

教你如何构建自己的Linux发行版 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/145/2021_2022__E6_95_99_E4_BD_A0_E5_A6_82_E4_c103_145370.htm Linux From Scratch

(LFS) 及其后代代表一种新方法，向用户揭示 Linux 操作系统是如何工作的。LFS 基于这样的假设，即一部分一部分地编译完整的操作系统不仅能够揭示操作系统是如何工作的，也允许独立的操作人员为速度、内存占用或安全性而构建系统。许多作家已经编写了有关 UNIX 风格的书籍，深入研究了调度、内存管理、多进程和线程、文件系统，以及用户与内核之间的交互。写作 Linux 书籍的作家相对于 UNIX 作家来说有一个优势：尽管团体发生了剧变，但是 Linux 内核不可能分成几个相互竞争的分支，由于 GNU Public License (GPL)、集中式研究实验室 Open Source Development Lab (OSDL) 和 Linus Torvalds 不可动摇的地位，使得 Linux 有幸成为一个缓慢移动的目标(slow-moving target)。为什么 UNIX 内核很重要除了某些方面具有一定的相似性之外，不同的 Unix 内核并不怎么相同。各种 UNIX 风格也具有一个 Linux 所缺少的优点：所有 UNIX 风格都被假定是完全的操作系统。Linux 通常被描述为“只是一个内核”(如果有这样的定义的话，也是一个武断的定义)，它给出公共功能和实现的核心，不管内核是运行在不太强大的 Pentium II 机器上还是 Symmetric Multiprocessing (SMP) 系统上，这些公共功能和实现都不会发生本质上的改变。为了更加简化，有人可能会说，离 Linux 内核越远，就会发现更多的变化，而 UNIX 系统则趋向于是各种 UNIX/POSIX 标准的离散实现。事情并没有这么简单。

检测 Linux 内核和系统级代码可能是一件很费时间的事情，并且在现实世界中会限制使用。LFS 项目旨在解决 Linux 上有限的系统级可理解性问题。关于内核需要大量的库和工具来让 Linux 系统执行最基本的任务这一事实，已经做过讨论了，但是如果一个比较熟练的用户具有一个 slim-line Linux 发行版，他不想下载几吉字节不让他优化系统、也不让他抛开所有这些麻烦且不必要的工具的二进制代码，那该怎么办？如果一个非常熟练的用户拒绝接受各种社团发行版的苛刻条件(diktat)，而想要运行一个来自 CD 的 Linux/Apache/MySQL/PHP (LAMP) 类型的应用程序堆栈，那该怎么办？LFS 可以解决这些问题。Linux From Scratch LFS 项目显然建立于那些对于构成基本的 Linux 系统来说充分而不必要的源文件的基础之上。它已经超越了 Linux 内核和设备驱动程序，因为要产生一个可工作的 Linux 系统，您必须添加一个完整的编译器工具链、许多 Linux 汇编程序实用工具、glibc 系统库、系统配置工具和连接到 userland shell 访问的工具。LFS 建立在这样一个假设的基础之上，即 Linux 或 UNIX 允许具有一些脚本编写知识的用户，了解一个完全有用的系统的工作方式，而不用深入研究内核代码本身。为了了解 Linux 系统的工作方式，LFS 的创建者们确定，通过遵循模块依赖性来编译系统，可能是了解一般操作系统和特定的 Linux 的机制最自然的方式。用户掌握了编译过程之后，就可以开始消除依赖性树的那些连接到与支持操作系统基本目的无关的系统组件的部件。例如，在编译完成之后，消除编译器工具链本身是可行的。在没有全套的命令行实用工具时，可以凑合使用嵌入式 LAMP 堆栈。配置实用工具也可以被丢

弃，大多数用户可以凑合着用一個而不是太多 Linux 将会支持的文件系统。Linux 部件 LFS 系统的一个重要部件是，可以作为 tar ball 得到的大量源文件。文档是另一个重要部件，并且是最重要的。实际上，很有可能利用一个最新的 LFS 书籍文件并创建一个 LFS 发行版，因为 LFS 书籍中详细描述了每个下载位置和每个源文件及其依赖性的特征。用于从内核到编译器到 shell 编译每组源文件的过程都是已经写好了的，如果可能，您也可以描述具有不同特征的系统的 LFS 书籍中找到替代的例程。LFS 系统的另一个不太可能出现在一般用户工具箱中的部件是，在基本 LFS 系统组合在一起之后引导系统所需的引导脚本。现在对 LFS 发行版的最大警告是：勇敢的发行版构建者所需的是一个可工作的 Linux 发行版，包括一个完整的编译器工具链和一套文件系统创建实用工具。自然，所有基于源代码的 Linux 发行版都需要使用各个发行版都完全不同的特定编译器版本来引导。LFS 不是该领域的惟一系统，但它是惟一允许您直接处理单个源文件的系统。大多数其他基于源代码的 Linux 系统，比如 Sourcemage 和 MyGeOS，提供一个完整的下载，建议用户使用。LFS 不作这样的假设，并且鼓励拆开 LFS 框架。预先假设起作用的 Linux 发行版已安装在非外来的(nonexotic)硬件上，即使 LFS 可能没有配置工具和脚本那么受关注。要编译 LFS，您需要准备一个分区和一个文件系统，还需要编译一个编译器和系统库。如果用手工完成的话，这是一个相当伤脑筋的过程，但是也的确可以增加您在处理其余安装方面的自信。整个系统的编译要花一小时到四天的时间，具体时间取决于底层硬件的年代和您的命令行技术熟练程度。作一个相当大的假设，如果

您愿意很大程度上保留书籍安装，并使对 LFS 书籍中提议的安装的更改保持最小，您也可以使用自动化的安装例程来安装基于 LFS 的发行版。安装例程没有在 LFS 书籍中给出，但是可以在名称 Automated Linux from Scratch (ALFS) 之下作为基于 XML 的发行版得到。活动安装可以作为基于 C 的脚本得到，该脚本使用 ncurses 来模拟图形安装。该安装也叫做 nALFS 并给出一个极为灵活的包安装框架。工作需要一个起作用的 Linux 系统，并带有可以工作的 C 编译器和 XML 解析器。一个可以工作的 LFS 系统就足够了。Automated Linux From Scratch ALFS 的目的是超越 LFS 本身。LFS 自己指导基于 Linux 的操作系统内部工作，但是它不具有单个图形用户界面 (GUI)。LFS 既不允许连接到网络，也不允许连接到 Internet。ALFS 可以简化系统的扩展，例如，通过添加支持 Internet 访问的库，或者通过安装图形桌面所需的 X 库。超越 LFS LFS 的创建者们认识到了对其他各种基于源代码的 Linux 系统的需求。为了这些想要超越 LFS 并添加 X Window System、GNOME 和网络支持的人，创建了另一个 LFS 派生物：Beyond Linux From Scratch (BLFS)。第三版 LFS 书籍(让我们不会忘记是在谈论书籍而非发行版)，形成一个以一个角为基础的三角形：对于自动化编译和完全的基于源代码的 Linux 发行版，基本的 LFS 版本是基础。BLFS 将基本的 Linux 系统转化成一个完全的用户广泛接受的 Linux 系统。ALFS 简化基于源代码的 Linux 安装的安装和扩展。整个基于源代码的系统的编译是由一个脚本指导的，您在针对运行脚本的硬件调整脚本之后，就可以让脚本自己运行了。在您(或安装工程师)确定需要运行哪些包，也即特定的办公应用程序套件之后

，您就可以容易地扩展安装顺序了。ALFS 也迟早会派上用场，因为它适用于从源代码进行网络范围的安装。Hardened LFS 家族的最后一个成员解决基于源代码的 Linux 的一个特别重要的方面：安全性。对于不想依赖于所选的 Linux 发行版服务器交付的补丁的那些人，安全性的普通方法是，针对所选的核心库和应用程序跟踪安全报告。对于 LFS 实现人员来说，问题有些不同：尽管不是不可能，但是也难以审计 Linux 内核代码，也许还包括集中于基于 Linux 的操作系统内部机能的很多库和实用工具。代码审计相当费时，并且只有当补丁服务器是由专门人员集中维护的时，添加大量补丁才是明智的。但是可以取代一些为反映安全问题的新方法而重新编写过的库。一个好的例子是，通过从一个适当大的随机数池中随机分配数字，使得猜测进程标识符极为困难。OpenBSD 项目最先采用了该方法，随后，各种 UNIX 风格和 Linux 发行版都采用了该方法。一个相当新的项目叫做 Hardened Linux From Scratch (HLFS)，在 Linux 下采用了这一安全方法。该项目假设相当正规地掌握了 LFS 和 BLFS 的一些部件，并使用了不会成为大多数 Linux 系统中的标准的几个实用工具和库。添加到 HLFS 中的最重要的部件可能是 Stack-Smashing Protector (SSP)，通过使用 gcc 指令可以启用该功能。SSP 开发来防御 stack-smashing 攻击，这类攻击属于影响 Linux 系统的一类最常见的安全威胁。其他安全措施包括一流的随机数生成器和位置独立的可执行程序的编译，其中通常转化成静态链接对象代码的可执行代码作为共享库出现，而位置独立的可执行库通过将地址随机化可以隐藏这些地址。当然，可从 HLFS Web 站点得到大量补丁，并可查看其

源代码。日益壮大的 LFS 家族在许多方面，Linux 版本的 LFS 家族是一个方法，给予黑客以构造基于 Linux 的操作系统的能力。但是对于 LFS 的创建者们来说，最重要的结果似乎是：通过 LFS，所有 Linux 发行版对于所涉及的用户都成了智能的了。通过允许用户一部分一部分地构建 Linux 发行版，并帮助用户将一个基于 Linux 的操作系统看作一个由许多部件组成的系统，还可以用另外的方法构建 Linux 发行版。更一般来说，用户要更改构建 Linux 发行版的方式，不必要是程序员：只要从构建 LFS 系统学到的一点脚本编写能力就足够了。LFS 专家可以改变和扩展 Linux 发行版的完美组合，而不会影响它的基本结构。对于有人员和专家维护 Linux 系统，但没有资金从咨询机构和公司购买商业支持的机构来说，该功能特别重要。已经针对教学目的和大型网络，演示了基于 LFS 的 Linux 系统。它们也有可能用于其他领域。100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com