

《代码英雄》第三季（4）：深入 Perl 语言的世界

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[《代码英雄》第三季（4）：深入 Perl 语言的世界](#)的[音频](#)脚本。

导语：语言来了又走。只有少数几种语言具备了登峰造极的能力，而能留在那里的则更少。Perl 有过一次惊人的崛起，也有过一次平静的低迷，现在已经在编程世界中找到了自己的位置。

Perl 似乎注定要统治网络。Michael Stevenson 和 Mike Bursell 描述了 Perl 的设计如何使它成为早期 web 的理想选择。我们从 Conor Myhrvold 那里听到了它的座右铭：“

There is more than one way to do it.

实现它的方法不止一种”。Elizabeth Mattijsen 分享了，尽管 Perl 有优势，但漫长的开发周期如何减缓了 Perl 的发展。虽然它不再是最顶级的网络语言，但 John Siracusa 指出，Perl 作为一种小众工具仍然存在。

00:00:02 - Saron Yitbarek:

想象一下 Perl 语言的创建者 Larry Wall 在 1999 年的一次会议上站在麦克风前，留着他标志性的浓密胡子和梳理过的刘海，感觉相当好，因为他所发明的语言正在越来越受欢迎。

00:00:19 - 配音演员：

声音测试。

00:00:19 - Saron Yitbarek:

Perl 语言轻而易举地超越了 COBOL、Visual Basic，而 Python 呢？Python 仍然是一个不入流的竞争者，是 Larry Wall 的几个玩笑素材。Wall 展示了 dice.com 报告中的人群数据，在那时来看，Perl 语言的未来是非常、非常光明的。然而之后 Perl 的未来就不再如此了。20 年之后，dice.com 在 2018 年夏天将 Perl 列为最可能灭绝的语言之一。短短 20 年之间发生了什么？

00:00:59 :

我是 Saron Yitbarek，这里是《代码英雄》，一档红帽公司的原创播客。这一季是关于我们编程语言的力量和前景的。在上一集我们追踪了搭上了互联网顺风车的 JavaScript 的疯狂崛起。

00:01:19:

并不是每一种语言都有一个不停地成长和成功的故事。大多数语言的出现，在当时非常特殊的生态系统中发挥了它们的作用，然后当我们的编程生活里需要其他类型的工具时它们就开始消退。

00:01:37:

本集我们将深入了解 Perl 世界。是什么造就了它早期的成功，为什么它的成功突然就被颠覆了？我们所处的环境，我们的基础设施，我们的硬件，各种因素都会决定哪些语言会繁荣，哪些会开始萎缩。这就是 Perl 语言的故事的魅力所在。

00:02:08:

我们知道 Perl 并没有统治世界，但是退回到上世纪 90 年代的时候，却看不到这一点。Tim Berners-Lee 在 1991 年发明了

World Wide Web

万维网（WWW），它迅速创造了一个全新的基于 Web 的开发领域。谁也说不准会是哪种编程语言在这个新领域取得成功。

00:02:31 - Michael Stevenson:

在 Web 出现的时候，所有人都等待着会有什么事情发生。那个时候整个世界都是令人兴奋的。

00:02:39 - Saron Yitbarek:

University of Amsterdam

Michael Stevenson 是阿姆斯特丹大学媒体研究的副教授。他为我们描述了早期的 Web。人们见过 Usenet，也见过 ARPANET。他们看到了美国在线，还有一些其它的东西。但是直到 Web 出现，互联网的全部发展潜力才真正得到体现。突然之间，你就可以通过鼠标和点击进入了这个巨大的互联世界。这是一个出乎意料的事情。

00:03:09 - Michael Stevenson:

你要是记得 1993 年，也就是 Web 开始崭露头角的那一年，也是

Wired Magazine

《连线杂志》开始出版的那年。在那之前类似《Mondo 2000》这类的杂志真的让电脑看起来像神秘知识的来源，让电脑看起来很酷。

00:03:32:

因此从这个意义上说，Web 也到达了一个相当特定的时期，人们已经准备好以这样的方式迎接技术的兴起。

00:03:43 - Saron Yitbarek:

故事在这个时候开始了：Larry Wall 在 1987 年创建了 Perl，4 年后万维网才开始兴起。Larry Wall 给世界带来的 Perl 最初只是一种通用的 Unix 脚本语言。当然，它确实很有用，但同时 Perl 还有一些秘密元素，这些元素使它成为即将到来的 Web 开发世界的理想语言。

00:04:14 - Michael Stevenson:

比较有名的是 Perl 是 Larry Wall 在参与的一个美国国家安全局（NSA）的秘密项目中发明的，基本上他所做的就是创建一个类似黑客版的新闻栏目，运行在 Usenet 上，因此 Perl 语言从一开始就是操作文本和将数据从一个地方移动到另一个地方，这完全符合 Web 的需求。而 Perl 作为一种易于使用的脚本语言，更加接近于自然语言。它可以用来快速开发东西，所有这些都让 Perl 成为了一个完美的契机，不仅适合专业用户，也适合新加入的业余爱好者。

00:05:09 - Saron Yitbarek:

很偶然的是，当 Web 出现时，Perl 已经为 Web 做好了准备。Larry Wall 不可能知道 Web 即将出现。但当它出现时正好相互吻合。但我认为还有一点很关键：Perl 是一种自由语言。Larry Wall 在
Free Software Foundation General Public License
自由软件基金会开发的 GPL（通用公共许可证）下发布了它。

00:05:37:

Larry Wall 让他的语言自由开源的决定，这完全符合刚刚开始出现的基于 Web 的新思维方式，同时 Perl 在其他方面也很有前瞻性。

00:05:50 - Michael Stevenson:

Perl 的开放性令人难以置信，在某种意义上，它总是愿意整合一些其他的東西，新的东西。这和 Larry Wall 的身份很相配。他是个很开放谦虚的人，总是仔细考虑别人的想法，并试着设身处地为别人着想。对我来说 Perl 作为一种语言，以及作为一个社区的性格，在很长一段时间里，真的很符合这一点。

00:06:27 - Saron Yitbarek:

同样 Perl 非常适合 Web 早期的、狂野的西部阶段，也就是黑客时代。实际上 Perl 的座右铭之一就是“实现它的方法不止一种”。
There is more than one way to do it.

00:06:39 - Michael Stevenson:

在 Perl 的鼎盛时期和这个试验性的开放 Web 的鼎盛时期之间，存在着一种近乎浪漫的联系，在它变得如此被几个平台所控制之前。

00:06:56 - Saron Yitbarek:

记得上世纪 90 年代是互联网历史上的一段启蒙时期，那时人们还在争先恐后地想知道还有哪些是可能的。那时对编程的需求是巨大的，每个人都需要有一个网页，这意味着一群全新的开发人员，他们都对新的做事方法持开放态度。问题变成了“好吧，我们有了一个全新的领域，但用什么语言来完成这项工作呢？”

00:07:26:

Perl 虽然并不是这个问题的最终答案，但对于很多人来说 Perl 是他们首选的答案。

00:07:34 - Michael Stevenson:

我并不是说我更愿意看到加载速度超慢的网页、也没有谷歌搜索引擎的时代，但我确实认为那个时代有一些特别的东西，当时有一些人，在他们的宿舍里，创建了一个类似 **slashdot** 的东西。而现在随着 Web 已经变得如此主流、专业化和被集中在几个大公司中，我们确实怀念那个时代。对我来说，Perl 比其他任何早期的故事都更能象征这一点。

00:08:15 - Saron Yitbarek:

Michael Stevenson 是阿姆斯特丹大学媒体研究的副教授。

00:08:24 - Saron Yitbarek:

之后，随着 90 年代的到来，Perl 作为一种能适应早期互联网发展潜力的语言出现了，它是一个时代的语言。Larry Wall 和他所创造的 Perl 正好明白了 Web 的本质。

00:08:40 - Mike Bursell:

在网上你可以随意搜索，也可以随时创建网页，这是互联网的美丽新世界，你可以随时做这些事情。

00:08:52 - Saron Yitbarek:

这位是 Mike Bursell，红帽公司的首席安全架构师。Mike 是 90 年代中期发现和使用 Perl 的黑客之一。

00:09:00 - Mike Bursell:

对于 Web 来说，Perl 是许多人的起点。Java™ 语言还处于早期阶段，它在文本输入输出方面并不是很出色。如果你想进行查询和生成页面，Perl 是人们都在使用的工具。

00:09:22:

Perl 非常适合，因为它擅长获取文本，并使用文本做其他事情，而这正是 Web 所需要的。

00:09:31 - Saron Yitbarek:

顺便需要说一下的是，Larry Wall 有语言学背景，这就解释了为什么 Perl 有强大的文本解析能力。正如 Mike Bursell 提到的，这是一笔巨大的好处，因为在早期，Web 主要是一种基于文本的媒介，因为那时候人们没有足够的带宽来处理图像。

00:09:51 - Mike Bursell:

它很容易使用，也很容易复制。人们在分享方面非常开放，而且它的输出速度很快，这些都是好东西。

00:10:02:

哦，当然还有一点，就是你可以用管道使用它。所以，这是非常多的人所习惯的，而且非常容易测试，甚至离线测试，这都是非常有用的。

00:10:13 - Saron Yitbarek:

尤其对那些在 Web 的世界中重新规划自己生活的系统管理员来说非常有用。

00:10:21 - Mike Bursell:

Perl 是给系统管理员们的真正礼物。即使在那个年代，如果你做一些事情，你会得到很多日志。管理这些日志，分解它们，搜索它们，并能够以不同的方式显示它们，或获取任何其他大型文本语料库（主要就是日志），甚至可以对它们进行调试。除非你要在命令行里用管道方式传输 `orc`、`sed` 之类的东西，以及 `ed` 命令，那样的话你很快就会变得非常痛苦，而 Perl 正好适合让你去处理这些事情。

00:10:55 - Saron Yitbarek:

到上世纪 90 年代后期，Perl 5 已经聚集了一个强大的用户群体。像 Fortran 和 C 这样的旧语言依赖于庞大而昂贵的硬件，而 Perl 更有生命力，也更容易移植。在那样一个硬件成本急剧下降的世界里，Perl 的设计使得它得到了蓬勃发展，Perl 让所有的新程序员快速、轻松地工作。这是因为 Larry Wall 以牺牲 CPU 和内存的需求为代价，使 Perl 语法更人性化。所有这些元素组合在一起，使 Perl 成为一种很受新开发社区欢迎的语言。

00:11:36 - Mike Bursell:

在一个正在壮大的社区里，你可以去和他们聊聊社区里的事情，然后 PerlMonks 论坛出现了，那里是一个讨论的好地方，能在论坛里知道正在发生的事情。

00:11:50 - Saron Yitbarek:

这个社区确实拥有 Web 所能提供的最好的东西。他们发现了一个巨大的软件模块库，一个叫做 CPAN 的存储库，这些系统管理员都很喜欢它。它给 Perl 提供了更多的灵活性，许多人都可以部署由少数几个编程大师编写的代码。

00:12:15 - Mike Bursell:

它有很多库可以做你想做的任何事情，如果你找不到你想要的库，你可以去问一下，然后就会有好心人写出你想要的库。

00:12:21 - Saron Yitbarek:

Mike Bursell 是红帽公司的首席安全架构师。

00:12:28 - Saron Yitbarek:

正是由于 Perl 是免费的，它受到不断增长的模块库的支持，它是可移植的，而且它有一个蓬勃发展的社区。一切看起来都很好。Perl 可以在上世纪 90 年代 Web 开发新天地中发挥所有优势，但就在 90 年代即将结束的时候，互联网的发展前景又一次发生了变化，时代来了一个大的转变。

00:12:57 - Alan Greenspan:

但我们又能如何知道非理性繁荣何时已过度推高了资产价值，进而导致了意料之外的长期经济收缩？

00:13:12 - Saron Yitbarek:

“非理性繁荣”是时任美联储主席 Alan Greenspan 在 1996 年对 American Enterprise Institute 美国企业协会所说的话。他那句“非理性繁荣”是对 90 年代人人都经历过的网络泡沫的警告。所有早期使用 Perl 的 Web 开发人员都在那个泡沫中乘风破浪，但正如 Greenspan 预测的那样，泡沫在 2000 年破裂了。

00:14:11 - Conor Myhrvold:

大家好，我是 Conor Myhrvold。在过去的五六年里我一直从事编程，现在在技术领域为 Uber 工作。

00:14:20 - Saron Yitbarek:

2000 年代初，当 Conor 还在高中的时候，Perl 仍然是一个非常重要的东西。但他越来越意识到一种与之竞争的语言，叫做 Python。

00:14:31 - Conor Myhrvold:

Python 所追求的是一种更结构化的语言，做很多不同的事情都有一种更显然的方式，它就是那样设计的。而 Perl 则喜欢做某件事有不止一种方法，但这让很多初学者感到困惑。

00:14:49 - Saron Yitbarek:

There is more than one way to do it.

Perl 有这样一句座右铭“实现它的方法不止一种”。而 Python 的理念实际上是相反的，Python 为每个问题都提供了一个明显的解决方案，这意味着查看别人的 Python 代码很容易；而另一方面查看其他人的 Perl 代码可能会令人困惑。Perl 作为一个程序员的第三或第四种语言是有意义的，而正是因为它是一种脚本语言，而脚本是互联网连结的基础。

00:15:23:

但是 Python 是一种你可以真正深入研究的语言，即使你是一个新手。Perl 有一套特定的优势，比如在搜索文本和生物信息学上。但是 Python 就是这样一种简单通用的语言。Python 获得了越来越多的信任，成为人们首先想学的语言，这是一件大事。

00:15:47 - Conor Myhrvold:

越来越多的人开始上网，越来越多的人开始学习如何编程。尤其是相对于 Perl 而言，Python 受益于它本身相对容易学习，因为它更结构化。而这样结果是，在一个快速增长的世界里，如果你能得到更多增长的份额，那将最终意味着有更多的教程和更多的东西可供你使用。

00:16:10 - Saron Yitbarek:

在之前提到过的 CPAN，它是 Perl 用户可以使用的强大的中央存储库。这在 90 年代是一大亮点，但 CPAN 的价值也在变化。

00:16:24 - Conor Myhrvold:

这也不能真正帮助你学习一门语言，因为你是在“复制粘贴”，只是用最少的方式替换你需要的东西。从长远来看这是一个劣势，因为如果你让人们通过自己进行原始开发来学习如何使用一种编程语言，即使这需要花费更长的时间，他们也会觉得自己对它投入了更多，而且他们也了解在这中间发生了什么。

00:16:48 - Saron Yitbarek:

Python 没有像 CPAN 那样的集中式存储库，但是对于那些在新千年时代来到这里的开发人员来说，在一个互联网搜索功能如此强大的世

界里，存储库并没有那么大的价值。

00:17:05 - Saron Yitbarek:

最终 Python 有了大量的教程，当然现在也有了像 GitHub 这样的平台。

00:17:13 - Conor Myhrvold:

最终发生的事情是 Perl 拥有的许多优势，是来自一个已经过时的时代的网络效应。

00:17:24 - Saron Yitbarek:

Conor Myhrvold 是 Uber 的一名工程师。

00:17:30 - Saron Yitbarek:

然而语言的兴衰很少是由外部力量单独决定的，而 Perl 的内部问题是，在它的发展过程中它似乎遇到了障碍。Python 正在以一种相当有序的方式发布新的迭代，而正如我们在本季度第一集中所了解到的，Perl 在 2000 年互联网泡沫破裂之时，Python 开始获得更多新开发人员的青睐。

00:17:59 - Saron Yitbarek:

每个人都期待着 Perl 6 的发布，人们都很兴奋。他们等啊，等啊，等啊.....他们等了 14 年。

00:18:15 - Elizabeth Mattijsen:

人们提出了大约 300 多件 Perl 6 应该能够完成的事情，当然其中很多事情基本上都是相互排斥的。

00:18:26 - Saron Yitbarek:

这是 Elizabeth Mattijsen，她是 Perl 6 的核心开发人员。2000 年，Elizabeth 参加了在蒙特雷^{Monterey}举办的 Perl 会议。那时开发者认为他们已

经停滞不前了，所以 Perl 6 是必要的。Larry Wall 同意了，但是如果说 Perl 5 是他对 Perl 的重写，那么他希望 Perl 6 是由社区来对 Perl 进行重写。由于团队合作可能需要更长时间，甚至用了 14 年，对于那些开发者来说，这是一条漫长而艰难的道路。

00:19:01 - Elizabeth Mattijsen:

我们可以说当前实施的 Perl 6 项目实际上是实现它的第三次尝试。

00:19:07 - Saron Yitbarek:

按照 Elizabeth 的说法，在这 14 年里有过多次尝试。中间经历了漫长而痛苦的深度的尝试。开发者们心力交瘁；人们陷入了死胡同。到 2015 年圣诞节那天 Perl 6 终于问世时，世界上的许多地方已经开始了新的发展。而需要注意的是 Perl 6 并没有给予成为某种革命性的新事物，从而实现对 Python 的反击。Perl 6 是对原版进行了深思熟虑的重新设计。

00:19:43 - Elizabeth Mattijsen:

我认为 Larry Wall 在他的“^{State of the Onion}洋葱状态”演讲中使用了一个很好的比喻。对他来说，Perl 5 就像《^{The Hobbit}霍比特人》，而 Perl 6 就像《^{Lord of the Rings}指环王》。如果你仔细看过《霍比特人》和《指环王》的故事，你会发现它们基本上是同一个故事。只是《霍比特人》比《指环王》小得多，情节漏洞也更多，没有《指环王》那么宏大的背景。我认为这很好地描述了 Perl 5 和 Perl 6 之间的区别。它基本上是同样的想法，同样的思路，同样的环境，只是对它的重新构想。

00:20:26 - Saron Yitbarek:

Elizabeth Mattijsen 是 Perl 6 的核心贡献者。

00:20:32:

如今，Perl 甚至可能不在前 20 种语言之列。在外部竞争和内部拖延之间，它还没有向大多数新开发人员证明自己。但这提出了一个大问题，我们真的应该根据一种语言的流行度来判断我们的编程语言吗？

或者说我们应该根据其他方面来判断一种编程语言的价值？当昔日的超级巨星成为陪衬时，这到底意味着什么呢？

00:21:06:

在世纪之交时互联网泡沫破裂时，Perl 的统治地位开始衰退时，Larry Wall 发表了一个有趣的声明。他认为尽管 Perl 永远不会再成为世界上最流行的编程语言，但它可以成为较小类别中的领先者。Larry Wall 说那才是真正的目标。成为同类中最好的，而不是世界上最好的。正如他所说的，SUV 永远不会和赛车竞争。

00:21:38 - Saron Yitbarek:

我想深入研究这个想法，我想了解在细分类别中做到最好对编程语言的真正含义。

00:21:48 - John Siracusa:

我是 John Siracusa，我是一个程序员，也是一个播客主。

00:21:53 - Saron Yitbarek:

John 实际上共同主持了三个播客：《Accidental Tech Podcast》、《Reconcilable Differences》和《Robot or Not?》。我们刚刚聊到了 Perl 在当今世界的地位。

00:22:06:

Perl 在当今世界排名如何？它仍然是最好的计算机语言吗？

00:22:10 - John Siracusa:

Perl 6 具有其他语言没有的、其他语言应该具有的东西，我一直在等待其他语言偷学它。例如语法是将常见任务概念化的一种好方法，而在我看来，使用语法来解决解析问题比使用现有的工具更令人愉快、更干净、更好。

00:22:31:

在 Perl 中，对象系统的许多部分看起来很琐碎而无关紧要，但我完全期待其他语言最终会采用它，就像许多语言最终采纳了 Perl 5 中的许多思想一样。因此我认为 Perl 6 在许多方面都是最好的。遗憾的是，很少有人有机会使用它。

00:22:52 - Saron Yitbarek:

你认为 Perl 6 社区的发展需要做些什么？想让人们更多地参与到 Perl 6 中，需要做些什么？

00:23:00 - John Siracusa:

这有点像 Perl 6 的故事，就像它一直在寻找一个真正奇妙的实现一样。这是第二系统问题的一部分.....^{second-system syndrome}第二系统综合症，我想他们是这样称呼.....

00:23:11 - Saron Yitbarek:

哦。

00:23:12 - John Siracusa:

Perl 6 的，人们希望修复世界上的所有问题。他们想要解决的问题之一是运行时环境。是什么在运行我们的代码？运行 Perl 5 和之前的 Perl 4 的东西是一个巨大的 C 程序，这是由具有独特编码风格的开发者编写的。还有大量的宏，它是一种相当难以理解的东西。

00:23:33:

Perl 6 的想法是让我们不要再那样做了，让我们不要制造大量的 C 代码。相反，让我们使用一个虚拟机，这在当时是一种时尚，有很多关于如何实现它的伟大想法。最终我们得到了几个中规中矩的虚拟机实现版本，有时这些还会相互竞争，但没有一个达到真正交付语言使用时需要的性能、稳定性和特性。

00:24:01 - Saron Yitbarek:

现如今 Perl 到底发生了什么？你对此有什么看法？

00:24:06 - John Siracusa:

Perl 5 绝对像是在走下坡路，因为与 Perl 5 同时代的所有其他语言都采纳了它的大部分最佳思想，并获得了更多的支持。也就是说，因为它在很长一段时间内都是王者，所以有很多 Perl 5 代码在运行一些大型的、重要的站点，人们需要维护和扩展这些代码。

00:24:29:

这需要很长时间才能消失。只要看看现今仍然存在的 COBOL，人们怎么还在雇佣人在 COBOL 上做维护吧？

00:24:35 - Saron Yitbarek:

嗯。是这样。

00:24:36 - John Siracusa:

你刚才问 Perl 是不是一门垂死的语言，我提到了 COBOL，这听起来并不乐观。Perl 6 本身会成为主流语言吗？看起来可能性不大。现在对其他语言有非常多的关注，如果 Perl 6 现在还没有得到开发者的关注，我不知道将会需要发生什么变化来让它流行起来。

00:24:54 - Saron Yitbarek:

如果你是这样想的，你对 Perl 有什么期望？你希望在 Perl 5 或 Perl 6 中看到什么，以及希望看到将来发生什么？

00:25:04 - John Siracusa:

我对 Perl 5 的希望是人们不要忽视它，因为尽管有其它更流行的语言，但今天许多公司仍然采用 Perl 5 做为解决问题的最佳方案。通常这些都是胶水类型语言的问题。如果你发现自己曾经编写过 shell 脚本，并且可能会说：“好吧，我不会用我的‘真正的编程语言’来做这件事。”不管是 Python，还是 Ruby，还是别的什么。但是 shell 脚本可以让我把一堆东西连接起来（胶水类型语言）。Perl 是完成这项工作的更好工具。编写正确的 Perl 脚本要比编写正确的 shell 脚本更容易。

00:25:40 - Saron Yitbarek:

我认为归根结底 Perl 可能不再是一个适合入门的语言，但对于经验更丰富的多语言开发人员来说，它是那个你永远不希望扔掉的工具箱中的小工具，而且特定的工具有时让你提升水平的工具。

00:25:58 - John Siracusa:

有时我为 Perl 6 感到难过和沮丧，认为它不会有任何进展，有时我想“好吧，这是个不错的小社区”。每个社区都不需要称霸世界，也不需要成为整个行业的主导语言。也许可以就这样一直走下去，就是，无限期地走下去。这就是开源，和编程语言的伟大之处。没人可以阻止你，你可以像以前一样继续开发 Perl 6。

00:26:27 - Saron Yitbarek:

John Siracusa 是一名程序员，也是三个科技播客的联合主持人。

00:26:34:

语言都是有生命周期的。当新的语言出现时它们能够精确地适应新的现实，像 Perl 这样的选择可能会占据更小的、更小众的领域，但这并不是一件坏事。我们的语言应该随着我们需求的变化而扩大或缩小它们的群体。在互联网开发的早期历史中，Perl 是一个至关重要的角色，它以各种方式与我们联系在一起，只要看一看它的历史就会发现它的存在。

00:27:11:

下次在《代码英雄》中，我们将讨论：是什么将一种语言变成了标准？以及在基于云的开发世界中，新的标准将如何出现？

00:27:26:

《代码英雄》是红帽的原创播客。如果你想深入了解 Perl 的故事，或者任何我们在第三季中探索的编程语言，请访问 redhat.com/commandlineheroes。我们的网站里有许多精彩内容等你去探索。

00:27:49:

我是 Saron Yitbarek。下期之前，编码不止。

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-3/diving-for-perl>

作者: [Red Hat](#) 选题: [bestony](#) 译者: [Mikedkmilk](#) 校对: [Northurland, wxy](#)

本文由 [LCRH](#) 原创编译, [Linux中国](#) 荣誉推出