

aptitude 简明手册

eTony etony@tom.com

dec 15, 2005

摘要

本文档是面向初学者的一个简短的指南, aptitude 是用于Debian GNU/Linux 软件包操作的一个控制台前端. 希望本文对进入Debian GNU/Linux 殿堂的初学者有所裨益.

版权声明

在遵守[GNU 公共许可证, Version 2](#) 或自由软件基金会发布的更新版本的条款下, 授权复制, 发布以及/或者修改本文档. 本文档的发布寄予有用的期望, 但不做任何保证.

在遵守并包含本文档版权声明的前提下, 允许制作和发布本文档的完整拷贝.

目录

1	概述	3
2	定义与介绍	3
3	快速开始	5
4	过滤与查询	6
5	软件包的选择	7
6	受损的软件包	7
7	清洁您的系统	8
8	虚拟包	9
9	任务的概念	9
10	强大的命令行	10
11	操作日志	11
12	有关这篇HOWTO的一些资料	11

1 概述

aptitude 是Debian GNU/Linux 系统中, 非常神奇的软件包管理器, 基于大名鼎鼎的APT 机制, 整合了dselect 和apt-get¹ 的所有功能, 并提供的更多特性, 特别是在依赖关系处理上. 本文档并不是一个全面的参考手册, 只是向新用户展示Debian 系统下的aptitude 是如何的强大, 并希望能使用户快速的掌握一些有关aptitude 的实用技巧. 更多信息请参阅作者提供的用户手册².

2 定义与介绍

要使用aptitude, 只要打开您喜爱的终端, 键入如下命令即可³:

```
# aptitude4
```

aptitude 的主视图组织的非常有条理, 而且界面布局清爽. 上部窗口列出的是软件包根据类别和它们在本地系统的状态排序. 这里是aptitude 的操作界面. 底部窗口给出主窗口的状态描述信息.

```
Actions Undo Package Search Options Views Help
f10: Menu ?: Help q: Quit u: Update g: Download/Install/Remove Pkgs
aptitude 0.2.15.9 Will use 25.9MB of disk space DL Size: 27.7MB
— Installed Packages
— Not Installed Packages
— Obsolete and Locally Created Packages
— Virtual Packages
— Tasks
```

在aptitude主菜单提供了七个选项, 这依赖于您系统中选用的软件包的状态.

1. Updated Packages(升级的软件包)

此选项列出在归档库中出先新版本的软件包. 通常, 当用户完成选择处理, aptitude 下载安装了请求的软件包后, 这里的软件包都将被升级.

2. New Packages (新软件包)

¹参阅APT-HOWTO

²/usr/share/aptitude/README

³可以是以非root用户直接运行, 但是如果你要对软件包进行操作的话, 则需要管理员权限

⁴本文档基于您对于Linux 以及Debian 有一定的了解

更新 (aptitude update,或在aptitude 中键入[u]) 软件包列表后所增加的软件包,让用户了解Debian 归档库增加了哪些内容.每次更新,新的软件包就会累加到这里,直至您键入[f]将累加清除,并将它们集合到主包库中.

3. Installed Packages(已安装软件包)

这些软件包已经安装在您的机器上,但资源库软件包列表显示不需要对它们进行升级.

4. Not Installed Packages(尚未安装软件包)

就象名称标识的,这些软件包当前没有安装在您的机器上.当您清除未安装的新软件包时,将会进入这部分.

5. Obsolete or Locally Created Packages(废弃或本地创建的软件包)

这些软件包目前已安装在您的机器上,但是它们并未出现在/etc/apt/sources.list 为APT 配置的资源库中.

6. Virtual Packages(虚拟包)

虚拟包是由一些软件包提供的逻辑概念.例如,mail-transport-agent 就是由postfix 和sendmail, 以及其它等等软件包提供的.

7. Tasks(任务)

任务是一组软件包,它们提供了一种简单的方法来选择一组预定义的完成特定任务的软件包.在后面将对此做更进一步的讲解.aptitude 允许对任务进行浏览和安装.

aptitude 是以树状结构组织的,相关节点以细线相连.可以使用方向键,[PageUp] 和[PageDown] 进行定位.[Return] 或[Enter] 用于展开或收起某个节点.aptitude 的开视图分为四列: 软件包状态和请求动作, 软件包名称, 当前安装版本(或<none>), 最新版本.

aptitude软件包列表中的软件包状态:

v	虚拟
B	损坏
u	解包
C	预配置
H	预安装
c	卸载未清除
p	清除软件包
i	已经安装
E	内部错误

在aptitude软件包列表中的请求操作:

h	保持
p	清除
d	删除(卸载)
B	损坏
i	安装
r	重装
u	升级

底部窗口给出主窗口当前显示的上下文信息. 可以用[a]和[z]实现滚屏,[i]在不同信息视图间切换. 最后,[D]则用于显示和隐藏信息区.

下边是默认安装时⁵相关背景颜色说明⁶:

颜色	意义
红	软件包受损(不能满足依赖关系, 或者出现冲突)
绿	安装新软件包
青	将软件包升级的最新
品红	删除(或清除)软件包
白	保存版本(禁止升级)

3 快速开始

对于任何程序来说, 您首先知道的就是如何退出— 在aptitude的主视图中键入[q](*quit*), 即可退出aptitude. 任何时候您键入[?](*help*), 都可以唤出帮助信息.

aptitude是基于apt机制的, 因此在进一步操作前你需要正确的配置sources.list⁷文件. 并确保软件包列表得到了及时的更新. [u](*update*)命令就是为此设计的, 这类似于*apt-get update* 操作.

如果出现了新的软件包(也就是说, 至上次软件包列表更新时, 仍为出现的软件包), 将会在*New Packages* 选项下显示. 您可以键入[f](*forget that packages are new*)命令, 将它们清除到可用软件包选项中去.

移动光标到您感兴趣的行, 然后键入回车, 将选项展开. 选定一个软件包后, 有关软件包的信息将会出现在屏幕底部. 如果要查看更详细的信息, 则需要键入[i](*information*). 键入[q] (*quit*)则会退出信息显示屏幕.

⁵当然, 您也可以通过配置文件修改相关显示颜色

⁶您的终端应当开启了色彩支持

⁷/etc/apt/sources.list

对于没有安装的软件包(位于 *New Packages* 或 *Not Installed Packages* 区域), 可以选定后, 键入 `[+]` (*add package*), 以标示安装.

对于已经安装的软件包, 则可以进行保持, 删除, 或清除等操作. 保持 `[=]` (*hold*) 是指保持软件包的现有版本, 而禁止对其升级. 删除 `[-]` (*remove package*) 标记已安装软件包为删除状态. `[_]` (*purge package*) 命令则将软件包标记为清除状态.

一旦您标记完感兴趣的软件包, 就可以键入 `[g]` (*go*) 以保存修改, 使其生效.

4 过滤与查询

键入 `[/]`, 并提供一些判断语句, 来查询软件包的名称. `[^]` 用于查询下一个匹配. 就象正则表达式一样, 可以用 `~` 和 `$` 匹配名称的起始字符和结束字符. 使用 `[1]`, 用户可以以某种标准限制软件包的显示, 这同样适用于查询⁸. 下边是一些重要的判断谓词:

aptitude 的搜索谓词

谓词	用途
<code>~ahold</code>	保持现有版本的软件包
<code>~b</code>	损坏的软件包
<code>~d<text></code>	描述中含有 <code><text></code> 内容的软件包
<code>~g</code>	无用的软件包
<code>~m<maint></code>	由 <code><maint></code> 维护的软件包
<code>~n<text></code>	名称中含有 <code><text></code> 的软件包
<code>~V<version></code>	版本号中含有 <code><version></code> 的软件包

这些谓词是可以联合起来使用的. 例如, `~ahold~dmail` 选择在描述字中含有 *mail* 的被保持现有版本的软件包. 在两个谓词间使用管道符号 (`|`), 可以对两个表达式进行逻辑或 (*OR*) 操作: `~v|~b` 选择所有受损软件包或虚拟软件包. 查询不会忽略两个谓词间的空格! 因此, `~V.0` 和 `~V .0` 是不一样的, 后边的一个不会有匹配结果. 感叹号对表达式取反: `!~b` 查询所有未受损的软件包. 可以使用括号来表达复杂的布尔逻辑: `~g(~dmanual|~sdoc)` 将查询所有包含 *manual* 或在 *doc* 部分的受损软件包.

需要注意的是, 当搜寻正则表达式中含有某些特殊字符, 象 `'()`, 代字号 (`~`), 和感叹号时, 则必须使用代字号进行转义.

⁸mutt 用户对于这些一定不陌生.

5 软件包的选择

软件包的选择可以通过浏览,然后按下对应键进行操作:

[+]	选定要安装的软件包.
[-]	选定要删除的软件包.
[_]	选定要清除的软件包.
[=]	保持软件包的当前版本,阻止其被升级.
[:]	仅在aptitude会话期间锁定软件包.
[L]	请求重装软件包.
[M]	将软件包标记为自动安装.自动安装的软件包 在手动安装的包对其没有依赖需求时会自动删除.
[m]	将软件包标记为手动.
[R]	请求重新配置软件包.
[I]	请求立即安装软件包(以其依赖包)并暂时锁定其它升级和 安装的软件包.作用与apt-get install <package>相同.
[F]	禁止安装某个版本的软件包.但是,对更高版本正常使用.
[B]	调用reportbug,申报一个软件包的错误.
[C]	下载并显示一个软件包的变更日志.
[g]	进入所有请求变更的预览窗口.如果在预览屏键入此键, 将执行变更操作.
[d]	查询相关软件包: suggest/recommanded/depends
[r]	查询依赖包
[/]	搜索
[N]	继续搜索

注: Control-u组合键可用于取消上一步的动作.

展开一个软件包的节点,将会在屏幕上输出软件包的详细信息.除了有关软件包的有用信息外,在发现aptitude给出的解决方案不能满足您的需求时,可以在屏幕中方便的浏览相关声明并交互的解决依赖问题.还可以使用[Return]展开节点,在主窗口用于操作软件包选择的按键也可在遵循相关声明的情况下,应用于这里列出的软件包.键入[q]令您回退一级,关闭软件包详述窗口,返回软件包列表.

6 受损的软件包

当aptitude遇到不能解决依赖问题时,会以红色背景高亮显示有问题的软件包,并显示标题窗口中错误数量.

并会在标题窗口中显示受损包的数量,例如: #Broken: 2

```
Actions Undo Package Search Options Views Help
f10: Menu ? : Help q : Quit u : Update g : Download/I
aptitude 0.2.15.9 #Broken: 2 Will use 1118kB of
--- Packages being automatically installed to satisfy
--\ Packages being deleted due to unsatisfied depende
id postfix -1974kB 2.1.5-9
iB postfix-mysql 2.1.5-9
iB postfix-tls 2.1.5-9
i webmin-postfix 1.180-4
--\ Packages to be installed
pi exim4 +32.8kB <none>
--- Packages which are suggested by other packages
```

受损的软件包是指那些不能满足依赖关系的,或相互冲突的软件包(这种情况下两个软件包都是*broken* 的). aptitude 显然不会允许用户选择安装*broken* 的软件包. 因此,需要首先修正这种问题:

当*broken* 软件包的个数非零,并且aptitude 被告知动作未定时(用户键入了[g] 键),在显示这些未决问题摘要之前aptitude将试图解决所有的问题.通常,使用保守的方法进行自动修正,所以前面的所做的选择有可能被丢弃.在键入[g] 前将一直检查*broken* 软件包的数量,或者详细显示*broken* 软件包的信息,这将丢弃前边所做的修改.幸运的是,aptitude 允许通过[C_] 或 [C_u] 来取消最后的操作,然后进行手动解决.

当在aptitude中修复*broken*软件包时,它的过滤(或限制)功能非常的有用.键入[I] 进入~b,aptitude 将会只显示*broken* 的软件包.使用软件包细节列表,通过选择缺少的依赖包,手动解决冲突,稍做努力就可修正出现的问题.另外,您也可以简单的键入[b],找到第二个*broken* 软件包.

当*broken* 的软件包数量增加时,可能需要过滤显示软件包列表.键入[I] 应用前面的过滤设置,以缩小仍处于*broken* 状态的软件包列表(希望少了很多).重复操作,在短时间内就会修复所有*broken*的软件包.

7 清洁您的系统

当aptitude 选择了软件包以解决其它的依赖问题时,它将这个软件包标记为自动安装.因此,当依赖软件包卸载时,这些软件包也会自动卸载,这有助于您的系统保持清洁.也可以使用[M] 和[m] 手动修改安装 标示: 首先添加这一标示,随后就会卸载它. ~g 谓词可以用于根据这一标示进行搜索和限制.

虽然aptitude的这一功能非常的不错,但是deborphan也提供近似的功能,并且操作起来更加灵活.

8 虚拟包

有时候,软件包可能需要其它必需选择的软件包提供一个概念.这种需求的一个典型例子可以在软件包的关联信息中找到.at被设计为依赖能发送电子邮件的程序.在Debian中,有不下十种邮件传输代理软件,并不是写死到at的依赖信息中,软件包只是简单的通过依赖于概念包mail-transport-agent 来实现.提供了所需功能的软件包都声明提供了这种概念,在Debian中,是通过所谓的“虚拟包”来实现的.在系统中安装了任意一个提供了mail-transport-agent虚拟包的软件包,Debian软件包工具都认为满足了依赖关系.

```
v  --\ mail-transport-agent      <none>      <none>
    --- Packages which depend on mail-transport-agent
    --\ Versions
p   xmail 1.21-3sarge1
p   masqmail 0.2.20-1sarge1
p   courier-mta 0.47-4sarge4
p   xmail 1.21-3
p   ssmtp 2.61-2
p   sendmail-bin 8.13.4-3
i   postfix 2.1.5-9
p   nullmailer 1.00RC7-22
p   masqmail 0.2.20-1
p   exim4-daemon-light 4.50-8
p   exim4-daemon-heavy 4.50-8
p   exim 3.36-16
p   esmtp-run 0.5.0-4
p   courier-mta 0.47-4
```

aptitude 的 *Virtual Packages*(虚拟包)选项可以用于快速定位提供同一功能的软件包列表. 这样当默认设置不能满足您的需求时,可以使用这一功能快速的完成调整.

9 任务的概念

一个Debian系统通常用于完成某些任务.比如,您可能把它作为您的桌面系统, 或数据库服务器, 或web 服务器, 邮件服务器, 等等. Debian提出的任务的概念是指满足某种需求的一系列典型的软件包; 上边的任务, 您可以

通过安装一系列的软件包,来分别构建一个相应的服务器,或一个桌面环境.

```

--\ Tasks
  --\ End-user
    --\ Desktop environment
p   fortune-mod          <none>    1:1.99.1-2
p   gdm                  <none>    2.6.0.8-1
p   gimp                 <none>    2.2.6-1
p   gnome               <none>    64
p   gnome-desktop-environment <none>    64
p   kde                 <none>    5:44
p   kde-core            <none>    5:44
Tasks are groups of packages which provide an easy way to select a
predefined set of packages for a particular purpose.
If you select a package, an explanation of its current state will appear
in this space.

```

传统的程序tasksel提供了一个选择这些软件集合的界面,不过它的功能被集成到了aptitude中,并进行了扩展,因此aptitude更受欢迎.

任务就是选择一个抽象的概念,然后安装一系列的所必需的软包件,让本地系统可以运行为典型的环境.本质上来讲,任务与关联包(或伪软件包)相似,依赖于扮演着不同角色的所需软件包,它只是一个虚拟软件包,并没有相关的真实*deb*文件.另外,任务仅仅是个建议,用户可以不选择任务建议中的一些软件包.您可能会觉得更喜欢aptitude主选择屏幕最下边的任务界面.与处理软件包相似,任务可以视作单个的实体,或展开展示的建议的软件包.另外,用户也可将任务展开,检查一下建议安装的软件包列表.可以只简单的安装任务所包含软件包的一部分,而不是全部安装.您会发现aptitude中的任务有点象软件包的规则匹配.

也可以通过将任务描述文件放置到/usr/share/tasksel目录下⁹,来自定义任务.在tasksel软件包安装的README文件中提供了有关如何构建任务的文档¹⁰.

10 强大的命令行

aptitude提供了一个有趣的命令行模式,可以作为一个基本的嵌入模式来取代apt-get 并具有apt-cache 的查询能力,并在aptitude 的交互接口增加了

⁹将来也会支持其它的路径,参阅<http://bugs.debian.org/286170>.

¹⁰/usr/share/doc/tasksel/README

搜索判断¹¹.

通常, aptitude的命令行请求形如:

```
# aptitude action [arguments...]
```

常用的一些操作:

```
# aptitude update * 更新软件包列表, 同图形界面的[u]键*
```

```
# aptitude upgrade * 升级软件包, 等同与apt-get upgrade *
```

```
# aptitude dist-upgrade *升级系统, 等同与apt-get dist-upgrade *
```

```
# aptitude [ install | remove | purge ] pkg1 [pkg2...] * ...*
```

```
# aptitude search pattern1 [pattern2...] * ...*
```

就象apt-get,可以在交互界面使用拼接字符将多个不同的查询动作置于同一命令行中.下面的情况中,安装A,删除B,清除C,保持D.'+'是冗余的,因为默认为安装.

```
# aptitude install A+ B- C_ D=
```

aptitude的命令行界面的优越之处在于在统一界面下集成了很多程序以及它们的功能.另外,响应安装请求时,aptitude 可以解决依赖问题,并且在不再需要这些软级包时,将其自动卸载.

11 操作日志

aptitude将您所有的请求动作写入/var/log/aptitude.这个文件可以方便的用于安装和删除软件的跟踪.如果您使用apt-get 安装软件,用dpkg卸载软件,aptitude的日志很快就会同步.另外,aptitude只记录请求.如果某一动作失败了,它是不会记录的.

12 有关这篇HOWTO的一些资料

¹¹更多信息参阅aptitude的联机手册aptitude(1)

参考文献

- [1] <http://www.debian.org>;
- [2] aptitude manpage;
- [3] aptitude user's manual;
- [4] APT HOWTO;
- [5] <http://www.linuxsir.org>;
- [6] <http://aptitude.sourceforge.net/>;
- [7] aptitude mini howto;
- [8] dselect 初学者指南.