

# 如何成为一名黑客

*Copyright © 2001 by Eric S. Raymond*

翻译: Barret 翻译水平有限, 欢迎来信指教, 我的 Email 是 [barret@ynmail.com](mailto:barret@ynmail.com), 但请勿问电脑技术问题 (反正也不懂。)

允许未经作者及译者的同意进行非商业目的的转载, 但必须保持原文的完整性。

---

**内容一览** [为什么会有这份文档?](#) [什么是黑客?](#) [黑客应有的态度](#) [黑客的基本技能](#) [黑客文化中的地位](#) [黑客和书呆子\(Nerd\)的联系](#) [风格的意义](#) [其它资源](#) [FAQ \(常问问题解答\)](#)

---

## 为什么会有这份文档?

作为 [Jargon File](#) 的编辑和 一些其他有名的类似性质文章的作者, 我经常收到充满热情的网络新手的email提问 (确实如此) “我如何才能成为一名出色的黑客?” 非常奇怪的是似乎没有任何的FAQ或者Web形式的文档来说明这个十分重要的问题, 因此我写了一份。

如果你现在读的是这份文档的离线拷贝, 那么请注意当前最新版本 (英文版) 在 <http://www.tuxedo.org/~esr/faqs/hacker-howto.html> 可以得到。

注意: 在这份文档最后有 [FAQ \(常问问题解答\)](#)。请在向我提出任何关于这份文档的疑问之前读两遍。

目前这份文档有很多翻译版本: [保加利亚语](#), [简体中文](#), [繁体中文](#), [丹麦语](#), [荷兰语](#), [法语](#), [德语](#), [匈牙利语](#), [印尼语](#), [日语](#), [朝鲜语](#), [葡萄牙语](#), [俄语](#) 及 [瑞典语](#)。注意由于这份文档时有修正, 所以以上翻译版本可能有不同程度的过时。

---

## 什么是黑客?

[Jargon File](#) 包含了一大堆关于“hacker”这个词的定义, 大部分与技术高超和热衷解决问题 及 超越极限有关。但如果你只想知道如何 **成为** 一名黑客, 那么只有两件事情确实相关。

这可以追溯到几十年前第一台分时小型电脑诞生, ARPAnet 实验也刚展开的年代, 那时有一个由程序设计专家和网络名人所组成的, 具有分享特点的文化社群。这种文化的成员创造了 “hacker” 这个名词。**黑客们建立了 Internet。黑客们发明出了现在使用的 UNIX 操作系统。黑客们使 Usenet 运作起来, 黑客们让 WWW 运转起来。**如果你是这个文化的一部分, 如果你对这种文化有所贡献, 而且这个社群的其它成员也认识你并称你为 hacker, 那么你就是一位黑客。

黑客精神并不仅仅局限在软件的黑客文化中。有人用黑客态度对待其它事情，如电子学和音乐——事实上，你可以在任何最高级别的科学和艺术活动中发现它。精于软件的黑客赞赏这些在其他领域的同类并把他们也称作黑客——有人宣称**黑客天性是绝对独立于他们工作的特定领域的**。但在这份文档中，我们将注意力集中在软件黑客的技术和态度，以及发明了“黑客”一词的以共享为特征的文化传统之上。

有一群人大声嚷嚷着自己是黑客，但他们不是。他们（主要是正值青春的少年）是一些蓄意破坏计算机和电话系统的人。真正的黑客把这些人叫做“骇客”(cracker)，并不屑与之为伍。多数真正的黑客认为骇客们又懒又不负责任，还没什么大本事。专门以破坏别人安全为目的的行为并不能使你成为一名黑客，正如用铁丝偷开走汽车并不能使你成为一个汽车工程师。不幸的是，很多记者和作家往往错把“骇客”当成黑客；这种做法一直使真正的黑客感到恼火。

根本的区别是：**黑客搞建设，骇客搞破坏。**

如果你想成为一名黑客，请接着读下去。如果你想做一个骇客，去读 [alt.2600](#) 新闻组，并在意识到你并不像自己想象的那么聪明后去坐五到十次监狱。关于骇客，我只想说这么多。

---

## 黑客应有的态度

黑客们解决问题，建设事物，同时他们**崇尚自由和无私的双向帮助**。要被他人承认是一名黑客，你的行为得体现出你好像具备了这种态度一般。而要想做得好象你具备这种态度一般，你就得切切实实坚持它。

但是如果你认为培养黑客态度只是一条在黑客文化圈中得到承认的路子，那就大错特错了。成为具备这些特质的这种人对**你自己**非常重要——有助于你学习，及给你提供源源不断的动力。同所有创造性的艺术一样，成为大师的最有效方法就是模仿大师的精神——不仅从智力上，也要从感情上进行模仿。

或许，下面这首现代的禅诗很好的阐述了这个意思：

To follow the path: (沿着这样一条道路：)

look to the master, (寻找大师，)

follow the master, (跟随大师，)

walk with the master, (与大师同行，)

see through the master, (洞察大师，)

become the master. (成为大师。)

嗯，如果你想成为一名黑客，反复读下面的事情直至你相信它们：

---

### 1. 世界充满了待解决的迷人问题。

做一名黑客会有很多乐趣，但却是**要费很多气力方能得到的乐趣**。这些努力需要动力。成功的运动员从锻炼身体、超越自我极限的愉悦中得到动力。同样，做黑客，**你得能从解决问题，磨练技术及锻炼智力中得到基本的乐趣**。

如果你还不是天生的这类人又想做黑客，你就要设法成为这样的人。否则你会发现，你的黑客热情会被其他分心的事物吞噬掉——如**金钱、性和社会上的虚名**。

**（同样你必须对你自己的学习能力建立信心——相信尽管你对某问题近乎一无所知，但只要你一点一点地试验、学习，最终会掌握并解决它。）**

---

## **2. 一个问题不应该被解决两次。**

聪明的脑袋是宝贵的有限的资源。当世界还充满非常多有待解决的有趣的新问题时，**它们不应该被浪费在重新发明轮子这类事情上**。

作为一名黑客，**你必须相信其他黑客的思考时间是宝贵的——因此共享信息，解决问题并发布结果给其他黑客几乎是一种道义**，这样其他人就可以去解决**新**问题而不是不断地忙于对付旧问题。

(你不必认为一定要把你**所有的**发明创造公布出去，但这样做的黑客是赢得大家极度尊敬的人。卖些钱来养家糊口，租房买计算机，甚至发大财和黑客价值观也是相容的，只要你别忘记你还是个黑客。)

---

## **3. 无聊和乏味的工作是罪恶。**

黑客（泛指具有创造力的人们）应该从来不会被愚蠢的重复性劳动所困扰，因为当这种事情发生时就意味着他们没有在做只有他们才能做的事情——解决新问题。这样的浪费伤害每一个人。因此，无聊和乏味的工作不仅仅是令人不舒服而已，而且是罪恶。

作为一个黑客，你必须坚信这点并尽可能多地**将乏味的工作自动化**，不仅为你自己，也为了其他人（尤其是其他黑客们）。

(对此有一个明显的例外。黑客有时也做一些在他人看来是重复性或枯燥的工作以进行“脑力休息”，或是为了获得某种技能，或是获得一些除此以外无法获得的特别经验。但这是自愿的——**有脑子的人不应该被迫做无聊的活儿**。)

---

## **4. 自由万岁。**

黑客们是天生的反独裁主义者。任何能向你发命令的人能够迫使你停止解决令你着迷的问

题，同时，按照独裁者的一般思路，他通常会给出一些极端愚昧的理由。因此，不论何处，任何独裁主义的作法，只要它压迫你和其他黑客，你就要和它斗到底。

(这并非向所有权威挑战。儿童需要监护，罪犯要被看管起来。如果服从命令得到某种东西比起用其他方式得到它更节约时间，黑客可以同意接受某种形式的权威。但这是一个有限度的，有意的交易；那种权威想要的个人服从不是你应该同意给予的。)

权威喜欢审查和保密。他们不信任自愿的合作和信息的共享——他们只喜欢由他们控制的所谓“合作”。因此，作为一个黑客，你得对审查、保密，以及使用武力或欺骗去压迫有行为能力的人们做法有一种本能的敌意。同时你要为此信念斗争的意愿。

---

## 5. 态度不能替代能力。

作为一名黑客，你必须培养起这些态度。但只具备这些态度并不能使你成为一名黑客，也不能使你成为一个运动健将和摇滚明星。成为一名黑客需要智力，实践，奉献精神 and 辛苦工作。

因此，**你必须学会怀疑，并尊重各种各样的能力。**黑客们不会为那些装模做样的人浪费时间，但他们却非常尊重能力——尤其是从事黑客工作的能力，不过任何能力总归是好的。具备很少人能具备的那些方面的能力尤其好，其中具备涉及脑力、技巧和专注方面能力的当然最好。

尊敬能力，**你就会享受到提高自己能力的乐趣——辛苦的工作和奉献会变成一种高度娱乐而非苦差事。**要想成为一名黑客，这一点非常重要。

---

## 黑客的基本技能

黑客态度重要，但技术更加重要。态度无法替代技术，在你被别的黑客称为黑客之前，有一些基本的技术你必须掌握。

这些基本技术随着新技术的出现和老技术的过时也随时间在缓慢改变。例如，过去内容包括使用机器语言编程，而直到最近才包括了HTML。总的来说现在主要包括以下技术：

---

### 1. 学习如何编程。

这当然是最基本的黑客技能。**如果你还不会任何编程语言，我建议你从Python开始。**它设计清晰，文档齐全，合适初学者入门。它是一门很好的入门语言，并且不仅仅只是个玩具；

它非常强大、灵活，**也适合做大型项目**。我有一篇 [Python评价](#) 详细说明这点。好的 [教程](#) 可以在 [Python](#) 网站得到。（译者：比较好的中文Python站点可能是<http://pythonrecord.51.net>。）

**Java也是好的入门语言。它比Python难得多，但是生成的代码速度也快得多。** 它同时也是一种优秀的计算机语言，**不止是用来入门**。

但是注意，如果你只会一两门语言，你将不会达到黑客所要求的技术水平，甚至也不能达到一个程序员的水平——**你需要学会如何以抽象的方式思考编程问题**，独立于任何语言。要做一名真正的黑客，**你需要学会在几天内通过一些手册，结合你现在所知，迅速掌握一门新语言**。这意味着你应该学会几种截然不同的语言。

**如果要做一些重要的编程工作，你将不得不学习C语言，Unix的核心语言。C++与C非常类似；如果你了解其中一种，学习另一种应该不难。但这两种都不适合编程入门者学习。而且事实上，你越避免用C编程，你的工作效率会越高。**

**C非常有效率，节约你的机器资源。不幸的是，C的高效是通过你手动做很多底层的管理（如内存）来达到的。底层代码都是复杂极易出现bug的，会使你花极多的时间调试。如今的机器速度如此之快，这通常是得不偿失——比较明智的做法是使用一种运行较慢、较低效率，但大幅节省你的时间的语言。**因此，选择Python。

其他对黑客而言比较重要的语言包括 Perl和 LISP。Perl实用，值得一学；**它被广泛用于动态网页和系统管理**，因此即便你从不用Perl写程序，至少也应该 **学会看**。许多人使用Perl的理由和我建议你使用Python的理由一样，都是为了避免用C完成那些不需要C高效率的工作。你会需要理解那些工作的代码的。

LISP值得学习的理由不同——**最终掌握了它时你会得到丰富的启迪和经验**。这些经验会使你在以后的日子里成为一个更好的程序员，即使你实际上很少使用LISP本身。

当然，**实际上你最好五种都会（Python，Java，C/C++，Perl，LISP）**。除了是最重要的黑客语言外，它们还**代表了截然不同的编程思路和方法**，每种都会让你受益非浅。

这里我无法给你完完全全的指导教会你如何编程——这是个复杂的技能。但我可以告诉你，书本和上课也不能作到（最好的黑客中，有许多，也许 **几乎** 都是自学成材的）。你可以从书本上学到语言的特点——只是一些皮毛，但要使书面知识成为自身技能只能通过实践和虚心向他人学习。因此要做到到 **（1）读代码** 及 **（2）写代码**。

**学习如何编程就象学习用优美的自然语言写作一样。最好的做法是读一些大师的名著，试着自己写点东西，再读些，再写点，再读些，再写点..... 如此往复，直到你的文章达到你体会到的范文的简洁和力量。**

**过去**找到适合阅读的好的代码是困难的，因为几乎没有大型程序的源代码能让新手练手。这种状况已经戏剧性地发生变化；开放源代码软件，编程工具和操作系统（全都由黑客写成）现在已经随处可见。让我们在下一个话题中继续讨论.....

---

## **2. 得到一个开放源代码的Unix并学会使用、运行它。**

我假设你已经拥有或者能使用一台个人电脑（今天的孩子们真幸福 :-) ）。新手们能够朝学习黑客技能迈出的最基本的一步就是得到一份Linux或BSD-Unix的一种，安装在个人电脑

上，并运行它。

没错，**这世界上除了Unix还有其他操作系统**。但它们都是以二进制形式发布的——你无法读到它的源代码，也不可能修改它。**尝试在运行DOS或Windows或MacOS的机器上学习黑客技术，就象是带着脚镣学跳舞。**

除此之外，**Unix还是Internet的操作系统**。你可以学会上网却不知道Unix，但你不了解Unix就无法成为一名Internet黑客。因此，**今天的黑客文化在很大程度上是以Unix为中心的**。（这点并不总是真的，一些很早的黑客对此一直很不高兴，但Unix和Internet之间的联系已是如此之强，甚至连Microsoft也无可奈何。）

所以，安装一套**Unix**——我个人喜爱Linux，但还有其他种类的（是的，你可以同时安装Linux及DOS/Windows在同一电脑上）。学习它，使用它，配置它。用它在Internet上冲浪，阅读它的源代码，修改它的源代码。你会得到比在Microsoft操作系统上更好的编程工具（包括C，LISP，Python及Perl）。**你会觉得乐趣无穷**，学到在你成为大师之前意识不到的更多的知识。

想知道更多关于学习Unix的信息，访问 [The Loginataka](#)。

想知道如何得到一份Linux，访问 [我在哪里可以获得Linux](#)。

你可以在 [www.bsd.org](http://www.bsd.org)找到BSD Unix的求助及其他资源。

我有写一篇关于 [Unix和Internet基础](#)的入门文章。

（注：如果你是一个新手，我不推荐自己独立安装Linux或者BSD。安装Linux的话，寻求本地Linux用户组的帮助；或联系 [Open Projects Network](#)。LISC维护着一些 [IRC频道](#)，在那里你可以获得帮助。）

---

### **3. 学会如何使用WWW和写HTML**

黑客文化建造的大多东西都在你看不见的地方发挥着作用，帮助工厂、办公室和大学正常运转，表面上很难看到它对非黑客的普通人生活的影响。Web是一个大大的例外。即便 **政客**也同意，这个巨大耀眼的黑客玩具正在改变整个世界。单是这个原因（还有许多其它的），你就需要学习掌握Web。

这并不是仅仅意味着如何使用浏览器（谁都会），而是要**学会如何写HTML，Web的标记语言**。**如果你不会编程，写HTML会教你一些有助于学习的思考习惯**。因此，先完成一个主页。（网上有很多好的教程。）

但仅仅拥有一个主页不能使你成为一名黑客。Web里充满了各种网页。大多数是毫无意义的，零信息量垃圾——界面时髦的垃圾，注意，垃圾的水准都类似（更多信息访问 [The HTML Hell Page](#)）。

要想有价值，你的网页必须有**内容**——它必须有趣或对其它黑客有帮助。这是下一个话题所涉及的……

---



## 4. 如果你不懂实用性的英语，学习吧。

作为一个美国人和一个以英语为母语的人，我以前很不情愿提到这点，免得成为一种文化上的帝国主义。但相当多以其他语言为母语的人一直劝我指出这一点，那就是 English 是黑客文化和 Internet 的工作语言，你需要懂得以便在黑客社区顺利工作。

这一点千真万确。大概 1991 年的时候我就了解到许多黑客在技术讨论中使用英语，甚至当他们的母语都相同，英语对他们而言只是第二语言的时候；据我知道的报导，当前英语有着比其他语言丰富得多的技术词汇，因此是一个对于工作来说相当好的工具。基于类似的原因，英文技术书籍的翻译通常不令人满意（如果有翻译的话）。

Linus B.Torvalds，一个芬兰人，用英语注释他的代码（很明显这对他来说不是凑巧）。他流利的英语成为他能够管理全球范围的 Linux 开发人员社区的重要因素。这是一个值得学习的例子。

---

## 黑客文化中的地位

象大部分不涉及金钱的文化一样，黑客王国靠**声誉**运转。你设法解决有趣的问题，但它们到底多有趣，你的解法有多好，**是要由那些和你具有同样技术水平的人或比你更厉害的人去评判的。**

相应地，当你在玩黑客游戏时，你得认识到你的分数主要靠其他黑客对你的技术的评价给出（这就是为什么只有在其它黑客称你为黑客时，你才算得上是一名黑客）。这个事实常会被黑客是一项孤独的工作这一印象所减弱；也会被另一个黑客文化的禁忌所减弱（现在逐渐减弱但仍强大）：拒绝承认自我或外部评估与一个人的动力有关系。

特别地，黑客王国被人类学家们称为一种奉献文化。在这里你不是凭借你对别人的统治来建立地位和名望，也不是靠美貌，或拥有其他人想要的东西，而是靠你的奉献。尤其是奉献你的时间，你的创造和你的技术成果。

要获得其他黑客的尊敬，基本上有五种事情你可以干：

---

## 1. 写开放源代码软件

第一个（也是最集中的和传统的）是**写些被其他黑客认为有趣或有用的程序，并把程序源代码提供给整个黑客文化使用。**

（过去我们称之为“free software（自由软件）”，但这却使很多不知 free 的精确含义的人感到困惑。现在我们很多人，根据搜索引擎网页内容分析至少有 2：1 的比率，使用“open-source”software（开放源代码软件）这个词）。

黑客王国里最受尊敬的偶像是那些写了大型的、好用的、具有广泛用途的软件， 并把它们公布出去，使得每人都在使用他软件的人。

---

## **2. 帮助测试并调试开放源代码软件**

黑客也尊敬那些**使用、测试开放源代码软件的人**。在这个并非完美的世界上，我们不可避免地要花大多数的开发时间在调试阶段。这就是为什么任何有头脑的开放源代码的作者都会告诉你好的beta测试员（知道如何清楚描述出错症状，很好地定位错误，能忍受快速发布中的bug，并且愿意使用一些简单的诊断工具）象红宝石一样珍贵。甚至他们中的一个能判断出哪个测试阶段是延长的，哪个是令人精疲力尽的噩梦，哪个只是一个有益的小麻烦。

如果你是个新手，试着找一个你感兴趣的正在开发的程序，做一个好的beta测试员。你会自然地帮着测试，进步到帮着抓臭虫，到最后帮着改程序。你会从中学到很多，并且与未来会帮你的人结下友谊。

---

## **3. 公布有用的信息**

另一个好事是**收集整理有用有趣的信息做成网页或文档如FAQ列表**，且让他们容易获得。

主要技术FAQ的维护者受到几乎同其他开放源代码的作者一样多的尊敬。

---

## **4. 帮助维护基础设施的运转**

黑客文化（还有Internet的工程方面的发展，就此而言）是靠自愿者运转的。要使Internet能正常工作，就要有大量枯燥的工作不得不去完成——管理mail list，新闻组，维护大型软件库，开发RFC和其它技术标准等等。

做这类事情的人会得到很多尊敬，因为每人人都知道这些事情是十分花时间又不象编程那样好玩。做这些事情需要奉献精神。

---

## **5. 为黑客文化本身服务**

最后，你可以**为这个文化本身做宣传**（例如，象我这样，写一个“如何成为黑客”的正面的教



程 :-)) (译者：不知道Barret把它翻成中文算不算？)。这并非一定要在你已经在这个圈子呆了很久，因以上四点中的某点而出名，有一定声誉后才能去做。

黑客文化没有领袖。精确地说，它确实有些文化英雄、部落长者、历史学家和发言人。若你在这圈内呆的够长，你或许成为其中之一。记住：**黑客们不相信他们的部落长者的自夸的炫耀，因此大举追求这种名誉是危险的。与其奋力追求，不如先摆正自己的位置 等它自己到你的手中——那时则要做到谦虚和优雅。**

---

## 黑客和书呆子(Nerd)的联系

同流行的迷思相反，做一名黑客并不一定要你是个书呆子。但它确实有帮助，而且许多黑客事实上是书呆子。**做一个深居简出的人有助于你集中精力进行十分重要的事情，如思考和编程。**

因此，很多黑客都愿意接受“书呆子”这个外号，更有甚者使用更尖刻的“geek（怪人）”一词并引以为豪——这是一种宣布他们独立于主流社会的声明方式。访问 [The Geek Page](#) 参加更多的讨论。

**如果你能集中足够的精力做好黑客工作同时还能有正常的生活，这很好。**现在作到这一点比我在1970年代是新手的时候要容易的多；如今主流文化对技术怪人要友善的多。甚至有越来越多的人意识到黑客通常是很好的恋人和配偶的材料。

如果你因为生活上不如意而迷上做黑客，那也没什么——至少你不会分神了。或许以后你会找到自己的另一半。

---

## 风格的意义

重申一下，作为一名黑客，你必须进入黑客精神之中。当你不在计算机边上时，你仍然有很多对黑客工作有帮助的事情可做。它们并不能替代真正的编程（没有什么能），但很多黑客都那么做，并感到它们与黑客的本质存在某些基本的连系。

- ◆ **学会流畅地用母语写作。尽管程序员不能写好文章的偏见已相当普遍，但是仍有有数目惊人的黑客——包括所有我知道的最棒的，都是不错的作家。**
- ◆ **阅读科幻小说。**参加科幻小说讨论会。（一个碰到黑客和未来会成为黑客的人的好方法）
- ◆ **学禅，并且/或者练习习武。**（精神修炼看来是惊人相似。）
- ◆ **练就能分析音乐的听觉，学会鉴赏特别的音乐。学会玩某种乐器，或唱歌。**
- ◆ **提高对双关语、文字游戏的鉴赏能力。**

这些事情，你已经做的越多，你就越是天生做黑客的材料。至于为什么偏偏是这些事情，原

因并不完全清楚，但它们都涉及用到**左—右脑**能力的综合，这似乎是关键所在（黑客们既需要清晰的逻辑思维，有时又需要偏离逻辑跳出问题的表象）。

最后，还有一些 **不要** 去做的事情。

**不要使用愚蠢的，哗众取宠的ID或昵称。**

**不要卷入Usenet（或其他地方的论坛）的骂战。**

**不要自称为“cyberpunk（网络叛客）”，也不要浪费时间和那些人打交道。**

**不要让你寄出的Email或张贴的帖子充满错误的拼写和乱七八糟的语法。**

做以上的事情，只会招来嘲笑。**黑客们个个记忆超群—— 你将需要数年的时间让他们忘记你犯下的错误。**

网名的问题值得深思。将身份隐藏在虚假的名字后是骇客、解密者、d00dz 及其他低等生物幼稚愚蠢的行为特点。黑客不会做这些事；他们对他们 **所做的** 感到骄傲，而且乐于人们将 **作品** 与他们的 **真名** 相联系。因此，若你现在用假名，放弃它。在黑客文化里它会令你失败的。

---

## 其它资源

Peter Seebach 维护着一个非常好的 [Hacker FAQ](#)，专给那些不懂如何与黑客打交道的经理看的。如果Peter的站点不能访问，下面这个 [Excite搜索](#)应该有一份拷贝。

我也著有 [黑客文化简史](#)。

我写了一份 [大教堂与市集](#)，对于Linux及开放源代码文化现象有详细的解释。我也在这个话题上进一步阐述导致的结局—— [开拓智域](#)。

Rick Moen写了一份很好的关于 [如何运转一个Linux用户组](#)的文档。

我和Rick Moen合作完成了另一份关于 [提问的智慧](#)的文章，可以让你事半功倍的获得帮助。

如果你想知道PC、UNIX及Internet基本概念和工作原理，参考 [The Unix and Internet Fundamentals HOWTO](#)。

当你释放出一个软件或为其打补丁，试着按 [软件发行惯例 HOWTO](#)去做。（以上的提到的文章的中文版大多都可以在[www.aka.org.cn](#)和[www.linuxforum.net](#)找到。）

---

## FAQ（常问问题解答）

问：[你能教我做黑客吗？](#) 问：[那么，我要如何开始？](#) 问：[我得什么时候开始学？现在会不会太迟了？](#) 问：[要学多久才能学会黑客道？](#) 问：[Visual Basic及Delphi是好的入门语言吗？](#)

问：你能帮我“黑”掉一个站点吗？或者教我怎么黑它？ 问：我怎样才能得到别人帐号的密码？ 问：我如何入侵/查看/监视别人的Email？ 问：我如何才能在IRC聊天室里偷到频道op的特权？ 问：我被黑了。你能帮我避免以后再被攻击吗？ 问：我的Windows软件出现问题了。你能帮我吗？ 问：我在哪里能找到可以与之交流的真正的黑客？ 问：你能推荐一些有关黑客的好书吗？ 问：成为一名黑客我需要擅长数学吗？ 问：我该从那种语言学起？ 问：我需要什么样的机器配置？ 问：我得因此憎恨和反对Microsoft吗？ 问：但开放源代码软件不会使程序员丢饭碗吗？ 问：我要如何开始？哪里有免费的Unix？

**问：你能教我做黑客吗？**

**答：**自从第一次发布这份文档，我每周都会收到一些请求，（频繁的话一天几封）要我“教会他们做黑客”。遗憾的是，我没有时间和精力来做这个；我自己的黑客项目，及我作为一个**开放源代码倡导者**的四处奔波已经占用了我110%的时间。

即便我想教你，黑客也依然基本上是一项**自行修炼的的态度和技术**。当真正的黑客想帮助你的时候，如果你乞求他们一汤匙一汤匙“喂”你的话，你会发现他们不会尊重你。

先去学一些东西。显示你在尝试，你能靠自己去学习。然后再去向你遇到的黑客请教特殊的问题。

如果你发E-mail给一位黑客寻求他的帮助，这是两件首要记住的事情。第一，写出来的文字显得懒且粗心的人通常非常懒于思考且非常马大哈，不能成为好黑客——因此注意拼写正确，使用正确的语法及发音，否则你可能会无人理睬。第二，不要试图要求回复到一个ISP帐号，而那个帐号与你的发信地址不同。这样做的人一般是使用盗用帐号，不会有人有兴趣为虎作伥帮助窃贼的。

**问：那么，我要如何开始？**

**答：**对你而言最佳的入门方式也许是去参加LUG（Linux用户组）的聚会。你可以找到在LDP的综合Linux信息页面上找到类似的组织；也许有一个在你家附近的，而且非常有可能与一所大学或学校挂钩。如果你提出要求，LUG成员兴许会给你一套Linux，当然此后会帮你安装并带你入门。

**问：我得什么时候开始学？现在会不会太迟了？**

**答：**你有动力学习的时候就是好时候。大多数人看来都是在15—20岁之间开始感兴趣的，但据我所知，在此年龄段之外的例外也是有的。

**问：要学多久才能学会黑客道？**

**答：**这取决于你的聪明程度和努力程度。大多数人只要他们专注，**就能在18个月到2年之间学会一套令人尊敬的技能**。但是，不要以为就此结束了；**如果你是一个真正的黑客，你要用你的余生来学习和完善你的技术**。

**问：Visual Basic及Delphi是好的入门语言吗？**

**答：不，因为他们不是可移植的。**他们不是那些语言的开放源代码实现，所以你被限制在厂商选择支持的那些平台里。接受这样一种垄断局面不是黑客的态度。

Visual Basic特别糟糕。它是Microsoft的私有语言这个事实就足够让它脸面全无，不像其他的Basic，它是一种设计糟糕的语言会教给你坏的编程习惯。

其中一个坏习惯是**会依赖于单一厂商的函数库、控件及开发工具**。一般而言，任何不能够支持至少Linux或者一种BSD，或其他第三方操作系统的语言，都是一种不适合应付黑客工

作的语言。

**问：你能帮我“黑”掉一个站点吗？或者教我怎么黑它？**

**答：**No。任何读完这份FAQ后还问这个问题的人，都是无可救药的蠢材，即使有时间指教我也不会理睬。任何发给我的此类E-mail都会被忽略或被痛骂一顿。

**问：我怎样才能得到别人帐号的密码？**

**答：**这是骇客行为。滚得远远的，白痴。

**问：我如何入侵/查看/监视别人的Email？**

**答：**这是骇客行为。在我面前消失，混蛋。

**问：我如何才能在IRC聊天室里偷到频道op的特权？**

**答：**这是骇客行为。去死吧，冥顽不灵的家伙。

**问：我被黑了。你能帮我避免以后再被攻击吗？**

**答：**不行。目前为止，**每次问我这个问题的，都是一些运行Microsoft Windows的菜鸟。**不可能有效的保护Windows系统免受骇客攻击；太多缺陷的代码和架构使保护Windows的努力有如隔靴搔痒。**唯一可靠的预防来自转移到Linux或其他设计得至少足够安全的系统。**

**问：我的Windows软件出现问题了。你能帮我吗？**

**答：**当然。进入DOS方式，然后键入“format c:”。你遇到的任何问题将会在几分钟之内消失。

**问：我在哪里能找到可以与之交流的真正的黑客？**

**答：**最佳办法是在你附近找一个Unix或Linux的用户组，参加他们的聚会。（你可以在Metalab的LDP站点 找到一些指向用户组的链接。）

我过去曾说过不能在IRC上找到真正的黑客，但我发觉现在情况有所改变。显然一些真正的黑客的社区像GIMP及Perl，也有IRC频道了。）

**问：你能推荐一些有关黑客的好书吗？**

**答：**我维护着一份Linux Reading List HOWTO，也许你会觉得有用。Loginataka也很有意思。关于Python的介绍，请访问在Python站点上的 [入门资料](#)。

**问：成为一名黑客我需要擅长数学吗？**

**答：**不用。黑客道很少使用常规的数学或算术，不过你绝对需要 **学会逻辑性地思考和进行精密的推理。**

尤其是你不会用到微积分或电路分析（我们把这些留给电子工程师们 :-))）。一些有限数学（包括**集合论，布尔代数，组合数学，图论**）的背景知识会有帮助。

**问：我该从那种语言学起？**

**答：HTML——如果你还不懂的话。**市面上有一大堆的封面精美，宣传得天花乱坠的 **糟糕的** HTML书籍，不幸的是很少有好的。我最喜欢的是 [HTML: The Definitive Guide](#)。

但 HTML 不完全是一种编程语言。当你准备开始编程时，我推荐从 [Python](#)起步。你会听到一大群人推荐 Perl，并且 Perl 依然比 Python 流行得多，但是难学得多且（以我之见）设计得不是很好。

C 确实重要，但它要比 Python 或 Perl 难多了。不要尝试先学C。

Windows用户 **不要** 满足于 Visual Basic。它会教给你坏习惯，而且它不可以移植，只能在 Windows下运行。避开它。

**问：我需要什么样的机器配置？**

**答：**过去个人电脑能力相当不够并且内存小，结果给黑客的学习过程设置了人为的障碍。不过一段时间以前开始就不是这样了；任何配置比一台 Intel 486DX50 好的 机器都有足够的能力进行开发工作，X，及 Internet 通讯，同时你现在买的最小的磁盘 都大得富足了。（依 Barret之见，现在要至少Pentium 166MMX才够。）

选择用来学习的机器时重要的一点是注意配件是否是 Linux 兼容的（或BSD 兼容，如果你选择学 BSD）。同刚才提到的一样，大多数现在的机器都是符合的；唯一的值得注意的区域在于 modem和打印机；有些具备为Windows设计的配件的机器不会在Linux 下工作。

关于硬件兼容性有一个FAQ；最新版本在 [这里](#)。

**问：我得因此憎恨和反对Microsoft吗？**

**答：**不，你不必如此。不是因为Microsoft不令人讨厌，而是因为黑客文化早在 Microsoft出现之前就存在了，且将在Microsoft成为历史后依然存在。你耗费在憎恨Microsoft的任何力气不如花在爱你的技术上。**写好的代码—— 那会相当有效地打击Microsoft又不会让你得到恶报应。**

**问：但开放源代码软件不会使程序员丢饭碗吗？**

**答：**看起来不太可能——**目前为止，开放源代码软件产业似乎创造了更多的就业机会而不是 减少就业机会。**如果写一个程序比起不写来是纯经济收益的话，那么在写完后，程序员应该得到报酬不管程序是否是开放源代码。并且，无论写出多么“免费自由”的软件，都存在更多对新的，定制的软件的需求。我有这方面更多的论述，放在[开放源代码](#) 网站资料中。

**问：我要如何开始？哪里有免费的Unix？**

**答：**在本份文档的某个地方我已经提到过何处可以得到最常用的免费Unix。

**要成为一名黑客，你需要自立自强，以及自学能力。现在开始吧……**