

《代码英雄》第一季（4）： DevOps，拆掉那堵墙

代码英雄讲述了开发人员、程序员、黑客、极客和开源反叛者如何彻底改变技术前景的真实史诗。

什么是《代码英雄》

Command Line Heroes

代码英雄是世界领先的企业开源软件解决方案供应商红帽（Red Hat）精心制作的原创音频播客，讲述开发人员、程序员、黑客、极客和开源反叛者如何彻底改变技术前景的真实史诗。该音频博客邀请到了谷歌、NASA 等重量级企业的众多技术大牛共同讲述开源、操作系统、容器、DevOps、混合云等发展过程中的动人故事。

本文是《[代码英雄](#)》系列播客[第一季（4）：DevOps，拆掉那堵墙](#)的[音频脚本](#)。

当应用开发的争斗暂告一段落，横亘在开发者与运维之间的那堵墙开始崩塌。在墙彻底倒塌的时候，墙两边的人都应该学会如何合作，彼此变得亲密无间。

不过到底什么是 DevOps？开发者一方的嘉宾，包括来自微软的 Scott Hanselman 和 Cindy Sridharan（即 @copyconstruct），他们从开发者的角度认为 DevOps 是一种惯实践。而来自运维一方的成员则他们一直在努力捍卫的东西。双方依然存在差异，不过因为 DevOps 的诞生，大家的合作效率会比之前更上一层楼。这集节目讲述了这种方法的诞生对于大家有多重要。

Saron Yitbarek: 请你想象这样一堵墙：这堵墙从你目之所及的最右侧延伸到最左侧。墙比你高，你无法看见墙背后。你知道墙的另一侧有很多很多人，但你不清楚他们是否和你一样，不清楚他们是敌是友。

00:00:30 - Gordon Haff: 开发者创造代码，然后把代码扔过墙丢给运维，之后发生的问题都是运维的责任了。

Richard Henshall: 他们随心所欲，并不真正关心服务质量。

Sandra Henry-Stocker: 墙这两面的人几乎做着相反的工作 —— 一方做出改变，另一方尽可能抵制那些改变。

Richard Henshall: 但对于他们到底想共同达成什么，却没有在同一幅蓝图中规划过。

00:01:00 - Saron Yitbarek: 我是 Saron Yitbarek，这里是《代码英雄》，由红帽公司推出的原创播客栏目。第四期，我们的标题是《DevOps，拆掉那堵墙》。

是的，数十年来，IT 界划分为各种角色。一边是开发者，他们要尽可能快地去创造更多变化。然后另一边是运维团队，他们要去阻止太多改变发生。与此同时，代码在没有充分共鸣和沟通的条件下，被盲目扔过两方之间的墙。怎样才能拆掉这堵墙呢？这需要一个重大的转变。

00:01:30 - Saron Yitbarek: 开源运动震撼了整个战场。[上一期](#)，我们看到了新的敏捷方法论，它重视不间断的迭代改进，而这种对速度的要求将迫使我们改变彼此的工作方式。一群彼此孤立的人工作的速度是有极限的，而这个极限是一个问题，因为.....

00:02:00 - Richard Henshall: 因为都是为了更快的将产品推向市场、提高敏捷性、更多的迭代，而不是长期而大量的工作。

Saron Yitbarek: Richard Henshall 是 Ansible 的一位产品经理。

Richard Henshall: 是的。还记得你以前下单购买服务器，四个月到货。所有东西都整合在一起，所以整个堆栈是一个整体，要花几年时间来设计和建造那些东西。现在这种情况已经不存在了，对于很多组织来说，这种方法已经.....已经寿终正寝，偶尔拿过来试试，然后放弃它。

00:02:30 - Saron Yitbarek: 如今，像亚马逊这样的公司每分钟都会部署几次新的代码。想象一下，用按部就班的瀑布式工作流，简直不可能完成这些工作。所以为了继续快速完成工作，那些关于稳定性、安全性、可靠性的顾虑都会被运维丢到一边不管。

00:03:00 - Saron Yitbarek: 同时，开发者也没有意识到他们的责任是创造真实环境可用的代码。开发者对稳定性和安全性毫无兴趣，但这些恰恰是我们需要解决的问题。所以，我们最终会有很多无谓的修改，在双方之间来来回回。

想象一下过度分工会如何拖慢公司效率，但开发法者很少被鼓励思考除代码之外的其他事务。

Sandra Henry-Stocker: 他们的目录规模只会越来越臃肿，但他们从不清理。除非已经无法工作才不得不清理。

00:03:30 - Saron Yitbarek: Sandra Henry-Stocker 是位退休的系统管理员，为 IDG 杂志撰稿。

Sandra Henry-Stocker: 我过去经常劝说别人，说，“嘿，你看，你用了这么多的磁盘空间。是不是有什么东西你可以整理一下，这样我们就有更多的存储空间来运行了——因为服务器上的存储空间快用完了。”是的，我们经常经历这些事。

Saron Yitbarek: 归根结底，这是一个心态问题。这种开发者和运维之间的态度分裂，一方不必去理解另一方的担忧。好吧，在过去这还没太大问题，但随着开发速度成为一种重要的优势，这种分裂的文化急需改进。孤立在自己的工作圈子里，效率太低了。

Jonah Horowitz 在 Stripe 的可靠性工程团队工作。他描述了即使开发人员和运维人员想一起工作，他们也无法做到。因为从某种意义上说，他们被安排在对立的团队中。

00:04:30 - Jonah Horowitz: 运维团队经常以正常运行时间和可靠性来衡量，而提高正常运行时间的最大方法之一，就是减少系统的变化量。当然，发布新功能就是在改变系统，而做产品工作的软件工程师有动力去尽快发布尽可能多的功能。所以，当开发和运维的职责不同时，他们自然有了冲突。

00:05:00 - Saron Yitbarek: 开发者致力于新建功能，运营致力于维持运行，两者目标相互矛盾。但就像我说的，由于对速度的需求越来越大，对快速迭代发布的需求越来越大，开发和运维之间的脱节已经到了一个临界点，必须要有所改变。

00:05:30 - Saron Yitbarek: 在 2009 年左右，将开发和运维分开的那堵墙看起来像是监狱的墙。我们需要的是一种新的方法论，它能使开发和运维之间的隔阂顺畅过度，让双方以更快、更整体化的方式工作。

视频平台 Small Town Heroes 的首席技术官 Patrick Debois 为想要拆掉这堵墙的人发起了一场会议。他把这个脑洞叫做 DevOps Days（开发运维日）。为了便利，他将其缩短为 DevOps，于是这场改革就有了名字。

00:06:00 - Saron Yitbarek: 不过名字并不代表改革的一步，我们知道为什么我们需要 DevOps，但究竟该如何做？我们应该如何将开发和运维结合起来而不引发战争？幸运的是，我有 Scott Hanselman 来指导我。Scott 是微软 .NET 和 ASP.NET 的首席项目经理。

所以，Scott，我认识你确实有几年了，但感觉还是相见恨晚啊。

00:06:30 - Scott Hanselman: 我也是，相见恨晚哈。

Saron Yitbarek: 我想和你聊聊你如何成为一个开发者，和 DevOps 这些年的变化。觉得如何？

Scott Hanselman: 嗯，听上去挺有意思。

00:07:00 - Saron Yitbarek: 好的。我认为究竟什么是 DevOps 是一个好的开场问题。你会怎么定义它呢？

Scott Hanselman: 在 2008 年，维基百科有个关于 DevOps 的定义确实很棒。它说，这是一套“惯例”，目的是在保证质量的前提下，缩短提交变更和变更投入生产之间的时间。所以，如果你想想，假如今天是周二，我检查了一些代码，而这些代码将在 6 月的版本中上线。这就很糟糕了，因为这不是持续集成，而是一年几次的集成。

00:07:30 - Scott Hanselman: 如果你有一个健康的 DevOps 体系，如果你已经有“设置 - 上线”的惯例，那么你就可以不断地将代码集成到生产中去。那么，你能做什么？你可以定义、创造怎样是最佳“惯例”，这将决定你能否成功。所以，我在周二检查的一些代码，周四就上线了。那么现在，为了保证高质量，最重要的事情就会是 —— 谨慎上线。

00:08:00 - Saron Yitbarek: 这个定义真的很有趣呢，是个“惯例”。但我觉得当我听人们谈论 DevOps 时，它具体一点。他们谈论它就像它是一个角色、一个工作、一个职位或一个头衔。你觉得这与它是一套“惯例”的观点是否有冲突？

Scott Hanselman: 我认为，当一套新的方法或一个新的流行语出现时，人们喜欢把它加在名片上。我不是不尊重那些正在收听这个播客，并且感到被我冒犯、正骂骂咧咧把名片掏出来看的人们。虽然，他们现在可能正要愤怒地关上笔记本电脑、退出这个播客。

00:08:30 - Scott Hanselman: 有一个帖子写得非常好，作者是 Brian Guthrie，他是一个脑力劳动者，在 SoundCloud 工作。他是一个超级聪明的人，几天前他在 Twitter 上的帖子中说到 DevOps。他说 DevOps 就是一套惯例，不是一个工作头衔、不是一个软件工具、不是一个你安装的东西、也不是一个团队的名字。

00:09:00 - Scott Hanselman: 他的原话是：“DevOps 不是神奇的‘企业万能药’”。如果你没有好的惯例，如果你没有良好的习惯，你就没有 DevOps。所以，这更多的是一种心态，而不是摆出一个工作头衔，然后“我们要雇佣一个 DevOps 工程师，然后我们要把这些神奇的 DevOps 工程师撒到组织中。虽然整个组织没有意志力，也没有信奉 DevOps 的想法。”所以，如果你认为 DevOps 是一个工具或者是用来安装的东西，那么你就完全理解错了。

00:09:30 - Saron Yitbarek: 好吧，让我们回到过去，在 DevOps 这个名词出现之前，在我们往名片上写 DevOps 或者把它作为一套“惯例”来讨论之前。在 10 年前，你会如何描述开发者和那些运维人员之间的关系？

Scott Hanselman: 那是相当的水火不容。举个例子，运维控制着生产，但开发人员从来没有接近过生产。我们站在一堵不透明的墙的两侧。我们在开发部的时候，尽可能地去做一些看起来像生产环境能用的东西，但实际上从来没有.....从来没有像样的产品。

00:10:00 - Scott Hanselman: 我们有相当多问题。我们的开发环境从各个方面来说都不像生产环境，所以你不可避免地会遇到那些“嘿，它在生产环境中的工作方式和在开发环境中的不同”的问题。然后，从开发到投入生产之间的间距是几周几周的长久间隔，所以你的大脑甚至不在正确的频道上。比如说，我在一月份的时候就在研究这个功能，现在四月份才刚刚上线，那么当 bug 不可避免地出现的时候，要等到六月份才能修复，我甚至不记得我们之前在干嘛。

所以运维团队的人，他们的工作是.....他们的工作几乎就是有意识地让我们慢下来。好像他们的存在是为了让开发人员更慢，然后他们还觉得我们随时会让生产环境崩坏。

00:11:00 - Saron Yitbarek: 那么为什么会这样呢？是对开发者想要做什么和他们做了什么不了解？还是信任问题？为什么会有这么大的冲突？

Scott Hanselman: 我觉得你已经回答了，而且回答得很到位。你说的很对，确实是信任的问题。我觉得开发人员认为他们是特殊的，或者某些方面比 IT 人员更优越，而 IT 人员认为开发人员不尊重生产。

00:11:30 - Scott Hanselman: 我认为这种文化的产生，一部分来源于高层。他们认为我们是不同的组织，并且我们的目标也不同。我认为软件业正在走向成熟，因为我们都意识到，无论业务是什么，我们写软件都是为了推动业务发展。

所以现在有种“我们都在往正确的方向推进”的感觉，就像他们说的，“专注一件产品并做到极致”。但这是需要绝对的信任，可 DevOps 工程师不信任产品工程师来部署代码，对吧？

00:12:00 - Scott Hanselman: 但 DevOps 工程师传统上并不写代码，所以他们并不了解什么被修改了。所以他们对于在各个层面的人都缺乏信任。没有人理解部署过程，人们只信任自己，他们的心

态.....举个例子，就像“我只信任自己的工作。我不能相信 Saron 的工作，她甚至不知道她在干些什么。我会做完所有的事情。”

00:12:30 - Scott Hanselman: 所以如果没有人真正理解这个系统，那么全栈工程师的概念就是一个神话。但是现在，我们开始将一整个组织称之为全栈。我们已经有了全产品所有权这样的名词，敏捷方法论也出现了，也就是说每个人都应该拥有产品。社区对于软件所有权和对于代码的想法都慢慢发生了变化，这种改变带来了一个充满信任的环境。

00:13:00 - Saron Yitbarek: 我是 Saron Yitbarek，你现在收听的是《代码英雄》，来自红帽公司的一档原创播客栏目。所以，要想让 DevOps 发挥出它的潜力，我们就需要双方都有更多的信任，这就意味着要有更多的沟通。回到 Richard Henshall 身上，他认为双方的共情是 DevOps 的基石。

00:13:30 - Richard Henshall: 一些 DevOps 的从业者，一群真正优秀的从业者，都参与过这两种角色。我认为这才是真正的力量所在——当人们真正做过了两种角色，而不是只看到其中一种。所以，你不该保持孤立，你实际上.....你应该去和双方都一起工作一段时间。我想这才是让人恢复同理心的方法。

Saron Yitbarek: 现在，这不仅仅是为了温情的沟通。Richard Henshall 所描述的是行业重点的转向——Scott 刚刚提到过。

00:14:00 - Saron Yitbarek: 一个关于持续集成（CI）的观点。软件不仅要以小批量快速编写和发布，还要以小批量进行快速测试。这意味着，开发人员需要即时反馈他们正在编写的代码在现实世界中的表现。

随着上市时间从几个月缩短到几天，再到几个小时，我们四处寻找一套新的工具，可以将任何可以自动化的元素自动化。

00:14:30 - Gordon Haff: 你需要一个全新的生态系统和工具，来最有效地进行 DevOps。

Saron Yitbarek: Gordon Haff 是一位红帽公司高级主管。

Gordon Haff: 我们看到有很多巨大的、DevOps 可以利用的新种集合工具和平台，它们都诞生于开源。

Saron Yitbarek: Gordon 是对的。新的集合工具是很庞大，关于开源这点他说的也对。在一个严格的专有系统中，自动化工具是不可能发展的。

00:15:00 - Gordon Haff: 其中有很多监控工具，Prometheus 是其中一个常见的工具。它开始引起很多人兴趣，用于编排服务的 STO 也出自这里。

Saron Yitbarek: GitHub 让你跟踪变化，PagerDuty 管理数字业务，NFS 可以跨网络挂载文件系统，Jenkins 让你自动测试你的构建。

00:15:30 - Saron Yitbarek: 这么多工具，这么多自动化流程。最终的结果是，开发人员可以将他们的变更直接推送到生产现场，自动创建构造，实行经过严格管理的编译与针对性的自动测试。Sandra Henry-Stocker 描述了这是怎样的变化。

Sandra Henry-Stocker: 所以，我可以把我正在工作编写的东西快速部署。我可以只在一个系统上，通过命令行控制许多系统，而不是必须在很多不同的系统上工作，也不用学习就可以利用网络，将代码部署到其他机器上。

00:16:00 - Sandra Henry-Stocker: 现在，在计算机系统中进行改动更容易了。坐在一个地方，就能实行一切操作。

Saron Yitbarek: 自动化工具已经解决了速度问题。但我不希望我们只赞美工具，而忽略了实际的方法论，文化的转变。Scott Hanselman 和我谈到了这个微妙的界限。

00:16:30 - Saron Yitbarek: 你在这次谈话开始时说，DevOps 是一套惯例，是一种心态，是一种思维方式。听起来，我们创造的工具是我们应该思考和操作方式的具体代码实现。

Scott Hanselman: 我喜欢这句话，你真是个天才。没错，我们以前让产品开发在 Word 文档中写下这些代码是如何工作的。他们写的是

规范，对吧？这些文档过期了吗？

00:17:00 - Saron Yitbarek: 没错。（答非所问）

Scott Hanselman: 哈？

Saron Yitbarek: 好吧，我只是很高兴 Scott 刚才说我是天才。但我也确实认为，这些工具几乎是我们文化转变的象征。它们鼓励我们拓宽我们的角色定义。我们开发者已经被迫，至少偶尔关注代码的运行。这样一来，开发和运维的主要职责就部分统一了。事实上，DevOps 的兴起告诉我们的是，在一个速度不断提升的世界里，没有人能够保持孤岛状态。

00:17:30 - Saron Yitbarek: Jonah Horowitz 曾在湾区多家公司工作，包括 Quantcast 和 Netflix。他说即使是世界上一些最大的公司，也从这个角度重新塑造了他们的文化。

Jonah Horowitz: 我们在文化上得到了整个公司的认同，就像，“这就是我们要部署软件的方式，我们将以小批量的方式进行部署，我们将使用这些程序帮助部署。”如果 DevOps 只是被运营团队所驱动，我不认为它可以……我不认为它可以成功。

00:18:00 - Jonah Horowitz: 这必须成为公司的管理层和领导层所认同的东西才能发挥作用，而这件事很大程度上，意味着一种文化转变。

Saron Yitbarek: 当 MacKenzie 对 800 名 CIO 和 IT 高管进行调查时，80% 的人表示，他们正在规划让自己的一部分下属组织实施 DevOps，超过一半的人计划到 2020 年，在全公司范围内实施。高管们意识到，自动化工具可以提升交付速度。

00:18:30 - Saron Yitbarek: 这些人过去也是这样的人，他们习惯于让一个货板先到达数据中心，然后在新机器上线之前让它在那里放上整整一个月。不过在今天，如果你等待的时间超过 10 分钟，就说明你做错了什么。随着竞争对手的速度增加，没有人能够承受得起落后。

00:19:00 - Saron Yitbarek: 我可以想象，运维团队一定很紧张，因为他们把所有的工具都交给开发人员。运维团队习惯了做成年人，而现在叫他们把车钥匙交给一贯的孩子——开发人员？呀，我想我们开发人员正在学习，如何在不破坏东西的前提下快速移动。但随着 DevOps 革命的尘埃落定，变化最大的可能是运维团队。

00:19:30 - Saron Yitbarek: DevOps 是否真的威胁到了运维的存在？开发是否在用它闪亮的新工具来吃掉运维？**Cindy Sridharan** 是一位开发者，她写了一篇长篇调查文章来讨论这些问题。在你的文章和博客中，你提到运维人员对事情这样的发展并不一定满意。到底发生了什么？你会说什么？

Cindy Sridharan: 这么说吧，DevOps 的理想是责任共享。开发者和运维将有，就像你知道的，更多的是五五分成，以真正确保软件的整体交付。

00:20:00 - Cindy Sridharan: 我认为很多运维工程师的不快乐源于这样一个事实，那就是这些改变都没有实际功效。他们仍然是总被鸡蛋挑骨头的人，他们仍然是总做苦力工作的那些人，他们还是那些主要肩负着实际运行应用的责任的人，而开发者不一定要做得足够好。

Saron Yitbarek: 这个问题在未来几年将至关重要。DevOps 的作用到底有多大？随着我们的自动化进程，运维的作用是被削弱，还是会发生转变？

00:20:30 - Saron Yitbarek: 但是我们要记住，DevOps 不仅仅是工具和新技术的应用。这种方法论实际上是在塑造技术，反过来技术也在塑造方法论，这样就有了一个神奇的反馈循环。文化造就了工具，而工具又强化了文化。

00:22:00 - Saron Yitbarek: 最后，我们在节目开头描述的那堵墙，也就是把开发和运维划分开来的那堵墙，我甚至不知道五年后“把你的代码扔过墙”的比喻对一个开发者来说是否有意义，如果五年后大家都听不懂这个比喻，那真是一件大好事。不过目前为止的访谈很有价值，我听到了很多新的故事。

现在说话的是云架构师 Richard Henshall。

Richard Henshall: 我觉得 DevOps 开始让人们意识到对方关心的是什麼，我看到了更多对彼此的理解。

00:23:00 - Saron Yitbarek: 现在是系统管理员 Jonah Horowitz。

00:23:00 - Jonah Horowitz: 我认为你需要很棒的技巧来写出真正好的软件，我在与我合作过最棒的开发者身上看到了一件事，那就是他们真的，他们贡献了关于的软件工程新技巧，或者说他们推动了软件开发这个行业的发展。

Saron Yitbarek: 最后一个是系统管理员 Sandra Henry-Stocker。

Sandra Henry-Stocker: 我认为，开发者会变得更加精明、更加谨慎。他们始终要提升自己的技能，我知道这需要很多辛苦的学习。

00:23:30 - Saron Yitbarek: DevOps 是个爱的结晶。原来，在那堵墙的另一边还有一些朋友，很高兴认识你们。所以，坦白一下，我以前总觉得 DevOps 很无聊，就是一堆硬核的自动化脚本和扩展问题。我的抵触情绪有一部分是出于它的实践难度。作为开发者，我每周都要面对一些新出来的工具，一些新的框架。DevOps 一直是那些可怕的、快速变化的一部分。但现在，尤其是听了这些故事之后，我明白了。

00:24:00 - Saron Yitbarek: DevOps 不仅仅是其工具。它是教导我们如何合作，更快地构建更好的产品的方法。

好消息是，随着为我这样的开发者准备的新平台出现，我们的工作变得更好、更快、更能适应不同的环境，我的业务圈也可以不断扩大。你会看到人们将 DevOps 扩大到安全部分，所以我们能得到 Sec DevOps。或者他们开始包含商务，那我们就得到了 Business DevOps。我们现在要辩论的话题是：对于一个开发者来说，不仅要了解如何使用这些工具，还有了解所有 DevOps 的东西是如何工作的必要吗？以及我们需要所有开发者都去了解这个新世界吗？

00:24:30 - Saron Yitbarek: 这场辩论的结果将决定未来一期《代码英雄》的内容。

你可能已经注意到，在所有关于工具和自动化的谈话中，我漏掉了一些工具。好吧，我把这些留到下一期，当所有这些 DevOps 自动化达到光速时，我们将追踪容器的崛起。

00:25:00 - Saron Yitbarek: 是的，这些都会留到第五期。

《代码英雄》是红帽公司推出的原创播客栏目。想要了解更多关于本期节目和以往节目的信息，请访问 redhat.com/commandlineheroes。在那里，你还可以注册我们的新闻资讯。想免费获得新剧集的自动推送，请务必订阅该节目。

00:25:30 - Saron Yitbarek: 只要在苹果播客、Spotify、Google Play、CastBox 中搜索《代码英雄》，或者通过其他方式收听，并点击订阅，这样你就能在第一时间知道最新剧集。我是 Saron Yitbarek。感谢您的收听，编程不止。

什么是 LCTT SIG 和 LCTT LCRH SIG

Special Interest Group
LCTT SIG 是 LCTT 特别兴趣小组，LCTT SIG 是针对特定领域、特定内容的翻译小组，翻译组成员将遵循 LCTT 流程和规范，参与翻译，并获得相应的奖励。LCRH SIG 是 LCTT 联合红帽（Red Hat）发起的 SIG，当前专注任务是《代码英雄》系列播客的脚本汉化，已有数十位贡献者加入。敬请每周三、周五期待经过我们精心翻译、校对和发布的译文。

欢迎[加入 LCRH SIG](#)一同参与贡献，并领取红帽（Red Hat）和我们联合颁发的专属贡献者证书。

via: <https://www.redhat.com/en/command-line-heroes/season-1/devops-tear-down-that-wall>

作者: [Red Hat](#) 选题: [bestony](#) 译者: [LikChung](#) 校对: [acyanbird](#), [FineFan](#)

本文由 [LCRH](#) 原创编译, [Linux中国](#) 荣誉推出