

学习攻略（一）：学好操作系统，需要掌握哪些前置知识？

2019-04-26 刘超

趣谈Linux操作系统

[进入课程 >](#)



讲述：刘超

时长 12:12 大小 11.19M



你好，我是刘超。

专栏已经更新一段时间了，随着涉及的技术越来越深，很多同学觉得学习的难度也越来越大了，并在留言区提出了一些问题。有些问题，是我之前学习也碰到过的或者我思考过的；也有些问题是我没想到的，但是这些问题归根结底其实就是一个，如何更好地去学习操作系统，学习这门课。所以，今天我就给你讲讲，我认为学习这个专栏的一些好的方法和正确的态度。

1. 态度：不要等一切都准备好了再前行

很多人问到，“学习这个专栏需要什么基础”，说实话，我在设计这个专栏的时候，和编辑也专门讨论过这个问题。为此，我曾经专门研究过大学的课程设计。大学的操作系统前置课

程有程序设计基础、数据结构与算法、计算机组成与体系结构、编译原理等。这几门前置课程，几乎每一门都非常“硬核”，学习难度和大多数人的掌握程度往往是成反比的。

你可能要问了，我是不是要把这些前置课程都搞定了才能学操作系统呢？当然不是。这些课程它们本身就很难，除此之外，它们也有前置课程。如果这样，你就会发现你永远开始不了。

如果把一切都准备好了再前行，那我们永远迈不出第一步。

你可能会说，没有这些基础知识，我每看一篇文章，知识就铺天盖地席卷过来，仿佛每一个知识点都准确地打在了自己的盲点上，这该怎么办呢？

我非常能理解这种心情，而我想说的是，你不需要对这几门前置课程有非常深入的理解，你其实就可以学好这门课。但如果你是编程零基础，学习起来的确比较困难。

毕竟我们这门课相对于大学的操作系统课，更加面向真实的操作系统分析，因而更深入一步，加上操作系统这门课本身的技术深度和知识点琐碎程度，就是很高的，所以感觉就更加困难一些。

这有点像咱们平时玩游戏，作为主人公，你既不能没有任何装备和法术就开始，也不能等练级练到满，将整个地图都了解清楚才开始。你只要有一个初始装备，例如一把木剑、一套基本的法术就可以开始了。先上路，之后在探险的路上，慢慢摸清地图，积累装备，训练法术，最终 PK 终极 Boss。

这么一来，想要学好操作系统，我们要解决的其实就是这样两个问题：第一是初始装备问题，也就是说前置知识应该掌握到什么程度；第二就是练级攻略问题，也就是掌握这些操作系统知识的方法。

2. 资料：C 语言 + 数据结构和算法 + 编译原理 + 计算机组成

学习操作系统需要对 C 语言基础知识、数据结构和算法、编译原理以及计算机组成原理有一些了解，这样学起来就会更轻松。

因为我个人是从 C 语言转向 Java 的，我觉得 Java 和 C 语言的基本语法没有太大的差别。

仅仅就学习咱们这个操作系统专栏来讲，其实你只要有计算机二级的那些 C 语言基础知识基本就够了。如果你要严肃地学好 C 语言，几乎每个前辈都会建议你从《C 程序设计语言》这本书学起。这是一本很薄很薄的书。如果你有编程基础，但是之前没有学过 C 语言，稍微翻一翻它，了解一下 C 语言的基本编程方法也就足够用了。

这本书里，第 1 章到第 3 章涉及类型、运算符、表达式、控制流，这些和其他语言区别不大，你应该很容易理解。你重点看第 4 章函数与程序结构、第 5 章指针与数组、第 6 章结构，这样基本就可以了。

对于数据结构和算法，业界有大量砖头一样厚的书籍。如果从头学起，你可能会比较崩溃。其实你只要了解表、栈、队列、树，这些基本的内容就够用了。现在学习资料这么多，想学会这些内容不是难事，关键看你是不是用心，是不是真心想要学习。既然是基于 C 语言的数据结构和算法，我这里给你推荐一本书《数据结构与算法分析：C 语言描述》。在这本书里，你重点看第 3 章表、栈和队列和第 4 章树，基本就可以了。

编译原理和计算机组成，这些都是操作系统的基础。在咱们的专栏里面，我也有简单的阐述，你只要掌握我平时说的那些，基本上足够了。计算机组成与系统原理的书，相对来说都比较复杂。

一般的汇编语言的书都会简单介绍 x86 的机制，以及简单的汇编语言。例如，《汇编语言：基于 x86 处理器》这本书里，你可以重点看第 2 章 x86 处理器架构和第 3 章汇编语言基础，掌握这些基本就可以了。

除此之外，极客时间出品的《数据结构与算法之美》《深入浅出计算机组成原理》也是非常优秀的学习资料。所有基础知识，本是一家。如果有精力，推荐你认真学习这两个专栏，对我们这门课会非常有帮助。

3. 方法：三遍学习法 + PPT 笔记法

读书有一个理念，是先把书读厚，再把书读薄，这往往是对有一定基础的同学的学习方法，如果你原来没有研究过操作系统，或者没有基于 Linux 的 API 进行程序设计，要想看懂每一行代码的确是非常难的事情。所以学习这门课程，可以采取“先读薄，再读厚，再读薄”这样的三遍学习法。

所谓的先读薄，就像我在开篇词中说的，了解操作系统的原理，一开始不要纠结一城一地的得失。不要死抠每一个代码细节，而重点了解主要的流程和原理。

学习是循序渐进的过程，不要妄想一遍就掌握所有。我自己在学习的时候，经常是秉持着这么一个方法：刚开始接触一个新东西，多去看别人的学习方法，从别人的方法中摸索中“主要矛盾”和“次要矛盾”，每一遍学习都提纲挈领地抓当前的主要矛盾，然后一遍一遍去学这个内容，当然每一遍的主要矛盾可能都不太一样。

比如在**第一遍学习这门课**的时候，你没必要把文稿中的代码细节当成学习目标，只把它看作一个对操作系统原理的佐证就行了。我讲操作系统这个地方是这样做的，你看代码，大概就能看出来，它真的就是这样做的。这样一方面先形成一个大致的印象，另一方面先在心里对原理有一个具象的理解。至于代码中每一行的意思，这一遍不要纠结。

另外，Linux 代码中有很多注释，如果实在看不懂代码，那你就阅读注释就行了，然后按照注释中的意思，浏览一下代码大概的意思就可以了。尤其是对于汇编语言，更是这样。比如说压栈操作，你看到 push，就理解是正在压栈，然后后面有注释说，压入了哪个栈。这时候，你暂时没必要纠结，为什么这行命令压入的是这个栈。

当你可以按照我每节画的图，把原理理解得差不多的时候，第一遍基本就完成了。

第一遍相当于咱们打游戏，到了一个新的场景，先到处走走看看情况，不纠结和妖怪的对战。

接下来的第二遍，就相当于我们需要完成在游戏里面的任务，碰到怪就打怪，碰到迷宫走迷宫了。这也是把书读厚的方法，就是见山开路，遇水搭桥。

这一遍并不需要你把所有不懂的知识都找到一本砖头厚的书全部看完，这样你会迷失在庞大的知识体系里面，丧失方向，偏离目标，你会忘了为什么要看这本书，甚至会疑惑通过这本书到底能不能帮你理解操作系统的某个知识。

所以把书读厚，还是要以操作系统的知识体系为主线，遇到不懂的知识点的时候，通过搜索引擎，看我给出的参考资料里面的某个章节，读某个博客等方式，达到了解这个知识点就可以了。一旦了解到能够帮助你理解操作系统的程度，就不需要再深入下去，马上回到主线，攻克下一个知识点。

这里可以分享一下我的学习大量陌生知识方法，就是记笔记，我一般是用 PPT 来记。当读了每篇文章的一个小节的时候，甚至可以更细到几个自然段，就像咱们学语文的时候划分自然段一样，你可以为这一段知识做一页 PPT，PPT 的标题就是这一段的段落大意，能够总

结出段落大意。其实你已经知道这一段主要是做什么的，只不过很多细节的知识点还不清楚，这样将来通过 标题浏览 PPT 的时候，就很容易把握整篇文章的总体思路。

接下来在这页 PPT 的正文中，将所有的知识点都列出来，其中不懂的部分编个号。例如，知识点 17，然后接下来的几页 PPT，我们就以知识点 17 为标题，开始查所有能够帮助你了解这个知识点的文章和书籍，将内容贴进来，直到你发现你已经懂了这个知识点，当然知识点可能还要再套知识点，接着编号就行。当从这个知识点的 PPT 返回的时候，你知道你理解了知识点 17，接下来应该攻克知识点 18 了。当所有的知识点都趟过去，你会发现这是一个特别长的 PPT，没有关系，你把所有的以知识点为标题的 PPT 页都放在最后做附页，再次看整篇文章，就会有一种一马平川的感觉。如果暂时忘了，可以到附页中去查看自己记的笔记，很容易就可以回忆起来。

这个特别长的 PPT 就是起到了把书读厚的作用。如果你有兴趣搜索我写的文章《别以为真懂 Openstack：虚拟机创建的 50 个步骤和 100 个知识点》，看看我是怎样趟过这 100 个 OpenStack 相关知识点的。

第三遍再读薄，是我希望能够达到这样一个效果。经过前两遍的学习，现在你对知识的细节和整体的脉络都应该有了一定了解。因此这第三遍无需面面俱到，你需要把这些知识真正变成你自己的东西。

专栏每篇文章后面我都会用一张图对这篇文章做一个总结。这些总结图代表了我对知识的梳理和理解。你可以先试着看着我的图来复习，之后，你可以自己尝试去总结这些内容。把知识真正装进你自己的脑袋里。等整个专栏全部结束的时候，你只需要对着这些图，将整个操作系统的知识串下来，基本就融会贯通了。

4. 做好练习，用好音频

你应该有感觉，操作系统的知识量非常大。但凡想要讲的面面俱到，都需要一本砖头厚的书籍才行。专栏不可能和书籍一样，我主要是基于 Linux 给你讲，我是怎么理解操作系统的。因而，我讲的肯定是最核心的内容，很多旁支的东西需要你自己去预习和复习。

每节课我留的练习题往往都是对当节课程的补充，也是对后面章节要讲知识的准备，所以你一定要重视课后题。每节课后都认真去做一做题，思考题目背后的知识点。

另外，音频这个形式非常重要。文稿里是最核心的内容，很多对于代码的分析放进去会非常繁琐，但是这块又非常重要，于是在音频讲解里，我会对代码部分进行更加详尽的讲述。如

果你遇到，仅仅看文字不能理解的部分，尝试听听语音，或许会有不一样的收获。

最后，我想借用陈皓的一句话，“学习不是为了找到答案，而是为了找到方法”。哪怕我已经尽力为你考虑周全，但是没有你自己主动学习，去钻研，去下苦功寻找适合自己的方法，成功是不可能自己找上门来的。希望你可以在为自己而努力。

好了，今天我想分享的就是这些。不知道我今天的分享哪个地方对你最有用呢？你在学专栏的过程中还遇到过什么问题呢？

欢迎留言和我分享你的疑惑和见解。你可以收藏今天的文章，反复研读。你也可以把今天文章分享给你的朋友，和他一起学习操作系统。

 极客时间

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



拼课微信：1716143661

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 13 | 进程数据结构（中）：项目多了就需要项目管理系统

下一篇 14 | 进程数据结构（下）：项目多了就需要项目管理系统



顾晓峰...

2019-04-26

👍 23

老师真的体谅我们啊，感觉有点跟不上的时候，给我们指引了一条明路，教我们好的学习方法，而不是一味传授知识，结个大大的赞！



张细敏

2019-04-26

👍 9

对于老师说的三遍学习发法，感觉还是很有用。我发现国外的人写书也有这个特点，一般第一章是对整本书的核心知识做个讲解。之后章节是对第一章的涉及到的进行细致讲解。这种方法能大量使用在书籍编写中，足以说明这种方法还是有作用的。今天的文章收获到了接下来如何做，如何学习



xavier

2019-04-27

👍 8

如果把一切都准备好了再前行，那我们永远迈不出第一步。值得自己好好反思。



Noah

2019-04-26

👍 2

这个三遍学习法不错，可以试试，不过可以使用思维导图替换 PPT
展开 ▾



Jimmy

2019-04-26

👍 2

老师，需要下载Linux内核代码来对照学习吗？买过内核分析方面的书，但没有坚持看下来。



why

2019-04-26

👍 1

李笑来老师写的自学是门手艺也很有借鉴意义, <https://github.com/selfteaching/the-craft-of-selfteaching>. 还是要肯花时间, 学习不能太着急.



毒液-致命...

2019-05-24

👍

老师说的非常好，我自己的学习策略，基本是看完一遍，回忆一遍，实际用的时候，在搜索一遍

作者回复: 赞



w3c10kt

2019-05-24



没有基础，好多前置课程呀，已经掉队了，加油

展开 ∨

作者回复: 加油，盘他



幻月剑

2019-05-03



嘿嘿，里面说的两个专栏我都买了，数据结构与算法看完了第一遍，找个合适的时间开始刷第二遍，计算机组成原理的专栏正在跟。

这篇文章讲的方法真的很好很好，谢谢！

展开 ∨



nora

2019-05-03



还有一个问题是，同时开了sql的

展开 ∨



nora

2019-05-03



个人认为这个方法简化了陌生知识带了的难度，一步一步从框架到细节，引导自己熟悉的领域更加深入，更加节省注意力，更容易长久的坚持。



海里的鱼◆...

2019-04-28



这句话感同身受！“学习不是为了找到答案，而是为了找到方法”



williamcai

2019-04-28



师傅领进门，修行看个人，每个人都会找到适合自己的方法

展开 ∨



Amanda

2019-04-28



老师这章来得刚刚好

展开 ∨



朱小豪

2019-04-27



是的开始跟不上了，还得回头捡起来

展开 ∨



Zj_scute

2019-04-27



老师这个三遍法真的很有用，现在每一天我都有在看老师的趣谈网络协议，以前老师讲TCP后面的我是完全看不懂，看了几遍以后发现真的能看懂了，也能理解其中的原理了。

展开 ∨



Dispy

2019-04-26



这篇文章看完了，脸有点烫，工作之后长期以来保持的学习态度和方法是不对的，真的是每次学个东西，遇到了不懂得知识点都要去深究，容易跑偏，还难以带回，很容易本末倒置。老想着一遍学会，结果是什么都学不到，第一遍看 or 听明白了，就觉得真的会了。谢谢老师，我觉得首先要把自己的学习态度纠正一下，缺少对知识体系的一种特殊的敬畏之感。

展开 ∨



火焰神

2019-04-26



听了老师的课，我想起了 同步 异步 阻塞 非阻塞，请老师在合适的章节，从最底层的原理讲下，一解我多年的疑惑，网上的文章说的太浅了。谢谢！



kdb_reboot

2019-04-26



老师的这篇软文很给力 解决了我心中的困惑：不要等全都准备好再开始
展开 ∨