

02 | 学习路径：爬过这六个陡坡，你就能对Linux了如指掌

2019-03-29 刘超

趣谈Linux操作系统

[进入课程 >](#)



讲述：刘超

时长 13:16 大小 12.15M



很多人觉得 Linux 操作系统刚开始学起来很难，主要是因为我们平时办公或者学习用的都是 Windows 系统，而 Windows 和 Linux 的使用模式是完全不一样的。

Windows 的基本使用模式是“**图形化界面 + 菜单**”。也就是说，无论我们做什么事情，首先要找一个图形化的界面。在这里面，“开始”菜单是统一的入口，无论是运行程序，还是做系统设置，你都能找到一个界面，界面上会有各种各样的输入框和菜单。我们只要挨个儿看过去，总能找到想操作的功能。实在不行，还有杀手锏，就是右键菜单，挨个儿一项一项看下去，最终也能实现想做的操作。

如果你刚刚上手 Linux，就会发现，情况完全不一样。你基本是这也找不着，那也找不着，觉得 Linux 十分难用，从而“从入门到放弃”。

Linux 上手难，学习曲线陡峭，所以它的学习过程更像一个**爬坡模式**。这些坡看起来都很陡，但是一旦爬上一阶，就会一马平川。你会惊叹 Linux 的设计之美，而 Linux 的灵活性也会使得你有 N 多种方法解决问题，从而事半功倍，你就会有一切尽在掌握的感觉。只可惜，大部分同学都停留在了山脚下。

那怎样才能掌握这项爬坡技能呢？我们首先需要明确，我们要爬哪些坡。

我总结了一下，在整个 Linux 的学习过程中，要爬的坡有六个，分别是：熟练使用 Linux 命令行、使用 Linux 进行程序设计、了解 Linux 内核机制、阅读 Linux 内核代码、实验定制 Linux 组件，以及最后落到生产实践上。以下是我为你准备的爬坡秘籍以及辅助的书单弹药。

第一个坡：抛弃旧的思维习惯，熟练使用 Linux 命令行

上手 Linux 的第一步，要先从 Windows 的思维习惯，切换到 Linux 的“**命令行 + 文件**”使用模式。

在 Linux 中，无论我们做什么事情，都会有相应的命令工具。虽然这些命令一般会在 bin 或者 sbin 目录下面，但是这些命令的数量太多了。如果你事先不知道该用哪个命令，很难通过枚举的方式找到。因此，在这样没有统一入口的情况下，就需要你对最基本的命令有所掌握。

一旦找到某个命令行工具，替代输入框的是各种各样的启动参数。这些参数怎么填，一般可以通过 -h 查看 help，挨个儿看过去，就能找到相应的配置项；还可以通过 man 命令，查看文档。无论是什么命令行工具，最终的配置一般会落到一个文件上，只要找到了那个文件，文件中会有注释，也可以挨个儿看下去，基本就知道如何配置了。

这个过程可能非常痛苦，在没有足够熟练地掌握命令行之之前，你会发现干个非常小的事情都需要搜索半天，读很多文档，即便如此还不一定能得到期望的结果。这个时候你一定不要气馁，坚持下去，继续看文档、查资料，慢慢你就会发现，大部分命令的行为模式都很像，你几乎不需要搜索就能完成大部分操作了。

恭喜你，这个时候你已经爬上第一个坡了。这个时候，你能看到一些很美丽的风景，例如一些很有技巧的命令 sed 和 awk、很神奇的正则表达式、灵活的管道和 grep、强大的 bash。你可以自动化地做一些事情了，例如处理一些数据，会比你使用 Excel 要又快又

准，关键是不用框框点点，在后台就能完成一系列操作。在处理数据的同时，你还可以干别的事情，半夜处理数据，第二天早上发个邮件报告，这都是 Excel 很难做到的事情。

不过，在这个专栏里，命令行并不是我们的重点，但是考虑到一些刚起步的同学，在第一部分我会简单介绍一些能够让你快速上手 Linux 的命令行。专栏每一模块的第一节，我都会有针对性地讲解这一模块的常用命令，足够你把 Linux 用起来。

如果你想全面学习 Linux 命令，推荐你阅读《**鸟哥的 Linux 私房菜**》。如果想再深入一点，推荐你阅读《**Linux 系统管理技术手册**》。这本砖头厚的书，可以说是 Linux 运维手边必备。

第二个坡：通过系统调用或者 glibc，学会自己进行程序设计

命令行工具也是程序，只不过是别人写的程序。从用别人写的程序，到自己能够写程序，通过程序来操作 Linux，这是第二个要爬的坡。

用代码操作 Linux，可以直接使用 Linux 系统调用，也可以使用 glibc 的库。

Linux 的系统调用非常多，而且每个函数都非常复杂，传入的参数、返回值、调用的方式等等都有很多讲究。这里面需要掌握很多 Linux 操作系统的原理，否则你会无法理解为什么应该这样调用。

刚开始学 Linux 程序设计的时候，你会发现它比命令行复杂得多。因为你的角色再次变化，这是为啥呢？我这么说，估计你就能理解了。

如果说使用命令行的人是吃馒头的，那写代码操作命令行的人就是做馒头的。看着简简单单的一个馒头，可能要经过 N 个工序才能蒸出来。同样，你会发现，你平时用的一个简单的命令行，却需要 N 个系统调用组合才能完成。其中每个系统调用都要进行深入地学习、读文档、做实验。

经过一段时间的学习，你啃下了这些东西，恭喜你，又爬上了一个坡。这时候，你已经很接近操作系统的原理了，你能看到另一番风景了。大学里学的那些理论，你再回去看，现在就会开始有感觉了。你本来不理解进程树，调用了 fork，就明白了；你本来不理解进程同步机制，调用了信号量，也明白了；你本来分不清楚网络应用层和传输层的分界线，调用了 socket，都明白了。

同样，专栏的第一模块，我会简单介绍一下 Linux 有哪些系统调用，每一模块的第一节，我还会讲解这一模块的常用系统调用，以及如何编程调用这些系统调用。这样可以使你对 Linux 程序设计入个门，但是这对于实战肯定是远远不够的。如果要进一步学习 Linux 程序设计，推荐你阅读《**UNIX 环境高级编程**》，这本书有代码，有介绍，有原理，非常实用。

第三个坡：了解 Linux 内核机制，反复研习重点突破

当你已经会使用代码操作 Linux 的时候，你已经很希望解开这层面纱，看看系统调用背后到底做了什么。

这个时候，你的角色要再次面临变化，**就像你蒸馒头时间长了，发现要蒸出更好吃的馒头，就必须要对面粉有所研究**。怎么研究呢？当然你可以去面粉厂看人家的加工过程，但是面粉厂的流水线也很复杂，很多和你蒸馒头没有直接关系，直接去看容易蒙圈，所以这时候你最好先研究一下，面粉制造工艺与馒头口味的关系。

对于 Linux 也是一样的，进一步了解内核的原理，有助于你更好地使用命令行和进行程序设计，能让你的运维和开发水平上升一个层次，但是我不建议你直接看代码，因为 Linux 代码量太大，很容易迷失，找不到头绪。最好的办法是，先了解一下 Linux 内核机制，知道基本的原理和流程就可以了。

一旦学起来的时候，你会发现，Linux 内核机制也非常复杂，而且其中相互关联。比如说，进程运行要分配内存，内存映射涉及文件的关联，文件的读写需要经过块设备，从文件中加载代码才能运行起来进程。这些知识点要反复对照，才能理清。

但是一旦爬上这个坡，你会发现 Linux 这个复杂的系统开始透明起来。无论你是运维，还是开发，你都能大概知道背后发生的事情，并在出现异常的情况时，比较准确地定位到问题所在。

Linux 内核机制是我们这个专栏重点要讲述的部分，我会基于最新 4.x 的内核进行讲解，当然我也意识到了内核机制的复杂性，所以我选择通过故事性和图形化的方式，帮助你了解并记住这些机制。

这块内容的辅助学习，我推荐一本《**深入理解 LINUX 内核**》。这本书言简意赅地讲述了主要的内核机制。看完这本书，你会对 Linux 内核有总体的了解。不过这本书的内核版本有点老，不过对于了解原理来讲，没有任何问题。

第四坡：阅读 Linux 内核代码，聚焦核心逻辑和场景

在了解内核机制的时候，你肯定会遇到困惑的地方，因为理论的描述和提炼虽然能够让你更容易看清全貌，但是容易让你忽略细节。

我在看内核原理的书的时候也遇到过这种问题，有的地方实在是难以理解，或者不同的书说的不一样，这时候该怎么办呢？其实很好办，Linux 是开源的呀，我们可以看代码呀，代码是精准的。哪里有问题，找到那段代码看一看，很多问题就有方法了。

另外，当你在工作中需要重点研究某方面技术的时候，如果涉及内核，这个时候仅仅了解原理已经不够了，你需要看这部分的代码。

但是开源软件代码纷繁复杂，一开始看肯定晕，找不着北。这里有一个诀窍，就是**一开始阅读代码不要纠结一城一池的得失，不要每一行都一定要搞清楚它是干嘛的，而要聚焦于核心逻辑和使用场景。**

一旦爬上这个坡，对于操作系统的原理，你应该就掌握得比较清楚了。就像蒸馒头的人已经将面粉加工流程烂熟于心。这个时候，你就可以有针对性地去课题，把所学和你现在做的东西结合起来重点突破。例如你是研究虚拟化的，就重点看 KVM 的部分；如果你是研究网络的，就重点看内核协议栈的部分。

在专栏里，我在讲述 Linux 原理的同时，也会根据场景和主要流程来分析部分代码，例如创建进程、分配内存、打开文件、读写文件、收发网络包等等。考虑到大量代码粘贴会让你看起来比较费劲，也会占用大量篇幅，因而我采取只叙述主要流程，只放必要的代码，大部分的逻辑和相互关系，尽量通过图的方式展现出来，给你讲解。

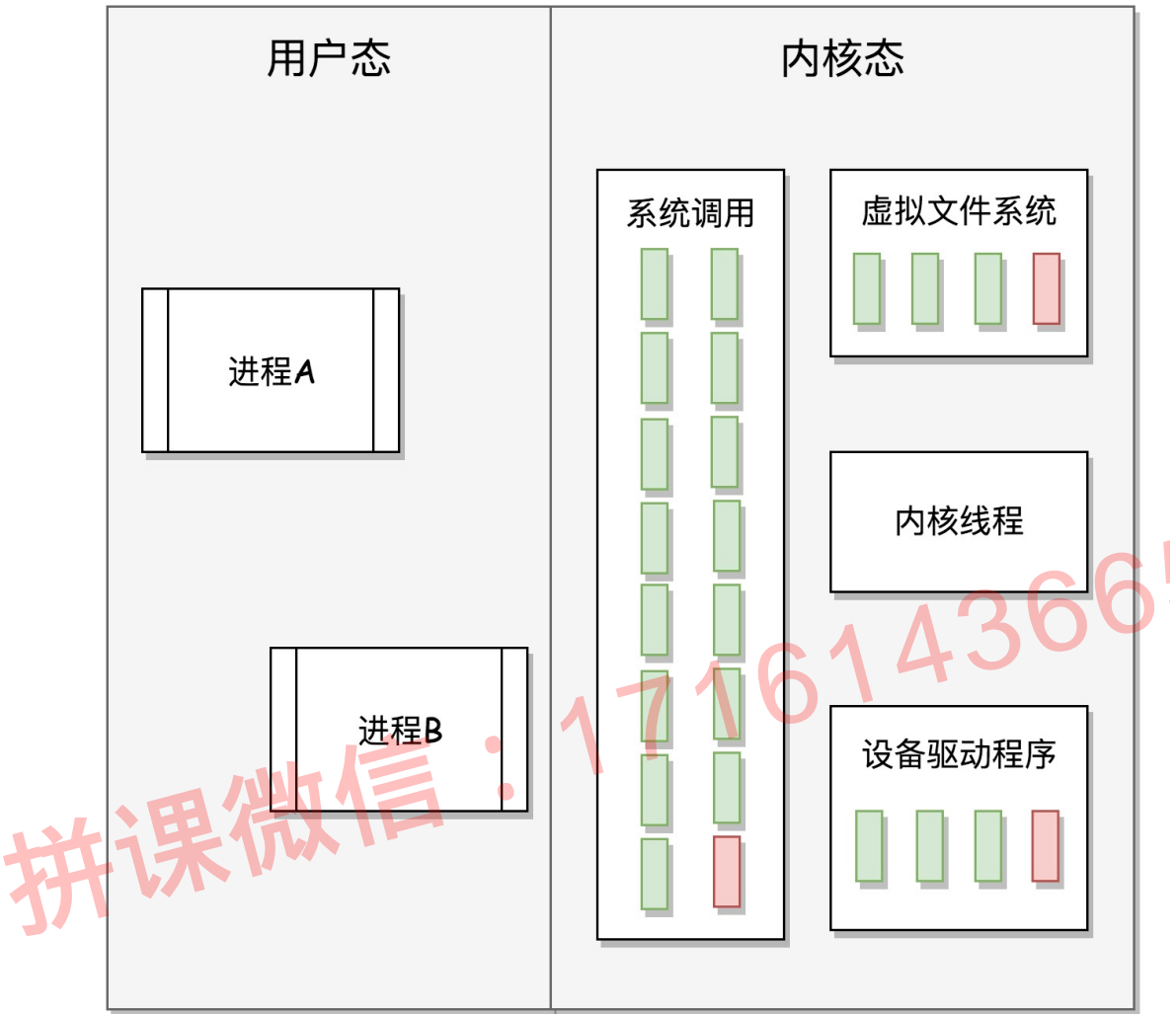
这里也推荐一本书，《**LINUX 内核源代码情景分析**》。这本书最大的优点是结合场景进行分析，看得见、摸得着，非常直观，唯一的缺点还是内核版本比较老。

第五坡：实验定制化 Linux 组件，已经没人能阻挡你成为内核开发工程师了

纸上得来终觉浅，绝知此事要躬行。从只看内核代码，到上手修改内核代码，这又是一个很大的坎。这相当于蒸馒头的人为了定制口味，要开始修改面粉生产流程了。

因为 Linux 有源代码，很多地方可以参考现有的实现，定制化自己的模块。例如，你可以自己实现一个设备驱动程序，实现一个自己的系统调用，或者实现一个自己的文件系统等

等。



这个难度比较大，涉及的细节比较多，上一个阶段，我的建议是不计较一城一地的得失，不需要每个细节都搞清楚，这一个阶段要求就更高了。一旦代码有一个细微的 bug，都有可能导致实验失败。

专栏最后一个部分，我专门设计了两个实验，帮你度过这个坎。只要跟着我的步伐进行学习，接下来，就没人能够阻挡你成为一名内核开发工程师了。

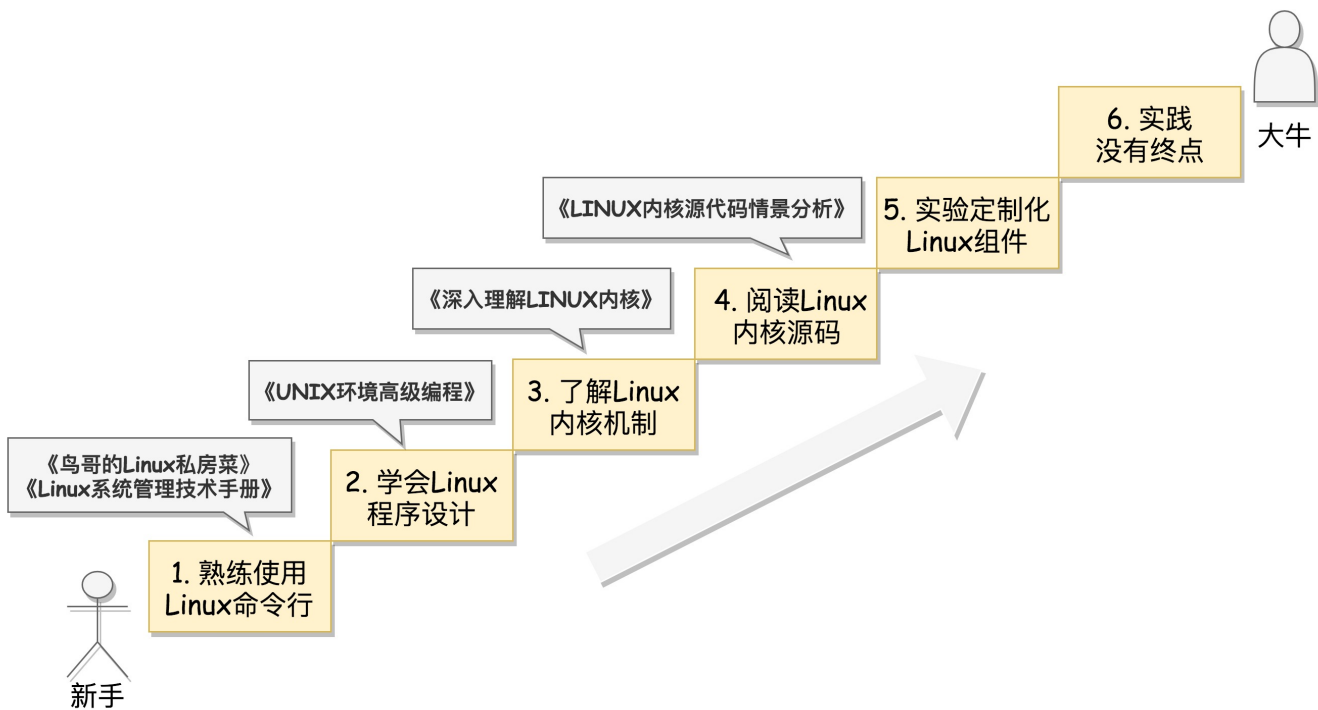
最后一坡：面向真实场景的开发，实践没有终点

说了这么多，我们都只是走出了万里长征第一步。我始终坚信，真正的高手都是在实战中摸爬滚打练出来的。

如果你是运维，仅仅熟悉上面基本的操作是不够的，生产环境会有大量的不可控因素，尤其是集群规模大的更是如此，大量的运维经验是实战来的，不能光靠读书。如果你是开发，对内核进行少量修改容易，但是一旦面临真实的场景，需要考虑各种因素，并发与并行，锁与保护，扩展性和兼容性，都需要真实项目才能练出来。

总结时刻

今天，我把爬坡的过程，分解成了六个阶段，并给你分享了我的私家爬坡宝典。你都记住了吗？我把今天的内容总结成了下面这张图。建议你牢牢记住这张图，在接下来的四个月中，按照这个路径稳步前进，攻克 Linux 操作系统。



Linux 操作系统爬坡路线图

课堂练习

你可以结合第一节的测试结果，并根据我今天讲的爬坡方法，思考一下，在接下来的四个月里，你准备怎么学习这个专栏。

欢迎在留言区写下你的**爬坡计划**，也欢迎你今天的文章分享给你的朋友，和他一起学习、进步。

编辑乱入：超哥推荐的图书，部分已上架极客时间商城，点击下方图片，即可购买。和专栏一起配合使用，学习效果会更好哦！

上新享优惠

 极客时间商城

Linux 学习书单



限时特惠 3月29日-4月5日

趣谈 Linux 操作系统

像故事一样的操作系统入门课

刘超

网易杭州研究院

云计算技术部首席架构师



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 01 | 入学测验：你究竟对Linux操作系统了解多少？

下一篇 03 | 你可以把Linux内核当成一家软件外包公司的老板

精选留言 (119)

写留言



rocedu

2019-03-29

👍 90

别出心裁的Linux命令学习法<https://www.cnblogs.com/rocedu/p/4902411.html>, 我这篇博客可以让你尽快过第一关。



sotey

2019-03-29

👍 62

学习计划：

我是做运维的

1、第一件事情当然是安装Linux操作系统了，我用虚拟机安装了CentOS7，第一遍升级到ml居然是5.0内核，内核还是升级到lt4.x， - -~！

2、买书!!! Unix环境编程、深入理解Linux内核、Linux源代码情景分析, 又多了三本...
展开 ▾



峰

2019-03-29

👍 43

感觉4个月后, 在阶段3就ok了。

展开 ▾



rocedu

2019-03-29

👍 25

第二个坡, 读了《UNIX 环境高级编程》当然可以一览众山小, 我的学习和教学经验更推荐《Unix/Linux编程实践教程》, 当年我是先读《UNIX 环境高级编程》, 后读的《Unix/Linux编程实践教程》, 要是反过来, 学习起来会更好。我的博客[别出心裁的Linux系统调用学习法](<http://www.cnblogs.com/rocedu/p/6016880.html>)可以让你快速入门Linux系统调用, 掌握学习方法。

展开 ▾



rocedu

2019-03-29

👍 21

爬第三个坡, 我推荐看一下《庖丁解牛Linux内核分析》(<https://book.douban.com/subject/30350365/>), 跟着MOOC一起自己动手搭建Linux内核, 相当于自己动手做面粉厂。

展开 ▾



苏柏

2019-03-29

👍 10

作为一个Linux 小白, 同时作为一名java开发, 定专栏时给自己定的目标就是能够到达第三阶段

首先能够熟练使用Linux命令行, 能做正常的系统运维操作。结合专栏和鸟哥的私房菜(已经买了3, 4年没看/捂脸)

第二、能够进行基本的linux程序设计, 实现代码调用linux操作, 主要跟专栏进行实践...

展开 ▾



William

2019-03-29

👍 8

学习路径

Step1: 熟悉Linux命令行

基础: --help、man...

展开 ∨



hua168

2019-03-29

👍 8

老师，这些书能不能加一个链接呢，比如豆瓣的链接地址，有些名有重名的，作者却不同。



久恰似一种...

2019-04-06

👍 5

做为一名Java开发，认为还是要靠实践出真知，主要是跟着刘老师的专栏进行学习

第一个坡：阅读《鸟哥的Linux私房菜》，实践Linux常用命令

第二个坡：阅读《《UNIX环境高级编程》，学习如何进行Linux程序设计

其它的还没有想到这么远，一个个坡爬吧，不要一直停留在山脚就可以，希望自己每爬一个坡，都能体验到不同的风景。

展开 ∨



微秒

2019-03-29

👍 4

我觉得只要让我了解操作系统的原理，能解决面试和用linux做上层的程序设计就够了。



江南皮革厂...

2019-03-29

👍 3

老师，我工作是负责搞自动化运维的，目前只会python编程和shell编程，针对自动化运维研究内核的话我应该从哪一块提高，是不是要把C再滚一遍呀。_-

展开 ∨



michael

2019-03-29

👍 3

鸟哥的私房菜看了一小半了，通俗易懂，强烈推荐啊*^_^*

展开 ▾



小牛人

2019-03-29

👍 3

这节课让我回想起了嵌入式linux的学习历程。

展开 ▾



L.

2019-05-21

👍 2

鸟哥的文档一开始读起来对部分名词不适应，后来越读越觉得台湾腔很可爱

展开 ▾



褚瑞

2019-04-01

👍 2

在torvalds/linux这个repo里搜system call，有3000个结果。不知道这些代码都是怎么组织在一起的。



lllllll

2019-03-31

👍 2

老师 能不能教一下安装Linux虚拟机

展开 ▾



岳文

2019-03-31

👍 2

说实话 这个专栏68元真的是捡到宝了。谢谢老师。好好学习，天天向上！

展开 ▾



胡鹏

2019-03-30

👍 2

现在阶段1, 定个小目标 阶段3

展开 ▾



return

2019-03-29

👍 2

目标宏伟，现在转管理了，但是依旧觉得技术原理非常重要，加油吧

展开 ▾



谷鱼

2019-04-10

👍 1

Linux一直很感兴趣，大学里也学了很多操作系统知识，工作中一直在熟悉各种命令，也熟悉了一部分系统调用，但是总会在一些实际中的环境部署，互相依赖中，失去了前进的能力，总是会放弃一部分修改，重新采用依赖完整的方案。四个月，目前时间有点不充足，希望自己能够得到一个清晰的学习路线，更多的知识可以在需要的时候，知道如何继续深入学习

展开 ▾