

Linux内核开发：使用Git管理源代码（4）PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/144/2021_2022_Linux_E5_86_85_E6_A0_c103_144959.htm 结束语 现在我们已经知道如何使用 Git 来获取 Linux 内核源代码和其他 Git 管理的项目了，接下来可以选择使用 Git 来管理下一个开发项目。Git 仍然相对较新，仍然处于不断开发中。其他脚本和工具正在实现用来简化 Git 的使用；请参看参考资料中给出的例子。非常简单地说，Git 是 Linus Torvalds 最近实现的源代码管理软件。正如所提供的文档中说的一样，“Git 是一个快速、可扩展的分布式版本控制系统，它具有极为丰富的命令集，对内部系统提供了高级操作和完全访问。”Torvalds 开始着手开发 Git 是为了作为一种过渡方案来替代 BitKeeper，后者之前一直是 Linux 内核开发人员在全球使用的主要源代码工具。开放源码社区中的有些人觉得 BitKeeper 的许可证并不适合开放源码社区的工作，因此 Torvalds 决定着手研究许可证更为灵活的版本控制系统。尽管最初 Git 的开发是为了辅助 Linux 内核开发的过程，但是我们已经发现在很多其他自由软件项目中也使用了 Git。例如，X.org 最近就迁移到 Git 上来了，很多 Freedesktop.org 的项目也迁移到了 Git 上。Git 目前主要由寻找 CVS 或专有代码管理解决方案替代物的软件开发人员所使用。Git 与 CVS 有很多区别：分支更快、更容易。支持离线工作；本地提交可以稍后提交到服务器上。Git 提交都是原子的，且是整个项目范围的，而不像 CVS 中一样是对每个文件的。Git 中的每个工作树都包含一个具有完整项目历史的仓库。没有哪一个 Git 仓库会天生比其他仓库更重要。安装

要安装当前版本的 Git，我们可以使用供应商在 Linux 发行版中提供的包，或者从最新的稳定快照开始手工进行编译。我建议下载包含最新 Git 源代码稳定快照的 tarball；截止到撰写本文时这个版本是 v1.4.0。我们可以在下面的参考资料一节中找到链接。有了这个 tarball 之后，请确保初始安装所依赖的包都已经安装了。系统中必须包含以下包，以及相应的开发头文件：zlib libcurl libcrypto (OpenSSL) rsync (2.6.0 或更高版本) 这些条件满足之后，我们就可以开始编译初始的 Git 安装系统了。这个过程对于大部分一直使用 Linux 的开发人员来说应该非常熟悉了。首先使用对应下载的包格式的命令将包展开：`$ tar -jxvf git-1.4.0.tar.bz2`或`$ tar -zxvf git-1.4.0.tar.gz`然后切换到适当的目录中，并执行 make 命令。（注意目录名取决于我们下载的快照的日期。）`$ cd git-1.4.0/$ make prefix=/usr/local install``$ sudo make prefix=/usr/local install`您会被提示输入 sudo 密码才能继续安装。现在就已经准备好使用 Git 工具了。获得最新的内核源代码树在使用 Git 管理源代码仓库时，我们可以使用两种方法开始我们的工作。我们可以使用现有代码的一个本地目录，然后从中生成一个仓库；也可以映射其他人发布的仓库。对于本文的目的来说，我们将获得 Torvalds 发布的 Git 仓库的一个镜像。下面的命令将创建一个名为 linux-2.6 的 Git 仓库。这个目录包含了一个隐藏目录 .git/。`$ git-clone \`
`git://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux-2.6.git`
`\linux-2.6`这个步骤会执行很长时间，因为 Git 正在将内核源代码（这有数百兆大）从 kernel.org 传输到本地机器上。输出结果可能有些晦涩难懂，但是如果您有一个快速的 Internet 链接

，卷屏的速度应该相当快。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