

图片：



提问的智慧（一）

在另一个网站上看到的一篇美文，当我看到这个标题时，第一印象就是这是在灌水，看了开头几段更觉得无聊，只有当今天我静下心来看完整篇文章后，才觉得它非常有意义。所以转过来给大家,希望大家有所收获:)

提问的智慧

[注] 此文中的“黑客”一词适用于任何技术高手。

提问的智慧

Copyright (C) 2001 by Eric S. Raymond

中文版 Copyleft 2001 by D.H.Grand(nOBODY/Ginux)

英文版: <http://www.tuxedo.org/~esr/faqs/smart-questions.html>

感谢 Eric 的耐心指点和同意, 本文才得以完成并发布,

本指南英文版版权为 Eric Steven Raymond 所有,

中文版版权由 D.H.Grand[nOBODY/Ginux] 所有。

在黑客世界里, 当提出一个技术问题时, 你能得到怎样的回答? 这取决于挖出答案的难度, 同样取决于你提问的方法。本指南旨在帮助你提高发问技巧, 以获取你最想要的答案。

首先你必须明白, 黑客们只偏爱艰巨的任务, 或者能激发他们思维的好问题。如若不然, 我们还来干吗? 如果你有值得我们反复咀嚼玩味的好问题, 我们自会对你感激不尽。好问题是激励, 是厚礼, 可以提高我们的理解力, 而且通常会暴露我们以前从没意识到或者思考过的问题。对黑客而言, “问得好!” 是发自内心的大力称赞。

尽管黑客们有蔑视简单问题和不友善的坏名声, 有时看起来似乎我们对新手, 对知识贫乏者怀有敌意, 但其实不是那样的。

我们不想掩饰对这样一些人的蔑视--他们不愿思考, 或者在发问前不去完成他们应该做的事。这种人只会谋杀时间--他们只愿索取, 从不付出, 无端消耗我们的时间, 而我们本可以把时间用在更有趣的问题或者更值得回答的人身上。我们称这样的人为“失败者”(由于历史原因, 我们有时把它拼作“lusers”)。

我们在很大程度上属于志愿者, 从繁忙的生活中抽出时间来解惑答疑, 而且时常被提问淹没。所以我们无情的滤掉一些话题, 特别是抛弃那些看起来象失败者的家伙, 以便更高效的利用时间来回答胜利者的问题。

如果你觉得我们过于傲慢的态度让你不爽, 让你委屈, 不妨设身处地想想。我们并没有要求你向我们屈服--事实上, 我们中的大多数人最喜欢公平交易不过了, 只要你付出小小努力来满足最起码的要求, 我们会欢迎你加入到我们的文化中来。但让我们帮助那些不愿意帮助自己的人是没有意义的。如果你不能接受这种“歧视”, 我们建议你花点钱找家商业公司签个技术支持协议得了, 别向黑客乞求帮助。

如果你决定向我们求助, 当然不希望被视为失败者, 更不愿成为失败者中的一员。立刻得到有效答案的最好方法, 就是象胜利者那样提问 —— 聪明、自信、有解决问题的思路, 只是偶尔在特定的问题上需要获得一点帮助。

转: 提问的智慧 (二)

= 提问之前做的准备工作 =

在通过电邮、新闻组或者聊天室提出技术问题前, 检查你有没有做到:

1. 通读手册，试着自己找答案。
2. 在 FAQ 里找答案（一份维护得好的 **FAQ(注解在最后)** 可以包罗万象:）。
3. 在网上搜索（个人推荐 google~~~）。
4. 向你身边精于此道的朋友打听。

当你提出问题的时候，首先要说明在此之前你干了些什么；这将有助于树立你的形象：你不是一个妄图不劳而获的乞讨者，不愿浪费别人的时间。如果提问者能从答案中学到东西，我们更乐于回答他的问题。

周全的思考，准备好你的问题，草率的发问只能得到草率的回答，或者根本得不到任何答案。越表现出在寻求帮助前为解决问题付出的努力，你越能得到实质性的帮助。

小心别问错了问题。如果你的问题基于错误的假设，普通黑客（J. Random Hacker）通常会用无意义的字面解释来答复你，心里想着“蠢问题...”，希望着你会从问题的回答（而非你想得到的答案）中汲取教训。

决不要自以为够资格得到答案，你没这种资格。毕竟你没有为这种服务支付任何报酬。你要自己去“挣”回一个答案，靠提出一个有内涵的，有趣的，有思维激励作用的问题--一个对社区的经验有潜在贡献的问题，而不仅仅是被动的从他人处索要知识--去挣到这个答案。

另一方面，表明你愿意在找答案的过程中做点什么，是一个非常好的开端。“谁能给点提示？”、“我这个例子里缺了什么？”以及“我应该检查什么地方？”比“请把确切的过程贴出来”更容易得到答复。因为你显得只要有人指点正确的方向，你就有完成它的能力和决心。

= 提问的注意事项 =

- 谨慎选择论坛

小心选择提问的场合。如果象下面描述的那样，你很可能被忽略掉或者被看作失败者：

1. 在风马牛不相及的论坛贴出你的问题
2. 在探讨高级技巧的论坛张贴非常初级的问题；反之亦然
3. 在太多的不同新闻组交叉张贴

- 用辞贴切，语法正确，拼写无误

我们从经验中发现，粗心的写作者通常也是马虎的思考者（我敢打包票）。回答粗心大意者的问题很不值得，我们宁愿把时间耗在别处。

正确的拼写，标点符号和大小写很重要。

更一般的说，如果你的提问写得象个半文盲，你很有可能被忽视。

如果你在使用非母语的论坛提问，你可以犯点拼写和语法上的小错--但决不能在思考上马虎（没错，我们能弄清两者的分别）

- 使用含义丰富，描述准确的标题

在邮件列表或者新闻组中，大约 50 字以内的主题标题是抓住资深专家注意力的黄金时机。别用喋喋不休的“帮帮忙”（更别说“救命啊！！！！”这样让人反感的话）来浪费这个机会。不要妄想用你的痛苦程度来打动我们，别用空格代替问题的描述，哪怕是极其简短的描述。

蠢问题：救命啊！我的膝上机不能正常显示了！

聪明问题：XFree86 4.1 下鼠标光标变形，Fooware MV1005 的显示芯片。

如果你在回复中提出问题，记得要修改内容标题，表明里面有一个问题。一个看起来象“Re: 测试”或者“Re: 新 bug”的问题很难引起足够重视。另外，引用并删减前文的内容，给新来的读者留下线索。

- 精确描述，信息量大

1. 谨慎明确的描述症状。
2. 提供问题发生的环境（机器配置、操作系统、应用程序以及别的什么）。
3. 说明你在提问前是怎样去研究和理解这个问题的。
4. 说明你在提问前采取了什么步骤去解决它。
5. 罗列最近做过什么可能有影响的硬件、软件变更。

尽量想象一个黑客会怎样反问你，在提问的时候预先给他答案。

Simon Tatham 写过一篇名为《如何有效的报告 Bug》的出色短文。强力推荐你也读一读。

- 话不在多，在精

你需要提供精确有效的信息。这并不是要求你简单的把成吨的出错代码或者数据完全转储摘录到你的提问中。如果你有庞大而复杂的测试条件，尽量把它剪裁得越小越好。

这样做的用处至少有三点。第一，表现出你为简化问题付出了努力，这可以使你得到回答的机会增加；第二，简化问题使你得到有用答案的机会增加；第三，在提炼你的 bug 报告的过程中，也许你自己就能找出问题所在或作出更正。

只说症状，不说猜想

告诉黑客们你认为问题是怎样引起的没什么帮助。（如果你的推断如此有效，还用向别人求助吗？），因此要确信你原原本本告诉了他们问题的症状，不要加进你自己的理解和推论。让黑客们来诊断吧。

蠢问题：我在内核编译中一次又一次遇到 SIG11 错误，我怀疑某条飞线搭在主板的走线上了，这种情况应该怎样检查最好？

聪明问题：我自制的一套 K6/233 系统，主板是 FIC-PA2007（VIA Apollo VP2 芯片组），256MB Corsair PC133 SDRAM，在内核编译中频频产生 SIG11 错误，从开机 20 分钟以后就有这种情况，开机前 20 分钟内从没发生过。重启也没有用，但是关机一晚上就又能工作 20 分钟。所有内存都换过了，没有效果。相关部分的典型编译记录如下...

按时间顺序列出症状

对找出问题最有帮助的线索，往往就是问题发生前的一系列操作，因此，你的说明应该包含

操作步骤，以及电脑的反应，直到问题产生。

如果你的说明很长（超过四个段落），在开头简述问题会有所帮助，接下来按时间顺序详述。这样黑客们就知道该在你的说明中找什么。

- 明白你想问什么

漫无边际的提问近乎无休无止的时间黑洞。最能给你有用答案的人也正是最忙的人（他们忙是因为要亲自完成大部分工作）。这样的人对无节制的时间黑洞不太感冒，因此也可以说他们对漫无边际的提问不大感冒。

如果你明确表述需要回答者做什么（提供建议，发送一段代码，检查你的补丁或是别的），就最有可能得到有用的答案。这会定出一个时间和精力上限，便于回答者集中精力来帮你，这很凑效。要理解专家们生活的世界，要把专业技能想象为充裕的资源，而回复的时间则是贫乏的资源。解决你的问题需要的时间越少，越能从忙碌的专家口中掏出答案。

因此，优化问题的结构，尽量减少专家们解决它所需要的时间，会有很大的帮助--这通常和简化问题有所区别。因此，问“我想更好的理解 X，能给点提示吗？”通常比问“你能解释一下 X 吗？”更好。如果你的代码不能工作，问问它有什么地方不对，比要求别人替你修改要明智得多。

- 别问应该自己解决的问题

黑客们总是善于分辨哪些问题应该由你自己解决；因为我们中的大多数都曾自己解决这类问题。同样，这些问题得由你来搞定，你会从中学到东西。你可以要求给点提示，但别要求得到完整的解决方案。

- 去除无意义的疑问

别用无意义的话结束提问，例如“有人能帮我吗？”或者“有答案吗？”。首先：如果你对问题的描述不很合适，这样问更是画蛇添足。其次：由于这样问是画蛇添足，黑客们会很厌烦你--而且通常会用逻辑上正确的回答来表示他们的蔑视，例如：“没错，有人能帮你”或者“不，没答案”。

- 谦逊绝没有害处，而且常帮大忙

彬彬有礼，多用“请”和“先道个谢了”。让大家都知道你对他们花费时间义务提供帮助心存感激。然而，如果你有很多问题无法解决，礼貌将会增加你得到有用答案的机会。

（我们注意到，自从本指南发布后，从资深黑客处得到的唯一严重缺陷反馈，就是对预先道谢这一条。一些黑客觉得“先谢了”的言外之意是过后就不会再感谢任何人了。我们的建议是：都道谢。）

问题解决后，加个简短说明

问题解决后，向所有帮助过你的人发个说明，让他们知道问题是怎样解决的，并再一次向他们表示感谢。如果问题在新闻组或者邮件列表中引起了广泛关注，应该在那里贴一个补充说明。补充说明不必很长或是很深入；简单的一句“你好，原来是网线出了问题！谢谢大家--Bill”

比什么也不说要强。事实上，除非结论真的很有技术含量，否则简短可爱的小结比长篇学术论文更好。说明问题是怎样解决的，但大可不必将解决问题的过程复述一遍。除了表示礼貌和反馈信息以外，这种补充有助于他人在邮件列表/新闻组/论坛中搜索对你有过帮助的完整解决方案，这可能对他们也很有用。最后（至少？），这种补充有助于所有提供过帮助的人从中得到满足感。如果你自己不是老手或者黑客，那就相信我们，这种感觉对于那些你向他们求助的导师或者专家而言，是非常重要的。问题久拖未决会让人灰心；黑客们渴望看到问题被解决。好人有好报，满足他们的渴望，你会在下次贴出新问题时尝到甜头。

还是不懂

如果你不是很理解答案，别立刻要求对方解释。象你以前试着自己解决问题时那样（利用手册，FAQ，网络，身边的高手），去理解它。如果你真的需要对方解释，记得表现出你已经学到了点什么。比方说，如果我回答你：“看来似乎是 `zEntry` 被阻塞了；你应该先清除它。”，然后：一个很糟的后续问题：“`zEntry` 是什么？”聪明的问法应该是这样：“哦~~~我看过帮助了但是只有 `-z` 和 `-p` 两个参数中提到了 `zEntry` 而且还都没有清楚的解释:<你是指这两个中的哪一个吗？还是我看漏了什么？”(重点)

转：提问的智慧（三）

=三思而后问 =

以下是几个经典蠢问题，以及黑客在拒绝回答时的心中所想：

问题：我能在哪找到 X 程序？

问题：我的程序/配置/SQL 申明没有用

问题：我的 Windows 有问题，你能帮我吗？

问题：我在安装 Linux（或者 X）时有问题，你能帮我吗？

问题：我怎样才能破解 root 帐号/窃取 OP 特权/读别人的邮件呢？

提问：我能在哪找到 X 程序？

回答：就在我找到它的地方啊--搜索引擎的那一头。天呐！还有人不会用 Google 吗？

提问：我的程序（配置、SQL 申明）没有用

回答：这不算是问题吧，我对找出你的真正问题没兴趣--如果要我问你二十个问题才找得出的话--我有更有意思的事要做呢。

在看到这类问题的时候，我的反应通常不外如下三种：

1. 你还有什么要补充的吗？
2. 真糟糕，希望你能搞定。
3. 这跟我有什么鸟相关？

提问：我的 Windows 有问题，你能帮我吗？

回答：能啊，扔掉微软的垃圾，换 Linux 吧。

提问：我在安装 Linux（或者 X）时有问题，你能帮我吗？

回答：不能，我只有亲自在你的电脑上动手才能找到毛病。还是去找你当地的 Linux 用户组寻求手把手的指导吧（你能在这儿找到用户组的清单）。

提问：我怎样才能破解 root 帐号/窃取 OP 特权/读别人的邮件呢？

回答：想要这样做，说明你是个卑鄙小人；想找个黑客帮你，说明你是个白痴！

= 好问题，坏问题 =

最后，我举一些例子来说明，怎样聪明的提问；同一个问题的两种问法被放在一起，一种是愚蠢的，另一种才是明智的。

蠢问题：我可以在哪儿找到关于 Foonly Flurbamatic 的资料？

// 这种问法无非想得到“STFW”这样的回答。

聪明问题：我用 Google 搜索过“Foonly Flurbamatic 2600”，但是没找到有用的结果。谁知道上哪儿去找对这种设备编程的资料？

// 这个问题已经 STFW 过了，看起来他真的遇到了麻烦。

蠢问题：我从 FOO 项目找来的源码没法编译。它怎么这么烂？

// 他觉得都是别人的错，这个傲慢自大的家伙

聪明问题：FOO 项目代码在 Nulix 6.2 版下无法编译通过。我读过了 FAQ，但里面没有提到跟 Nulix 有关的问题。这是我编译过程的记录，我有什么做得不对的地方吗？

// 他讲明了环境，也读过了 FAQ，还指明了错误，并且他没有把问题的责任推到别人头上，这个家伙值得留意。

蠢问题：我的主板有问题了，谁来帮我？

// 普通黑客对这类问题的回答通常是：“好的，还要帮你拍拍背和换尿布吗？”，然后按下删除键。

聪明问题：我在 S2464 主板上试过了 X、Y 和 Z，但没什么作用，我又试了 A、B 和 C。请注意当我尝试 C 时的奇怪现象。显然边带传输中出现了收缩，但结果出人意料。在多处理器主板上引起边带泄漏的通常原因是什么？谁有好主意接下来我该做些什么测试才能找出问题？

// 这个家伙，从另一个角度来看，值得去回答他。他表现出解决问题的能力，而不是坐等天上掉答案。

在最后一个问题中，注意“告诉我答案”和“给我启示，指出我还应该做什么诊断工作”之间微妙而又重要的区别。事实上，后一个问题源自于 2001 年 8 月在 Linux 内核邮件列表上的一个真实的提问。我（Eric）就是那个提出问题的人。我在 Tyan S2464 主板上观察到了这种无法解释的锁定现象，列表成员们提供了解决那一问题的重要信息。

通过我的提问方法，我给大家值得玩味的东西；我让人们很容易参与并且被吸引进来。我显示了自己具备和他们同等的能力，邀请他们与我共同探讨。我告诉他们我所走过的弯路，以避免他们再浪费时间，这是一种对他人时间价值的尊重。后来，当我向每个人表示感谢，并且赞赏这套程序（指邮件列表中的讨论 --译者注）运作得非常出色的时候，一个 Linux 内核邮件列（lkm1）成员表示，问题得到解决并非由于我是这个列表中的“名人”，而是因为我用了正确的方式来提问。我们黑客从某种角度来说是有丰富知识但缺乏人情味的家伙；我相信他是正确的，如果我像个乞讨者那样提问，不论我是谁，一定会惹恼某些人或者被他们忽视。他建议我记下这件事，给编写这个指南的人一些指导。

= 找不到答案怎么办 =

如果仍得不到答案，请不要以为我们觉得无法帮助你。有时只是看到你问题的人不知道答案罢了。没有回应不代表你被忽视，虽然不可否认这种差别很难区分。

总的说来，简单的重复张贴问题是个很糟的想法。这将被视为无意义的喧闹。

你可以通过其它渠道获得帮助，这些渠道通常更适合初学者的需要。有许多网上的以及本地的用户组，由狂热的软件爱好者（即使他们可能从没亲自写过任何软件）组成。通常人们组建这样的团体来互相帮助并帮助新手。

另外，你可以向很多商业公司寻求帮助，不论公司大还是小（Red Hat 和 LinuxCare 就是两个最常见的例子）。别为要付费才能获得帮助而感到沮丧！毕竟，假使你的汽车发动机汽缸密封圈爆掉了--完全可能如此--你还得把它送到修车铺，并且为维修付费。就算软件没花费用你一分钱，你也不能强求技术支持总是免费的。

对大众化的软件，就象 Linux 之类而言，每个开发者至少会有上万名用户。根本不可能由一个人来处理来自上万名用户的求助电话。要知道，即使你要为帮助付费，同你必须购买同类软件相比，你所付出的也是微不足道的（通常封闭源代码软件的技术支持费用比开放源代码软件要高得多，且内容也不那么丰富）。

注释:FAQ 是英文 Frequently Asked Questions 的缩写，中文意思就是“经常问到的问题”，或者更通俗地叫做“常见问题解答”。在很多[网站](#)上都可以看到 FAQ，列出了一些[用户](#)常见的问题，是一种在线帮助形式。在利用一些网站的功能或者服务时往往会遇到一些看似很简单，但不经过说明可能很难搞清楚的问题，有时甚至会因为这些细节问题的影响而失去用户，其实在很多情况下，只要经过简单的解释就可以解决这些问题，这就是 FAQ 的价值。

在[网络营销](#)中，FAQ 被认为是一种常用的在线顾客服务手段，一个好的 FAQ 系统，应该至少可以回答用户 81% 的一般问题，这样不仅方便了用户，也大大减轻了

网站工作人员的压力，节省了大量的顾客服务成本，并且增加了顾客的满意度。因此，一个优秀的网站，应该重视 **FAQ** 的设计。