程序员练级攻略（2018）：系统知识

2018-06-19 陈皓

程序员练级攻略（2018）：系统知识
朗读人：柴巍　11′27″ | 5.24M

进入专业的编程领域，学习系统知识是非常关键的一部分。

首先推荐的是翻译版图书《深入理解计算机系统》，原书名为《Computer Systems A Programmer's Perspective》。不过，这本书叫做《程序员所需要了解的计算机知识》更为合适。

本书的最大优点是为程序员描述计算机系统的实现细节，帮助其在大脑中构造一个层次型的计算机系统。从最底层的数据在内存中的表示到流水线指令的构成，到虚拟存储器，到编译系统，到动态加载库，到最后的用户态应用。通过掌握程序是如何映射到系统上，以及程序是如何执行的，你能够更好地理解程序的行为为什么是这样的，以及效率低下是如何造成的。

再强调一下，这本书是程序员必读的一本书！

然后就是美国计算机科学家 理查德·史蒂文斯（Richard Stevens） 的三套巨经典无比的书。（理查德·史蒂文斯于 1999 年 9 月 1 日离世，终年 48 岁。死因不详，有人说是滑雪意外，有

人说是攀岩意外，有人说是滑翔机意外。总之，家人没有透露。大师的 个人主页 今天还可以访问。）

- 《Unix 高级环境编程》。
- 《Unix 网络编程》 第 1 卷 套接口 API 、第 2 卷 进程间通信 。
- 《TCP/IP 详解 卷 I 协议》。

这几书的地位我就不多说了，你可以自己看相关的书评。但是，这三本书可能都不容易读，一方面是比较厚，另一方面是知识的密度太大了，所以，读起来有点枯燥和乏味。但是，这没办法，你得忍住。

这里要重点说一下《TCP/IP 详解》这本书，是一本很奇怪的书。这本书迄今至少被 近五百篇学术论文引用过 。这本写给工程师看的书居然被各种学院派的论文来引用，也是很神奇的一件事了。而且，虽然理查德·史蒂文斯不是 TCP 的发明人，但是这本书中把这个协议深入浅出地讲出来，还画了几百张时序图，也是令人叹为观止了。

如果你觉得上面这几本经典书比较难啃，你可以试试下面这些通俗易懂的（当然，如果读得懂上面那三本的，下面的这些也就不需要读了）。

- 《Linux C 编程一站式学习》。
- 《TCP/IP 网络编程》。
- 《图解 TCP/IP》，这本书其实并不是只讲了 TCP/IP，应该是叫《计算机网络》才对，主要是给想快速入门的人看的。
- 《The TCP/IP Guide》，这本书在豆瓣上的评分 9.2，这里给的链接是这本书的 HTML 英文免费版的，里面的图画得很精彩。

另外，学习网络协议不单只是看书，你最好用个抓包工具看看这些网络包是什么样的。所以，这里推荐一本书《Wireshark 数据包分析实战》。在这本书中，作者结合一些简单易懂的实际网络案例，图文并茂地演示使用 Wireshark 进行数据包分析的技术方法，可以让我们更好地了解和学习网络协议。当然，也拥有了一定的黑客的技能。

看完《Unix 高级环境编程》后，你可以趁热打铁看看《Linux/Unix 系统编程手册》或是罗伯特·拉姆（Robert Love）的 Linux System Programming 英文电子版 。其中文翻译版Linux 系统编程 也值得一读，虽然和《Unix 高级环境编程》很像，不过其主要突出的是 Linux 的一些关键技术和相关的系统调用。

关于 TCP 的东西，你还可以看看下面这一系列的文章。

- Let's code a TCP/IP stack, 1: Ethernet & ARP
- Let's code a TCP/IP stack, 2: IPv4 & ICMPv4
- Let's code a TCP/IP stack, 3: TCP Basics & Handshake

- Let's code a TCP/IP stack, 4: TCP Data Flow & Socket API
- Let's code a TCP/IP stack, 5: TCP Retransmission

对于系统知识，我认为主要有以下一些学习要点。

- 用这些系统知识操作一下文件系统，实现一个可以拷贝目录树的小程序。
- 用 fork / wait / waitpid 写一个多进程的程序，用 pthread 写一个多线程带同步或互斥的程序。比如，多进程购票的程序。
- 用 signal / kill / raise / alarm / pause / sigprocmask 实现一个多进程间的信号量通信的程序。
- 学会使用 gcc 和 gdb 来编程和调试程序（参看我的《用 gdb 调试程序》一、二、三、四、五、六、七）。
- 学会使用 makefile 来编译程序（参看我的《跟我一起写 makefile》一、二、三、四、五、六、七、八、九、十、十一、十二、十三、十四）。
- Socket 的进程间通信。用 C 语言写一个 1 对 1 的聊天小程序，或是一个简单的 HTTP 服务器。

C10K 问题

然后，当你读完《Unix 网络编程》后，千万要去读一下 "C10K Problem（中文翻译版）"。提出这个问题的人叫丹·凯格尔（Dan Kegel），目前工作在美国 Google 公司。

他从 1978 年起开始接触计算机编程，是 Winetricks 的作者，也是 Wine 1.0 的管理员，同时也是 Crosstool（ 一个让 gcc/glibc 编译器更易用的工具套件）的作者。还是 Java JSR 51 规范的提交者并参与编写了 Java 平台的 NIO 和文件锁，同时参与了 RFC 5128 标准中有关 NAT 穿越（P2P 打洞）技术的描述和定义。

C10K 问题本质上是操作系统处理大并发请求的问题。对于 Web 时代的操作系统而言，对于客户端过来的大量的并发请求，需要创建相应的服务进程或线程。这些进程或线程多了，导致数据拷贝频繁（缓存 I/O、内核将数据拷贝到用户进程空间、阻塞），进程 / 线程上下文切换消耗大，从而导致资源被耗尽而崩溃。这就是 C10K 问题的本质。

了解这个问题，并了解操作系统是如何通过多路复用的技术来解决这个问题的，有助于你了解各种 I/O 和异步模型，这对于你未来的编程和架构能力是相当重要的。

另外，现在，整个世界都在解决 C10M 问题，推荐看看 The Secret To 10 Million Concurrent Connections -The Kernel Is The Problem, Not The Solution 一文。

实践项目

我们已经学习完了编程语言、理论学科和系统知识三部分内容，下面就来做几个实践项目，小试牛刀一下。实现语言可以用 C、C++ 或 Java。

实现一个 telnet 版本的聊天服务器，主要有以下需求。

- 每个客户端可以用使用`telnet ip:port`的方式连接到服务器上。
- 新连接需要用用户名和密码登录，如果没有，则需要注册一个。
- 然后可以选择一个聊天室加入聊天。
- 管理员有权创建或删除聊天室，普通人员只有加入、退出、查询聊天室的权力。
- 聊天室需要有人数限制，每个人发出来的话，其它所有的人都要能看得到。

实现一个简单的 HTTP 服务器，主要有以下需求。

- 解释浏览器传来的 HTTP 协议，只需要处理 URL path。
- 然后把所代理的目录列出来。
- 在浏览器上可以浏览目录里的文件和下级目录。
- 如果点击文件，则把文件打开传给浏览器（浏览器能够自动显示图片、PDF，或 HTML、CSS、JavaScript 以及文本文件）。
- 如果点击子目录，则进入到子目录中，并把子目录中的文件列出来。

实现一个生产者 / 消费者消息队列服务，主要有以下需求。

- 消息队列采用一个 Ring-buffer 的数据结构。
- 可以有多个 topic 供生产者写入消息及消费者取出消息。
- 需要支持多个生产者并发写。
- 需要支持多个消费者消费消息（只要有一个消费者成功处理消息就可以删除消息）。
- 消息队列要做到不丢数据（要把消息持久化下来）。
- 能做到性能很高。

小结

到今天，我们已经学习完了专业编程方面最为重要的三部分内容：编程语言、理论学科和系统知识，我们针对这些内容做个小结。如果想看完我推荐的那些书和知识，并能理解和掌握，我估计怎么也得需要 4-5 年的时间。嗯，是的，就是一个计算机科学系科班出身的程序员需要学习的一些东西。这其中，最重要的是下面这几点。

编程语言。以工业级的 C、C++、Java 这三门语言为主，这三门语言才是真正算得上工业级的编程语言，因为有工业级的标准化组织在控制着这几门语言，而且也有工业级的企业应用。尤其是 Java，还衍生出了大量的企业级架构上的开源生态。你至少需要掌握 C 语言和 Java 语言，这对你以后面对各式各样的编程语言是非常重要的。

此外，还推荐学习 Go 语言，它已成为云计算领域事实上的标准语言，尤其是在 Docker、Kubernetes 等项目中。而且，Go 语言在国内外一些知名公司中有了一定的应用和实践，并且其生态圈也越来越好。

算法和数据结构。这个太重要了，尤其是最基础的算法和数据结构，这是任何一个称职的程序员都需要学习和掌握的。你必需要掌握。

计算机的相关系统。你至少要掌握三个系统的基础知识，一个是操作系统，一个是网络系统，还有一个是数据库系统。它们分别代表着计算机基础构架的三大件——计算、存储、网络。

如果你能够走到这里，把前面的那些知识都了解了（不用精通，因为精通是需要时间和实践来慢慢锤炼出来的，所以，你也不用着急），那么你已经是一个非常非常合格的程序员了，而且你的潜力和可能性是非常非常高的。

如果经历过这些比较枯燥的理论知识，而且你还能有热情和成就感，那么我要恭喜你了。因为你已经超过了绝大多数人，而且还是排在上游的比较抢手的程序员了。我相信你至少可以找到年薪 50 万以上的工作了。

但是，你还需要很多的经验或是一些实践，以及一些大系统大项目的实际动手的经验。没关系，我们后面会有教你怎么实操的方法和攻略。

但是，往后面走，你需要开始需要术业有专攻了。下面给一些建议的方向。

- **底层方向**：操作系统、文件系统、数据库、网络……
- **架构方向**：分布式系统架构、微服务、DevOps、Cloud Native……
- **数据方向**：大数据、机器学习、人工智能……
- **前端方向**：你对用户体验或是交互更感兴趣，那么你走前端的路吧。
- **其它方向**：比如，安全开发、运维开发、嵌入式开发……

这些方向你要仔细选择，因为一旦选好，就要勇往直前地走下去，当然，你要回头转别的方向也没什么问题，因为你有前面的这些基础知识在身，所以，不用害怕。只是不同的方向上会有不同的经验积累，经验积累是看书看不来的，这个是转方向的成本。

下篇文章，我们将进入《软件设计篇》。敬请期待。

下面是《程序员练级攻略（2018）》系列文章的目录（持续更新中）。

  - 编程语言
  - 理论学科
  - 系统知识

- 软件设计篇
  - 软件设计

- 高手成长篇

精选留言

耗子粉丝 14

周二周四等待耗子叔更新文章成了一种习惯！程序员练级攻略这个系列，真是太精彩了！作为计算机科班出生的半吊子！整天想着怎样读完这一系列推荐书！是不是要辞职精心研究一到两年，再去上班！纠结！这些内容真的太精彩了！

2018-06-19

Daemon.An 9

耗子叔，有一个问题想问您，就是您推荐的《深入理解计算机系统》、《Unix环境高级编程》、《TCP/IP》等这些书都出了新版，但是有些已经不是原作者了；但是如果看旧版的，有些知识可能已经有点过时了。对于老版、新版之间的区别以及选择，您怎么看的呢？希望耗子叔指点迷津

2018-06-19

JasonHu 5

已经在 Github 上整理了这系列文章中提到的书 https://github.com/jasonim/ebook

2018-06-20

saiyn 4

郁闷啊，这么多篇都是介绍一些链接内容

2018-06-19

作者回复

对不住了。另外，这系列的文章只是画个地图。路还是要自己一步一步走的，饭还是要自己一口口吃的。

2018-06-20

墨梵 3

一个计算机科班出身毕业7-8年的半吊子表示，感谢耗子哥指明了方向

2018-06-19

笨笨熊 3

感谢皓哥的精彩分享。关于Wireshark抓包这两本书「Wireshark网络分析就这么简单」、「Wireshark网络分析的艺术」也很不错。

2018-06-19

蹦蹦逗 2

“如果想看完我推荐的那些书和知识，并能理解和掌握，我估计怎么也得需要 4-5 年的时间。”

耗子哥，“4-5年”是脱产情况下4-5年吗？

2018-06-19

作者回复

无996的在职

2018-06-20

林子 2

这"货"干的有点脱水吧。。。

2018-06-19

KingPoker 2

回头看，大学四年多多努力多重要

2018-06-19

escray 1

如果能在「左耳听风读者群」里面，和大家一起打磨升级就更好了。用 5-7 年的时间，成为高手

2018-06-19

云学 1

作者提到的这些经典书籍每本都涉猎过，也难得自己静下心来看这些书，我现在保持每天看书的习惯，但是理解还是有限的，毕竟工作中用的不多，转行的代价不在于看书而在于经验积累，深有感触！！

2018-06-19

冷雨 1

关键还是太懒了

2018-06-19

力挽狂澜爆炸输出的臭臭宁 1

我觉得这篇文章最让我激动的一点就是，他给出了一些实践案例；平时也会看其他一些技术文章，但是深刻感觉到只看理论不实践很快就会忘记并且对理论也会缺乏深刻理解，特别希望有经验的人能给出一些简易但是有深度的实践案例，看完这篇文章真的很佩服作者

2018-06-19

守护露 0

我刚开始学linux的，可以看unix环境高级编程吗？

2018-06-20

JasonHu 0

有没有读者群 朋友可以发一个吗

2018-06-20

作者回复

看我的微博置顶

2018-06-20

75819 0

谁能告诉我为什么要读unix环境高级编程？

2018-06-20

作者回复

这是操作系统的系统调用，上层的所有语言都逃不掉这些原理

2018-06-20

NonStatic 0

我个人的经验是Fiddler比wireshark好用，尤其是抓https包的时候。请问耗子哥为啥推荐wireshark呢？

2018-06-19

D瓜哥 0

我觉得《编码》这本书从灯光，盲文开始展开，说明信息编码的原理，然后代领大家做一个CPU，也非常通俗易懂。

2018-06-19

zliweijk

0

计算机科班出身的半吊子感谢耗子叔的分享，一切都只是时间，觉得看书枯燥算是没热情吗？

2018-06-19

作者回复

枯燥也得咬牙啃，好多知识都这样

2018-06-20

子轩Zixuan

0

请问有左耳听风读者群吗

2018-06-19