

## 18 软件安装，如虎添翼

更新时间：2019-07-09 11:35:47



“

人生太短，要干的事太多，我要争分夺秒。

——爱迪生

”

### 内容简介

1. 前言
2. 初体验软件包和依赖关系
3. 软件仓库
4. 包管理工具
5. 总结

### 1. 前言

上一课配置Nano和终端挺轻松的。

这一课也会比较轻松，所以好好享受吧~

我们来谈谈 Linux 下软件的安装，会在我们的 Ubuntu 这个 Linux 发行版中演示。

一般提到软件安装，有过 Windows 系统使用经验的朋友肯定深有体会，挺麻烦的。大致流程是这样的：

- 在搜索引擎（比如百度或 Google）中搜索我们的软件名称；
- 看到一个不错的结果，点击进入，如果是官网一般会进入官网；
- 下载此软件的安装程序；
- 双击下载下来的安装程序，开始安装，一直按“下一步”，直到“完成”。

但是：

- 有的软件还会问你要安装在何处；
- 有的还会在你不知情的状况下安装捆绑软件，相当不开心；
- 有的还有病毒... 这日子简直没法过了。

当然了，现在 Windows 里面也有 Apps Store 这样统一的软件平台，安装软件也比较方便了，只要在软件平台中搜索，点击安装即可。不过不少第三方软件可能没有收录进 Windows 的官方软件平台，安装起来还是很麻烦。

但是在 Linux 系统中，特别是 Ubuntu 中，安装软件的流程绝对不是这样的，比 Windows 中安装起来方便、安全、快捷多了。

而且，你会发现在 Linux 中安装软件是一种乐趣，就是这个 feel，倍儿爽。

这也是 Linux 相对 Windows 的一个优点，我们一起来体验。

## 2. 初体验软件包和依赖关系

首先，我们得声明：这一课涉及到的软件包安装等知识点是基于 Debian 一族的 Linux 发行版，包括 Ubuntu。

其它不是 Debian 一族的 Linux 发行版，例如 Fedora、Redhat、CentOS 等等。其实安装软件的方式大同小异，就是每个 Linux 发行版之间所用的命令不一样，软件包的名称有点区别罢了，所以一通百通。

在 Windows 下，我们如果要安装软件，一般都是使用“安装程序”——一般是以 .exe 结尾的可执行程序，双击即可运行。大多数情况下我们会把软件安装到 Program Files 这个文件夹下。

在 Ubuntu 下，我们也有类似的程序，但是我们不称之为“安装程序”，而称之为“软件包”，英语是 Package。

一个软件包其实是软件的所有文件的压缩包，二进制形式的，包含了安装软件的所有指令。

在 Debian 一族里，软件包的后缀是 .deb（是 Debian 的前三个字母），Ubuntu 作为 Debian 一族的一员，当然也是用 .deb 的软件包。而如果是 Red Hat 一族（包括 Red Hat、Fedora 等），软件包是 .rpm 为后缀。

那么，一个 .deb 格式的软件包，其实是类似 Windows 下的 .exe 的安装程序咯？

确实很像，但是其实它们的运作方式不一样。我们列出主要的两个不同点：

- 软件包管理包括了依赖关系的管理；
- 软件包不需要我们通过搜索引擎来找到并下载，几乎所有的 .deb 软件包都存放在相同地方，称为软件仓库，英语叫 repository。

这两个方面我们会一一来讲解。

### 依赖关系

通常来说，很少有一个软件可以单独在 Linux 上运行，也就是说它不依赖于其它程序。

一个软件经常需要使用到其它程序或者其它程序的片段（我们称之为“库”）。我们说这个软件依赖其它程序，这就是依赖关系。

比如说，Linux 下类似 Photoshop 的软件 GIMP，它不能单独运行，它的正常运作还需要调用图片读取的库（比如说如何读取一个 JPG 图片），往往依赖关系还有下层依赖关系，环环相扣。

当然，在 Windows 里面你也遇到过依赖问题，比如你运行 Eclipse 软件，它会说你还没安装 Java 环境；你运行一个大型游戏，它提示你先安装 Direct X 等等。

幸运的是，Debian 的包管理系统非常智能。每一个软件包都“知道”它依赖于哪几个程序。这可以让系统去寻找并安装缺失的依赖程序。

所以，你只要告诉包管理系统“我要安装某某软件”，包管理系统会为你打理一切。

### 3. 软件仓库

上面我们说了，Linux 的软件包都存放在一个地方，叫做软件仓库，repository。

对于 Windows 的各种软件，没有这样一个集中存放的地方，所以网上有软件的各种版本、各种型号任君选择。

因为 Linux 是在 Windows 之后诞生的（1991 年前后），所以为了避免 Windows 的这个“弊端”，Linux 选择创建一个集中存放软件的地方。

当然了，Linux 的这一优势也得益于它的开源性。Linux 中的软件基本都是开源自由的，任何人都可以分发，传播软件，只要遵守一定软件协议。因此把这些软件集中存放在某个地方是可行的。

但是 Windows 的大部分软件都是有版权的、闭源的、不开放甚至收费的，这些软件的作者不会同意把他们本该赚钱的软件存放在一个公共仓库任大家分发使用。

#### 软件仓库的概念

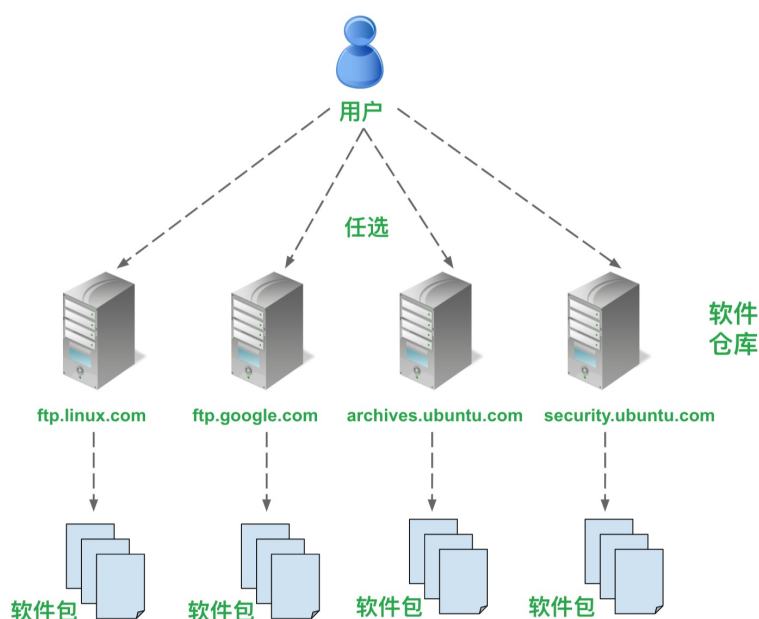
前面我们说了，Linux 的软件会集中存放在一个地方，叫做软件仓库，英语是 repository。

但是你也许会问：假如全球的 Linux 用户都到同一个地方去下载软件，那么存放软件的那个服务器岂不是会死机？

好问题，很有道理。所以，其实软件仓库绝不只是存放在一台服务器上，全球有很多个软件仓库（也就是有很多服务器），大部分软件仓库里的软件都是一样的，只不过是拷贝而已。

有一些特别的软件仓库，会存放其它软件仓库没有的软件，但一般我们用不到这样的软件仓库。

为了加深理解，我们可以用下图来表现软件仓库的原理：



所以说，是用户选择软件仓库，因为基本上各个软件仓库中的软件都是一样的。

我们一般建议用户选择离自己所在地较近的软件仓库的服务器，这样下载速度会比较快，“就近原则”。

例如在法国，Ubuntu 默认会使用 <http://fr.archive.ubuntu.com> 这个服务器上的软件仓库；如果在中国，Ubuntu 会默认使用 <http://cn.archive.ubuntu.com> 这个服务器上的软件仓库。

fr 是国际上代表 France（法国）的缩写，cn 是国际上代表我们的祖国 China（中国）的缩写。

但是使用官方的软件仓库并不一定很好，因为有时候这些服务器会比较卡，例如 Ubuntu 更新版本和其下软件的时候。

接下来，我们来学习如何切换软件仓库吧。

## 管理软件仓库

上面说了，用默认的官方版本没有太大坏处，但是有时候会卡。所以我们要学习如何切换软件仓库。

我们 Ubuntu 系统使用的软件仓库列表是记录在一个文件中。为了修改这个文件，要用文本编辑器，幸好之前的课程我们已经接触了文本编辑器，知道如何使用了。

我们要编辑的那个包含软件仓库的列表文件是：

```
/etc/apt/sources.list
```

这个文件是系统文件，只能被 root 用户修改。所以我们有两种选择：

1. 暂时性地成为 root，用 Nano 来编辑此文件：

```
sudo nano /etc/apt/sources.list
```

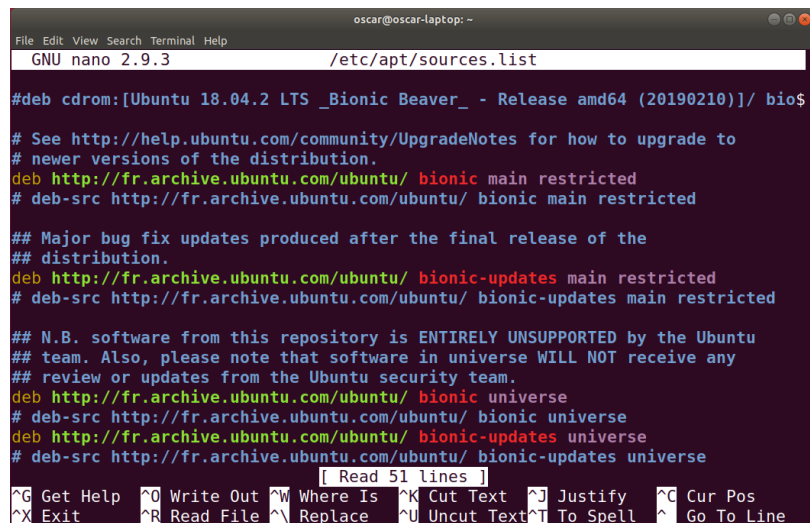
2. 先永久成为root:

```
sudo su
```

再用 Nano 来编辑此文件：

```
nano /etc/apt/sources.list
```

用 Nano 打开此文件后，我们看到文件内容类似如下：



```
oscar@oscar-laptop: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 /etc/apt/sources.list

#deb cdrom:[Ubuntu 18.04.2 LTS _Bionic Beaver_ - Release amd64 (20190210)]/ bios$
# See http://help.ubuntu.com/community/UpgradeNotes for how to upgrade to
# newer versions of the distribution.
deb http://fr.archive.ubuntu.com/ubuntu/ bionic main restricted
# deb-src http://fr.archive.ubuntu.com/ubuntu/ bionic main restricted

## Major bug fix updates produced after the final release of the
## distribution.
deb http://fr.archive.ubuntu.com/ubuntu/ bionic-updates main restricted
# deb-src http://fr.archive.ubuntu.com/ubuntu/ bionic-updates main restricted

## N.B. software from this repository is ENTIRELY UNSUPPORTED by the Ubuntu
## team. Also, please note that software in universe WILL NOT receive any
## review or updates from the Ubuntu security team.
deb http://fr.archive.ubuntu.com/ubuntu/ bionic universe
# deb-src http://fr.archive.ubuntu.com/ubuntu/ bionic universe
deb http://fr.archive.ubuntu.com/ubuntu/ bionic-updates universe
# deb-src http://fr.archive.ubuntu.com/ubuntu/ bionic-updates universe

[ Read 51 lines ]
^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File ^_ Replace   ^U Uncut Text ^T To Spell  ^_ Go To Line
```

我写这篇文章的时候还在法国巴黎。如果是在中国，那么上面截图中的链接里的 `fr` 就会替换为 `cn`。 <http://fr.archive.ubuntu.com/ubuntu/> 就会是 <http://cn.archive.ubuntu.com/ubuntu/>。

上图中，由 `#` 开始的行是注释，会被忽略，不会被处理。

通常来说，这个文件中有作用的行是由以下两个指令开头：

- `deb`：用于下载软件的二进制版本，大多数情况下我们都是用这个；
- `deb-src`：用于下载软件的源代码。一般我们用不到，除非你很好奇想查看软件的源代码。这也是自由软件运动的一个好处，默认是用 `#` 号注释掉的。

以下是一行的例子：

```
deb http://fr.archive.ubuntu.com/ubuntu/ bionic universe
```

- 第一个参数 <http://fr.archive.ubuntu.com/ubuntu/> 是法国 Ubuntu 官方软件仓库的服务器；
- 第二个参数 `bionic`，代表我们所使用的 Ubuntu 的版本号。`bionic` 是 Ubuntu 18.04 的代号，`bionic` 的全称是 Bionic Beaver（仿生河狸）；
- 最后一个参数（或者后面还有其它参数）是软件仓库的区域，也就是你要查看的代码仓库的不同分区，`universe` 是英语“宇宙、世界”的意思。

上面的概念稍微有点复杂，但其实你只要知道一件事：如果要切换其它的软件仓库，只需要将 `/etc/apt/sources.list` 这个文件中的 <http://fr.archive.ubuntu.com/ubuntu/> 服务器地址替换为其它软件仓库的服务器地址即可。

那么问题来了：我怎么知道其它软件仓库的服务器地址呢？

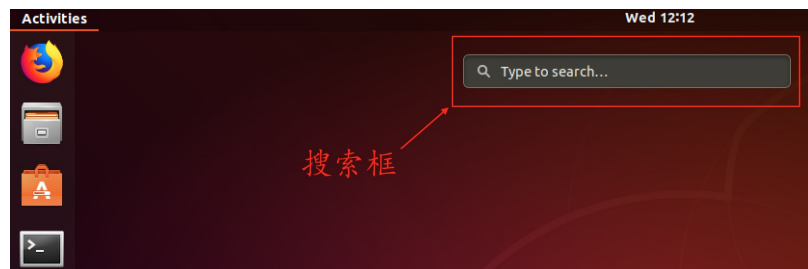
你可以用搜索引擎来寻找，比如可以搜“Ubuntu repository”。

但是我们要演示一个更快捷方便的方式，也就是要用到图形界面的方式了。

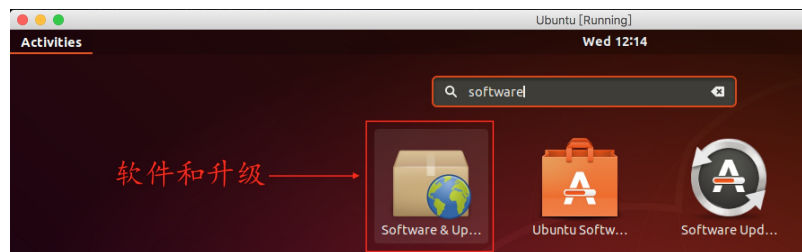
## 图形界面工具

Ubuntu 提供了一个图形化的代码仓库管理软件，可以按以下步骤启动：

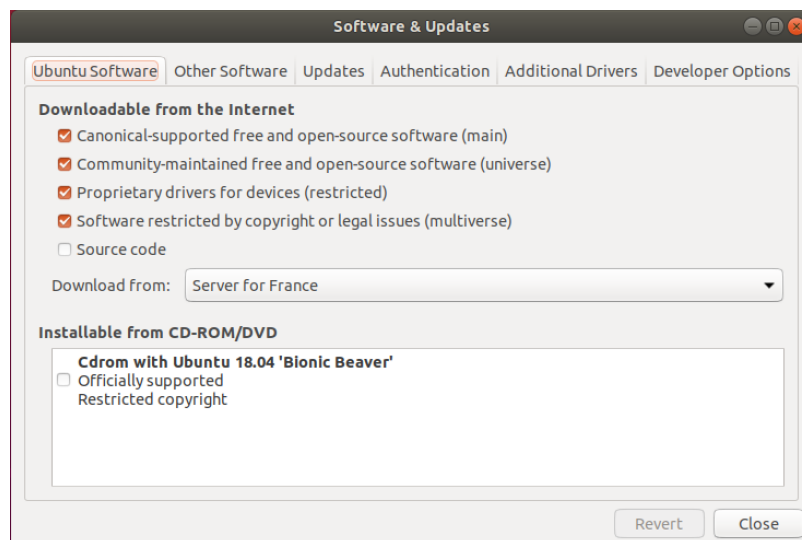
像我们之前的课程演示过的，单击左上角的 `Activities` 字样，会在屏幕中上方显示搜索框：



在搜索框中输入 `software`（`software` 是英语“软件”的意思），它会为你显示搜索到的几个结果：



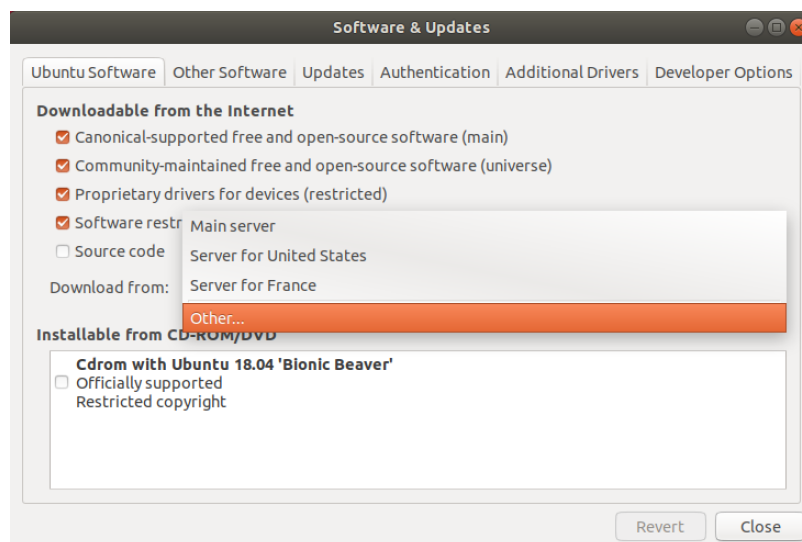
单击那个 Software & Updates（表示“软件和升级”）的图标，打开后如下图：



上图中，我们可以看到默认是打开 Ubuntu Software（表示“Ubuntu 软件”）这个菜单。

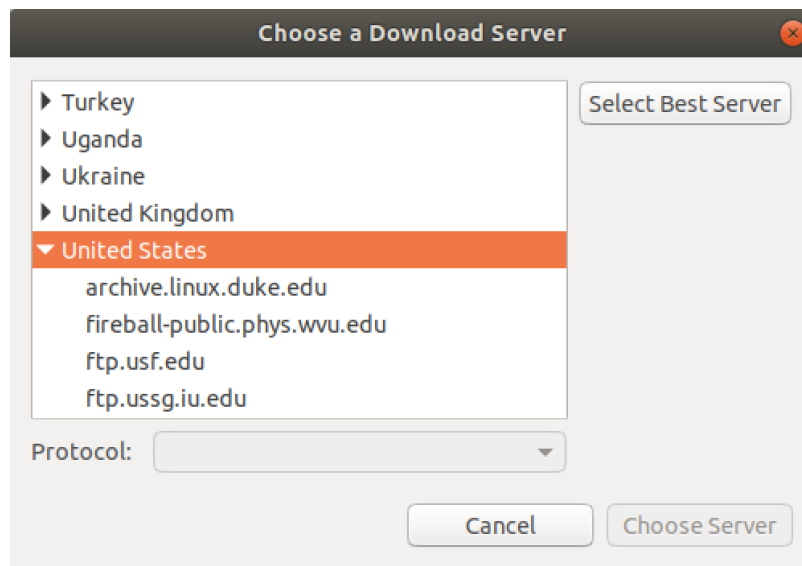
在这个菜单里可以看到，默认的服务（“Download from: ”表示“从...下载”）是法国的：Server for France（意思是“为法国提供的服务器”）。

我们可以将这个下拉列表选成“Other...”（other 是英语“其他”的意思）：



那么会打开另一个窗口，如下图：

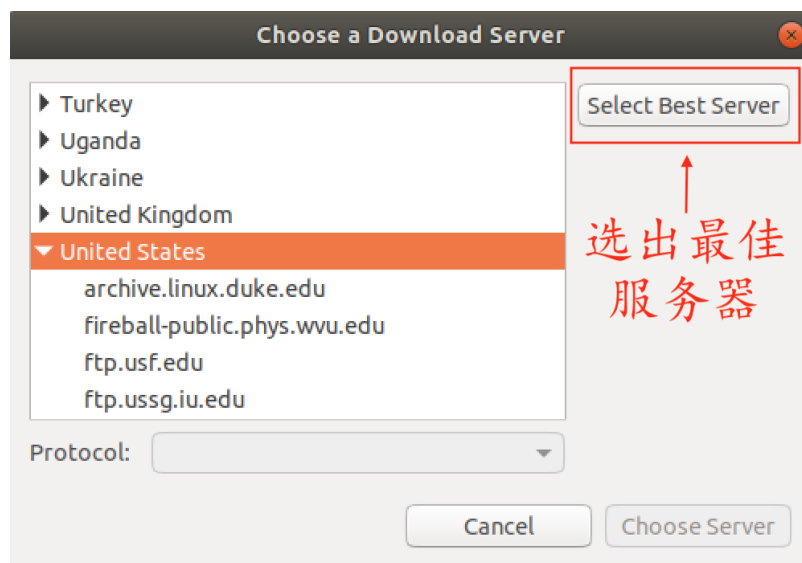




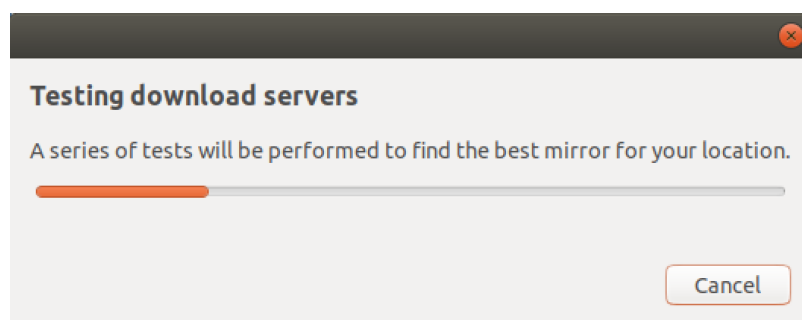
可以看到列表中有很多服务器可供选择，这些都是软件仓库的服务器地址。按国家分类有：United States（美国）、United Kingdom（英国）、Ukraine（乌克兰）、Turkey（土耳其）、Uganda（乌干达）等等。

为了选出最快的服务器，我们可以做一个实际测试：

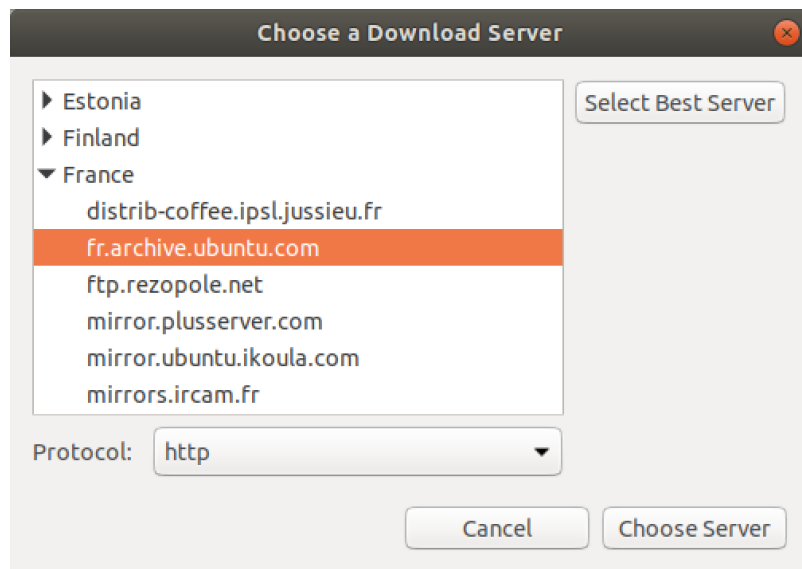
点击右上角的 Select Best Server （表示“选出最佳服务器”）按钮：



然后就会启动测试，测试各个服务器的速度：



测试完之后，最快的服务器会被选中，如下图：



我测试的时候，最快的服务器还是法国的 Ubuntu 官方服务器。不过，有几次测试的时候，它给我定位到了意大利的某个服务器。你的情况，应该会是其它服务器。

点击右下角 Choose Server（“选择服务器”）来选择这个最快的服务器，需要输入你的用户密码来生效。

选择之后，会关闭当前小窗口。回到主窗口，我们可以看到 Download from: 后面的下拉列表已经变为了最快的那个服务器地址。

重新用 Nano 来打开你的 `/etc/apt/sources.list` 这个文件，可以看到服务器的地址都已经换成了新的地址。

所以，图形界面的方式更方便、更直观，比直接用文本编辑器逐行修改 `/etc/apt/sources.list` 这个文件快多了。

## 4. 包管理工具

我们来总结一下：

- **package**：包。这是软件的二进制安装包，类似 Windows 中软件的安装程序（大多以 `.exe` 结尾）；
- **dependency**：依赖。一个软件包可能需要其它软件包作为运行基础，这是依赖关系；
- **repository**：仓库。软件的仓库，就是存放软件的服务器，我们从这些服务器上下载软件。

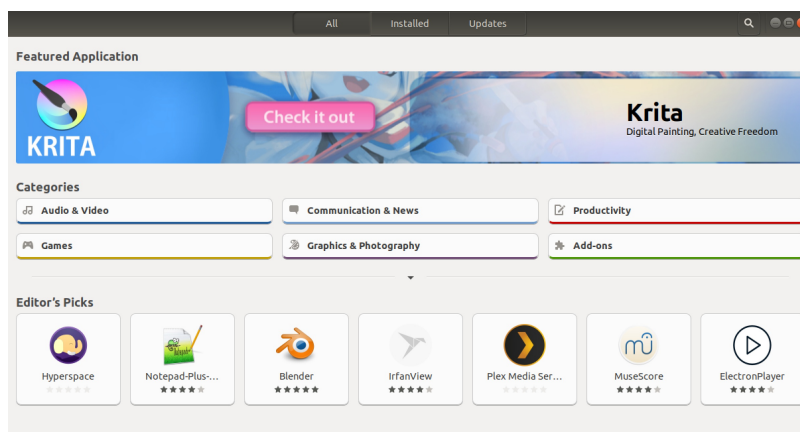
我们上面演示了如何切换其他软件仓库，但也不一定要切换，用默认的软件仓库也是可以的。



在 Ubuntu 中，我们可以使用图形界面的软件包管理，例如 Ubuntu 自带的 Ubuntu Software（Ubuntu 软件），要打开很简单，在左边的快捷方式栏里面就有这个软件的快捷方式：



单击打开后，如下图：



在这个软件里，我们可以搜索你想要的软件、安装和升级软件。不过很多第三方的软件没有被收录在里面。

当然了，我们也可以使用终端来管理软件包，终端的软件包管理命令一般用两个：

- apt-get: 以前一般用这个比较多，Ubuntu 自带了这个软件；
- apt: 现在一般用这个比较多，Ubuntu 自带了这个软件。Ubuntu 16.04 发布时，一个引人注目的新特性便是 apt 命令的引入。apt 正在逐步取代 apt-get，推荐大家使用 apt。关于 apt 和 apt-get 的区别，可以参考：[Linux中apt与apt-get命令的区别与解释](#)。我平时也是用 apt 比较多，毕竟可以少输入几个字母；
- aptitude: 这个命令在卸载软件时可以卸载不用的依赖。Ubuntu 没有自带 aptitude，如果你想安装，用 `sudo apt install aptitude` 命令来安装。

以后的课程我们会使用 apt-get 或 apt 一族来演示，比较标准，一般大家都是用这个。

以下我们再学习几个常用的终端软件包命令，记得这些命令都需要 root 身份来运行：

### apt update: 软件包缓存的更新

一般是：

```
sudo apt update
```

## apt search: 搜索软件包

一般是：

```
sudo apt search xxx
```

xxx 是对应软件包名。

## apt install: 安装软件包

一般是：

```
sudo apt install xxx
```

xxx 是对应软件包名。

## apt autoremove: 删除软件包

一般是：

```
sudo apt autoremove xxx
```

xxx 是对应软件包名。

## apt upgrade: 升级所有已安装的软件包

一般是：

```
sudo apt upgrade
```

---

有时候，我们也希望查询软件包的信息，可以用以下命令：

```
apt-cache show package_name
```

上面的 package\_name 用实际软件包名来代替。例如：

```
apt-cache show emacs
```

可以查询 Emacs 这个软件的很多信息。

如果是 RedHat 系的系统（例如 RedHat，Fedora 和 CentOS），那么可以用以下命令：

```
yum info package_name
```

补充一下：

dpkg 这个命令也是很有用的。事实上，dpkg 是 apt-get、apt 和 aptitude 这三个命令的后端（backend）命令，而 apt-get、apt 和 aptitude 又是 Ubuntu Software 这样的图形界面的软件后端。

依赖链:

dpkg --> apt-get、apt、aptitude --> Ubuntu Software

因此，如果是本地的 .deb 软件包，我们可以用 dpkg 来安装：

```
sudo dpkg -i *.deb
```

卸载：

```
sudo dpkg -r xxx
```

xxx 是对应软件包名。

### 小结

1. 大部分 Linux 的发行版都提供了集中式的软件包下载和安装方式，因此软件的安装比 Windows 下来得简单很多，这些集中存放软件的服务器被称为软件仓库；
2. 在 Ubuntu 下，我们可以使用图形界面的软件包管理来安装软件，当然我们也可以使用终端命令来安装软件，就用 apt-get 或 apt 命令族；
3. apt update 命令：更新可安装的软件列表；
4. apt search 命令：在软件列表里搜索可用软件；
5. apt install 命令：安装软件；
6. apt autoremove 命令：卸载软件；
7. apt upgrade 命令：升级所有已安装软件。

今天的课就到这里，一起加油吧！