系统、分布式原理等各个方面的容器化理解和改造。

这就导致了很多初学者，对于容器技术栈表现出来的这些难题，要么知识储备不足，要么杂乱无章、无法形成体系。这，也是很多初次参与 PaaS 项目的从业者们共同面临的一个困境。

其实，容器技术体系看似纷乱繁杂，却存在着很多可以“牵一发而动全身”的主线。比如，Linux 的进程模型对于容器本身的重要意义；或者，“控制器”模式对整个 Kubernetes 项目提纲挈领的作用。

但是，这些关于 Linux 内核、分布式系统、网络、存储等方方面面的积累，并不会在 Docker 或者 Kubernetes 的文档中交代清楚。可偏偏就是它们，才是真正掌握容器技术体系的精髓所在，是每一位技术从业者需要悉心修炼的“内功”。

而这，也正是我开设这个专栏的初衷。

我希望借由这个专栏，给你讲清楚容器背后的这些技术本质与设计思想，并结合着对核心特性的剖析与实践，加深你对容器技术的理解。为此，我把专栏划分成了 4 大模块：

1. **“白话”容器技术基础：**我希望用饶有趣味的解说，给你梳理容器技术生态的发展脉络，用最通俗易懂的语言描述容器底层技术的实现方式，让你知其然，也知其所以然。
2. **Kubernetes 集群的搭建与实践：**Kubernetes 集群号称“非常复杂”，但是如果明白了其中的架构和原理，选择了正确的工具和方法，它的搭建却也可以“一键安装”，它的应用部署也可以浅显易懂。
3. **容器编排与 Kubernetes 核心特性剖析：**这是这个专栏最重要的内容。“编排”永远都是容器云项目的灵魂所在，也是 Kubernetes 社区持久生命力的源泉。在这一模块，我会从分布式系统设计的视角出发，抽象和归纳出这些特性中体现出来的普遍方法，然后带着这些指导思想去逐一阐述 Kubernetes 项目关于编排、调度和作业管理的各项核心特性。“不识庐山真面目，只缘身在此山中”，希望这样一个与众不同的角度，能够给你以全新的启发。
4. **Kubernetes 开源社区与生态：**“开源生态”永远都是容器技术和 Kubernetes 项目成功的关键。在这个模块，我会和你一起探讨，容器社区在开源软件工程指导下的演进之路；带你思考，如何同团队一起平衡内外部需求，让自己逐渐成为社区中不可或缺的一员。

我希望通过这些对容器与 Kubernetes 项目的逐层剖析，能够让你面对容器化浪潮时不再踌躇无措，有一种拨云见日的酣畅淋漓。

最后，我想再和你分享一个故事。

2015 年我在 InfoQ 举办的第一届容器技术大会上，结识了当时 CoreOS 的布道师 Kelsey Hightower，他热情地和大家一起安装和体验微信，谈笑风生间，还时不时地安利一番自家产品。

但两年后也就是 2017 年，Kelsey 已经是全世界容器圈儿的意见领袖，是 Google 公司 Kubernetes 项目的首席布道师，而他的座右铭也变为了“只布道，不推销”。此时，就算你漂洋过海想要亲自拜会 Kelsey，恐怕也得先预约下时间了。

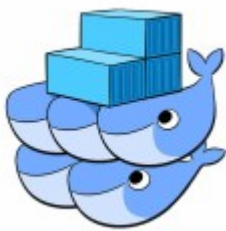
诚然，Kelsey 的“一夜成名”，与他的勤奋和天赋密不可分，但他对这次“容器”变革走向的准确把握却也是功不可没。这也正应了一句名言：一个人的命运啊，当然要靠自我奋斗，但是也要考虑到历史的行程。

眼下，你我可能已经错过了互联网技术大爆炸的时代，也没有在数字货币早期的狂热里分到一杯羹。可就在此时此刻，在沉寂了多年的云计算与基础设施领域，一次以“容器”为名的历史变革，正呼之欲出。这一次，我们又有什么理由作壁上观呢？

如果你也想登上“容器”这趟高速前进的列车，我相信这个专栏，可以帮助你打通学习容器技术的“任督二脉”。**在专栏开始，我首先为你准备了 4 篇预习文章，详细地梳理了容器技术自兴起到现在的发展历程，同时也回答了“Kubernetes 为什么会赢”这个重要的问题，算是我额外为你准备的一份开学礼物吧。**

机会总是留给有准备的人，现在就让我们一起开启这次充满挑战的容器之旅！

选留言(63)



llitfkitfk@dockone.io

如果拿汽车来做比：

Docker好比汽车引擎，

Dockerfile相当于汽车蓝图，

Docker image(镜像)就是汽车样板，

Docker container(容器)类似于汽车的零部件，

Docker Registry可以看作是4s店，

Docker Compose就像老司机，

Docker Volume就像是汽车的油箱，如果把容器间的io数据流比喻成汽油，

Docker Swarm(或者K8s)就是交通枢纽。

2018-09-09

推荐一下<https://github.com/knarfeh/k8s-digitalocean-terraform> 这个项目，可以在 Digital Ocean 上一键启动 kubernetes 机器

2018-09-03

□

□ 20

Twilight26

今天刚用二进制文件的方式，搭建了一套完整的k8s集群，虽然搭建成功了，但是对里面参数还是不是很清楚，所以还需要好好深入学习一下，希望这个课程能带给我大的收获！

作者回复: 二进制绝对是hard模式

2018-08-27

□

□ 16

arcTanh

竟然让我这个半瓶子算法工程师听的热血沸腾啊

作者回复: 相信我，开发人员才是容器技术的真正用户，AI基础设施了解一下

2018-08-28

□

□ 13

@特

我主要涉及到openshift的运维工作，而openshift是基于k8s的一套完整的devops解决方案。我在使用openshift的过程中发现容器化对于无状态的服务确实非常友好，但是对于需要存储的和需要固定ip的服务非常不友好。比如zk集群,MySQL的MGR集群，ES集群等等。部署难度和运维难度都非常高。而statefulsets并没有想象中的好用

作者回复: 有状态应用管理自然是最难的, 要学完整个专栏的所有内容恐怕才有更深的体会

2018-08-28

□

□ 8

付盼星

k8s和yarn到底是相互如何配合呢, 最近看yarn, 容器化进程目前还是实验特性, 官方在未来计划中说, 为了做到暂停机会容器而不是杀死机会容器以最大化资源利用, 需要用到容器技术, 就是最常见的pause功能, 让我在思考, 到底k8s的强项是资源调度还是应用编排呢, 如果两个共同协作, 到底利用了各自哪些优势之处解决了哪些问题。

作者回复: 可以直接给你结论: kubernetes更关注容器与编排, 在资源管理方面优势不大。但正确的姿势, 是用好kubernetes 可扩展能力, 让它跟yarn等一起都发挥出真正的实力。国内阿里和蚂蚁在这一块走得不错。

2018-08-28

□

□ 6

无

您提到: kubernetes更关注容器与编排, 在资源管理方面优势不大。但正确的姿势, 是用好kubernetes 可扩展能力, 让它跟yarn等一起都发挥出真正的实力。国内阿里和蚂蚁在这一块走得不错。

强烈跪求讲一下如何弥补k8s的资源管理缺点, 与yarn结合 这方面的知识与经验阿!!!

作者回复: 说白了就是给kubernetes 写自定义调度器。

2018-09-08

□

□ 5

jerryduren

听了前面的觉得上瘾了，期待更新快一点。

另外希望听到虚拟机容器,裸机容器,虚拟机容器三者间优劣势系统的对比分析方面的内容

作者回复: 当然有

2018-09-04

□

□ 5

赵冬晨

从接触市面上大部分web容器(tomcat.glassfish.weblogic.websphere.jetty等等)，就觉得切换环境部署太费劲了，尤其是要跑兼容性就需要安装不同版本，太难受了...后来接触到了docker，绝对好牛逼啊，真是超级方便，再后来手里服务器多了就接触到了kubernetes，觉得这个更牛逼啊，超舒服，得好好学，这趟车我上了，滴，滴滴

作者回复: 恭喜，容器晋级之路你算是上道了

2018-08-29

□

□ 5

cxyfreedom

搭建环境太痛苦了，还不容易集群搭建成功了，部署插件又出问题，有时候找各种别人的方法还无效

作者回复: 弄懂这些插件的工作原理才是正解。

2018-08-29

□

□ 2

杜鹏

请问后期的课程会介绍k8s和yarn结合的案例吗？能否先给一些资料熟悉一下？现在项目中需要用到k8s的资源管理，总的来说是怎么省钱。

作者回复: 会讲资源管理

2018-09-19