



首页

最新资讯

开源黄页

我的空间

开源服务

优麒麟/CCN

开源联盟

关于我们

内容导航菜单



标题

搜索

热门关键词: HTML5 陆首群 Firefox 袁萌

你的位置: 开源中国网 &gt;&gt; 最新资讯 &gt;&gt; 技术应用 &gt;&gt; 详细内容

在线投稿

技术应用

## Ubuntu命令行下安装、卸载、管理软件包的方法

来源: 开源中国 发布者: chenyl

时间: 2012年3月05日 13:23



排行榜

收藏

打印

发给朋友

举报

热度298票 浏览10525次【共1条评论】【我要评论(不用登录)】

## 一、Ubuntu中软件安装方法

## 1、APT方式

- (1)普通安装: `apt-get install softname1 softname2 ...;`
- (2)修复安装: `apt-get -f install softname1 softname2... ;(-f Atempt to correct broken dependencies)`
- (3)重新安装: `apt-get --reinstall install softname1 softname2...;`

## 2、Dpkg方式

- (1)普通安装: `dpkg -i package_name.deb`

## 3、源码安装(.tar、tar.gz、tar.bz2、tar.Z)

首先解压缩源码压缩包然后通过tar命令来完成

- 解xx.tar.gz: `tar xzf xx.tar.gz`
- 解xx.tar.Z: `tar xzf xx.tar.Z`
- 解xx.tgz: `tar xzf xx.tgz`
- 解xx.bz2: `bunzip2 xx.bz2`
- 解xx.tar: `tar xf xx.tar`

然后进入到解压出的目录中, 建议先读一下README之类的说明文件, 因为此时不同源代码包或者预编译包可能存在差异, 然后建议使用 `ls -F --color` 或者 `ls -F` 命令 (实际上我的只需要 `ls` 命令即可) 查看一下可执行文件, 可执行文件会以\*号的尾部标志。

一般依次执行 `./configure``make``sudo make install`

即可完成安装。

## 二、Ubuntu中软件包的卸载方法

## 1、APT方式

- (1)移除式卸载: `apt-get remove softname1 softname2 ...;` (移除软件包, 当包尾部有+时, 意为安装)
- (2)清除式卸载: `apt-get --purge remove softname1 softname2...;` (同时清除配置)  
清除式卸载: `apt-get purge sofname1 softname2...;` (同上, 也清除配置文件)

## 2、Dpkg方式

- (1)移除式卸载: `dpkg -r pkg1 pkg2 ...;`
- (2)清除式卸载: `dpkg -P pkg1 pkg2...;`

## 三、Ubuntu中软件包的查询方法

Dpkg 使用文本文件来作为数据库. 通称在 `/var/lib/dpkg` 目录下. 通称在 `status` 文件中存储软件状态, 和控制信息. 在 `info/` 目录下备份控制文件, 并在其下的 `.list` 文件中记录安装文件清单, 其下的 `.mdasums` 保存文件的 MD5 编码.

体验使用数据库的时刻到了:

`$ dpkg -l``Desired=Unknown/Install/Remove/Purge/Hold``| Status=Not/Installed/Config-files/Unpacked/Failed-config/Half-installed``|/ Err?=(none)/Hold/Reinst-required/X=both-problems (Status,Err: uppercase=bad)`

最新资讯

## 龙芯28nm芯片已经进入实操阶段

去年年底, 龙芯推出了使用新架构GS46E的3A-1500四核处理器, 随后该处理器的性能测试也曝光出来, 这款国产芯也是在部分性能(Whetstone浮点运算)甚至比i78...[【全文】](#)

- ✦ Ubuntu MX4 今日将在魅族官网在
- ✦ 宏碁携手Ubuntu共抢物联网商机
- ✦ Ubuntu或抢先Windows 10实现手机
- ✦ Firefox for Android下载超过一
- ✦ 优麒麟 15.04 各城市发布派对活
- ✦ 化蛹成蝶的蜕变 红旗Linux满血复

精华推荐

- ✦ 化蛹成蝶的蜕变 红旗Linux满血复
- ✦ Windows开源: 微软真实的"谎
- ✦ 让你玩转 Ubuntu 桌面的十一件武
- ✦ Ubuntu将逐步放弃deb包格式, 默认
- ✦ 开源软件在商业世界中的十大最佳
- ✦ 华为推出体积仅10KB的物联网操作
- ✦ 【2014】GitHub中国开发者年度报
- ✦ 开源势力正在扩大的五大标志
- ✦ 从GitHub上看开源许可证多舛的命
- ✦ GNU 宣言发表三十周年

月度热点

- ✦ 开源技术手册黄页
- ✦ 开源外卖平台(Web, 微信, APP)
- ✦ 利用 Bootstrap 进行快速 Web 开
- ✦ Ubuntu将逐步放弃deb包格式, 默认
- ✦ 优麒麟(Ubuntu Kylin)15.04发布
- ✦ Firefox 38 正式版用户特性速览
- ✦ 华为推出体积仅10KB的物联网操作
- ✦ Ubuntu MX4 今日将在魅族官网在
- ✦ 一大波你可能不知道的 Linux 网
- ✦ 龙芯28nm芯片已经进入实操阶段

```

||/ Name      Version      Description
+++-----
=====
ii aalib1     1.4p5-28      ascii art library - transitional package
ii adduser    3.85          Add and remove users and groups
ii alien      .63           install non-native packages with dpkg
... ..

```

每条记录对应一个软件包, 注意每条记录的第一, 二, 三个字符. 这就是软件包的状态标识, 后边依此是软件包名称, 版本号, 和简单描述.

第一字符为期望值,它包括:

u 状态未知,这意味着软件包未安装,并且用户也未发出安装请求.

i 用户请求安装软件包.

r 用户请求卸载软件包.

p 用户请求清除软件包.

h 用户请求保持软件包版本锁定.

第二列,是软件包的当前状态.此列包括软件包的六种状态.

n 软件包未安装.

i 软件包安装并完成配置.

c 软件包以前安装过,现在删除了,但是它的配置文件还留在系统中.

u 软件包被解包,但还未配置.

f 试图配置软件包,但是失败了.

h 软件包安装,但是但是没有成功.

第三列标识错误状态,可以总结为四种状态. 第一种状态标识没有问题,为空. 其它三种符号则标识相应问题.

h 软件包被强制保持,因为有其它软件包依赖需求,无法升级.

r 软件包被破坏,可能需要重新安装才能正常使用(包括删除).

x 软件包被破坏,并且被强制保持.

也可以以通配符模式进行模糊查询, 比如我要查找以nano字符开始的所有软件包:

```
$ dpkg -l nano*
```

```
Desired=Unknown/Install/Remove/Purge/Hold
```

```
| Status=Not/Installed/Config-files/Unpacked/Failed-config/Half-installed
```

```
|/ Err?=(none)/Hold/Reinst-required/X=both-problems (Status,Err: uppercase=bad)
```

```
||/ Name      Version      Description
```

```

+++-----
=====
ii nano       1.3.10-2     free Pico clone with some new features
pn nano-tiny  <none>       (no description available)

```

```
un nanoblogger <none>       (no description available)
```

以上状态说明: 系统中安装了 nano 版本为 1.3.10-2 ;安装过 nano-tiny , 后来又清除了; 从未安装过 nanoblogger .

如果觉得 dpkg 的参数过多, 不利于记忆的话, 完全可以使用 dpkg-query 进行 dpkg 数据库查询.

应用范例:

查询系统中属于nano的文件:

```
$ dpkg --getfiles nano
```

or

```
$ dpkg-query -W -f='${Package} ${Version} ${Architecture}\n'
```

查看软件nano的详细信息:

```
$ dpkg-query -W -f='${Package} ${Version} ${Architecture}\n'
```

or

```
$ dpkg-query -s nano
```

查看系统中软件包状态, 支持模糊查询:

```
$ dpkg -l
```

or

```
$dpkg-query -l
```

查看某个文件的归属包:

```
$ dpkg-query -S nano
```

or

```
$ dpkg -S nano
```

三、其他应用总结

apt-cache search # -----(package 搜索包)

apt-cache show #----- (package 获取包的相关信息, 如说明、大小、版本等)

apt-get install # -----(package 安装包)

apt-get install # -----(package --reinstall 重新安装包)

apt-get -f install # -----(强制安装, "-f = --fix-missing"当是修复安装吧...)

apt-get remove #----- (package 删除包)

apt-get remove --purge # -----(package 删除包, 包括删除配置文件等)

apt-get autoremove --purge # -----(package 删除包及其依赖的软件包+配置文件等(只对6.10有效, 强烈推荐))

apt-get update #-----更新源

apt-get upgrade #-----更新已安装的包

apt-get dist-upgrade # -----升级系统

apt-get dselect-upgrade #-----使用 dselect 升级

apt-cache depends #----- (package 了解使用依赖)

apt-cache rdepends # ----- (package 了解某个具体的依赖, 当是查看该包被哪些包依赖吧...)

apt-get build-dep # ----- (package 安装相关的编译环境)

apt-get source #----- (package 下载该包的源代码)

apt-get clean && apt-get autoclean # -----清理下载文件的存档 && 只清理过时的包

apt-get check #-----检查是否有损坏的依赖

dpkg -S filename -----查找filename属于哪个软件包

apt-file search filename -----查找filename属于哪个软件包

apt-file list packagename -----列出软件包的内容

apt-file update --更新apt-file的数据库

dpkg --info "软件包名" --列出软件包解包后的包名称.

dpkg -l --列出当前系统中所有的包.可以和参数less一起使用在分屏查看. (类似于rpm -qa)

dpkg -l |grep -i "软件包名" --查看系统中与"软件包名"相关联的包.

dpkg -s 查询已安装的包的详细信息.

dpkg -L 查询系统中已安装的软件包所安装的位置. (类似于rpm -ql)

dpkg -S 查询系统中某个文件属于哪个软件包. (类似于rpm -qf)

dpkg -I 查询deb包的详细信息, 在一个软件包下载到本地之后看看用不用安装(看一下呗).

dpkg -i 手动安装软件包(这个命令并不能解决软件包之前的依赖性问题),如果在安装某一个软件包的时候遇到了软件依赖的问题,可以用apt-get -f install在解决依赖性这个问题.

dpkg -r 卸载软件包.不是完全的卸载,它的配置文件还存在.

dpkg -P 全部卸载(但是还是不能解决软件包的依赖性的问题)

dpkg -reconfigure 重新配置

apt-get install

下载软件包, 以及所有依赖的包, 同时进行包的安装或升级.如果某个包被设置了 hold (停止标志, 就会被搁在一边(即不会被升级)。更多 hold 细节请看下面。

apt-get remove [--purge]

移除 以及任何依赖这个包的其它包。

--purge 指明这个包应该被完全清除 (purged), 更多信息请看 dpkg -P。

## apt-get update

升级来自 Debian 镜像的包列表, 如果你想安装当天的任何软件, 至少每天运行一次, 而且每次修改了 /etc/apt/sources.list 後, 必须执行。

## apt-get upgrade [-u]

升级所有已经安装的包为最新可用版本。不会安装新的或移除老的包。如果一个包改变了依赖关系而需要安装一个新的包, 那么它将不会被升级, 而是标志为 hold。apt-get update 不会升级被标志为 hold 的包 (这个也就是 hold 的意思)。请看下文如何手动设置包为 hold。我建议同时使用 '-u' 选项, 因为这样你就能看到哪些包将会被升级。

## apt-get dist-upgrade [-u]

和 apt-get upgrade 类似, 除了 dist-upgrade 会安装和移除包来满足依赖关系。因此具有一定的危险性。

## apt-cache search

在软件包名称和描述中, 搜索包含xxx的软件包。

## apt-cache show

显示某个软件包的完整的描述。

## apt-cache showpkg

显示软件包更多细节, 以及和其它包的关系。

## dselect

### console-apt

### aptitude

### gnome-apt

APT 的几个图形前端(其中一些在使用前得先安装)。这里 dselect 无疑是最强大的, 也是最古老, 最难驾驭。

## 普通 Dpkg 用法

### dpkg -i

安装一个 Debian 包文件, 如你手动下载的文件。

### dpkg -c

列出 的内容。

### dpkg -I

从 中提取包信息。

### dpkg -r

移除一个已安装的包。

### dpkg -P

完全清除一个已安装的包。和 remove 不同的是, remove 只是删掉数据和可执行文件, purge 另外还删除所有的配制文件。

### dpkg -L

列出 安装的所有文件清单。同时请看 dpkg -c 来检查一个 .deb 文件的内容。

### dpkg -s

显示已安装包的信息。同时请看 apt-cache 显示 Debian 存档中的包信息, 以及 dpkg -I 来显示从一个 .deb 文件中提取的包信息。

## dpkg-reconfigure

重新配制一个已经安装的包, 如果它使用的是 debconf (debconf 为包安装提供了一个统一的配制界面)。你能够重新配制 debconf 它本身, 如你想改变它的前端或提问的优先权。例如, 重新配制 debconf, 使用一个

dialog 前端, 简单运行:

`dpkg-reconfigure --frontend=dialog debconf` (如果你安装时选错了, 这里可以改回来哟:)

```
echo " hold" | dpkg --set-selections
```

设置 的状态为 hold (命令行方式)

```
dpkg --get-selections ""
```

取的 的当前状态 (命令行方式)

支持通配符, 如:

```
Debian:~# dpkg --get-selections *wine*
```

```
libwine hold
```

```
libwine-alsa hold
```

```
libwine-arts hold
```

```
libwine-dev hold
```

```
libwine-nas hold
```

```
libwine-print hold
```

```
libwine-twain hold
```

```
wine hold
```

```
wine+ hold
```

```
wine-doc hold
```

```
wine-utils hold
```

例如:

大家现在用的都是 gaim-0.58 + QQ-plugin, 为了防止 gaim 被升级, 我们可以采用如下方法:

方法一:

```
Debian:~# echo "gaim hold" | dpkg --set-selections
```

然後用下面命令检查一下:

```
Debian:~# dpkg --get-selections "gaim"
```

```
gaim hold
```

现在的状态标志是 hold, 就不能被升级了。

如果想恢复怎么办呢?

```
Debian:~# echo "gaim install" | dpkg --set-selections
```

```
Debian:~# dpkg --get-selections "gaim"
```

```
gaim install
```

这时状态标志又被重置为 install, 可以继续升级了。

同志们会问, 哪个这些状态标志都写在哪个文件中呢?

在 `/var/lib/dpkg/status` 里, 你也可以通过修改这个文件实现 hold。

有时你会发现有的软件状态标志是 purge, 不要奇怪。

如: 事先已经安装了 amsn, 然後把它卸了。

```
apt-get remove --purge amsn
```

那么状态标志就从 install 变成 purge。

方法二:

在 `/etc/apt` 下手动建一个 preferences 文件

内容:

```
Package: gaim
```

```
Pin: version 0.58*
```

保存

dpkg -S

在包数据库中查找，并告诉你哪个包包含了这个文件。(注:查找的是事先已经安装的包)

-----  
Debian的软件包管理工具命令不完全列表  
-----

Debian系统中所有的包信息都在/var/lib/dpkg下.其中/var/lib/dpkg/info目录中保存了各个软件包的信息及管理文件.每个文件的作用如下:

以 ".conf files" 结尾的文件记录软件包的配置列表.

以 ".list" 结尾的文件记录了软件包的文件列表,用户可在文件当中找到软件包文件的具体安装位置.

以 ".md5sums" 结尾的文件记录了md5信息,用来进行包的验证的.

以 ".config" 结尾的文件是软件包的安装配置脚本.

以 ".postinst" 脚本是完成Debian解包之后的配置工作,通常用来执行所安装软件包相关的命令和服务的重新启动.

以 ".preinst" 脚本在Debian解包之前运行,主要作用是停止作用于即将升级的软件包服务直到软件包安装或升级完成.

以 ".prerm" 脚本负责停止与软件包关联的daemon服务,在删除软件包关联文件之前执行.

以 ".postrm" 脚本负责修改软件包链接或文件关联,或删除由它创建的文件.

/var/lib/dpkg/available是软件包的描述信息.

包括当前系统中所有使用的Debian安装源中所有的软件包,还包括当前系统中已经安装和未安装软件包.

## 1. dpkg包管理工具

dpkg -r 卸载软件包.不是完全的卸载,它的配置文件还存在.

dpkg --info "软件包名" --列出软件包解包后的包名称.

dpkg -l --列出当前系统中所有的包.可以和参数less一起使用在分屏查看.

dpkg -l |grep -i "软件包名" --查看系统中与"软件包名"相关联的包.

dpkg -s 查询已安装的包的详细信息. dpkg -L 查询系统中已安装软件包所安装的位置.

dpkg -S 查询系统中某个文件属于哪个软件包.

dpkg -I 查询deb包的详细信息,在一个软件包下载到本地之后看看用不用安装(看一下呗).

dpkg -i 手动安装软件包(这个命令并不能解决软件包之前的依赖性问题),如果在安装某一个软件包的时候遇到了软件依赖的问题,可以用apt-get -f install在解决依赖性这个问题.

dpkg -reconfigure 重新配置

dpkg -P 全部卸载(但是还是不能解决软件包的依赖性的问题)

## 2. apt高级包管理工具

(1)**GTK**图形的"synaptic",这是APT的前端工具.

(2)"aptitude",这也是APT的前端工具.

用APT管理工具进行包的管理,可以有以下几种方法做源:

(1)拿安装盘做源,方法如下:

apt-cdrom ident 扫描光盘的信息

apt-cdrom add 添加光盘源

(2)这也是最常用的方法就是把源添加到/etc/apt/source.list中,之后更新列apt-get update

## APT管理工具常用命令

apt-cache 加上不同的子命令和参数的使用可以实现查找,显示软件,包信息及包依赖关系等功能.

apt-cache stats 显示当前系统所有使用的Debian数据源的统计信息.

apt-cache search +"包名",可以查找相关的软件包.

apt-cache show +"包名",可以显示指定软件包的详细信息.

apt-cache depends +"包名",可以查找软件包的依赖关系.

apt-get upgrade 更新系统中所有的包到最新版

apt-get install 安装软件包

apt-get --reinstall install 重新安装软件包

apt-get remove 卸载软件包

apt-get --purge remove 完全卸载软件包

apt-get clean 清除无用的软件包

在用命令apt-get install之前,是先将软件包下载到/var/cache/apt/archives中,之后再进行安装的.所以我们可以用apt-get clean清除/var/cache/apt/archives目录中的软件包.

#### 源码包安装

apt-cache showsrc 查找看源码包的文件信息(在下载之前)

apt-get source 下载源码包.

apt-get build-dep +"包名" 构建源码包的编译环境.

#### 清除处于rc状态的软件包

dpkg -l |grep ^rc|awk '{print \$2}' |tr ["\n"] [" "] | sudo xargs dpkg -P -

上一篇: [Linux可插拔认证模块的基本概念](#)

下一篇: [一大波你可能不知道的 Linux 网](#)



【已经有242人表态】



#### 最新评论

[查看全部回复](#) 共1条评论

开源中国网湖南省衡阳市电信网友 发表于19分钟前 IP: 220.169.\*.\*

[引用](#) [删除](#)

总感觉作者没有仔细检查文章内容, 有很多重复的。。。不过看完后还是有所收获的。。。

#### 我来说两句


我也来评论! (不登录访客也能评论了)

验证码:  [换一张](#) [发表](#)

[关于社区](#) | [COPU](#) | [站点地图](#) | [友情链接](#) | [联系我们](#) | [@社区微博](#)

2011 中国开源软件推进联盟(COPU) All Rights Reserved 版权所有

E-Mail: [oss@csip.org.cn](mailto:oss@csip.org.cn) 联系电话: 010-63951881-8516 备案: 京ICP备09035011号

 中国开源软件  
推进联盟