

Package Maintenance and Update

INTRODUCTION

While a simple piece of software may consist of only a single executable file, most of the software applications available and running on your system are more complex. A typical application or utility will consist of several executable files, configuration files, documentation notes and guides and possibly even libraries too. All these files and information about where to place them in the filesystem are put together in what is referred to as a package. So when we talk about the installation or upgrading of applications, we are referring to the installation and maintenance of these packages.

There are many packaging formats available in Linux, and some are easier to use than others. In this Chapter we shall cover tarballs, the RPM packaging format (RPM) and the Debian packaging format (DEB).

TARBALLS

Tarballs are the standard, and are common with file extensions such as ".tar.gz" or ".tar.bz2". This is the generic, distribution-free method of distribution software packages in the Linux world. However, tarballs are not very user-friendly; for example, to get a tarball from the Internet running, one might have to issue the following commands from the command line in a shell,

```
# bunzip2 myapp.tar.bz2
# tar -xvpf myapp.tar
# cd myapp
# ./configure
# make
# make install
```

This is a tedious task, and involves getting the software to compile before being able to run. If know-how is lacking, this method will also cause a lot of grief, as sometimes during the "configure" stage, dependencies to get it running aren't met.

This is the aim of package management formats like RPM and DEB – to ease the burden of dependency resolution, so that the end-user will just install the software with ease, and if dependencies are required, they get installed along.

KEEPING UP-TO-DATE

On Red Hat Linux/Fedora Core systems, there is a graphical front-end called up2date. It is invoked by the little icon at the bottom of the notification area (nearby where the clock is located).



Fig. Up2date Icon

It can also be accessed via,

```
Main Menu --> System Tools -->
Red Hat Network
```

When you run it, it will require that you enter the root password (as this affects the system, administrative rights are required). Once that is entered, an image like the one below is displayed.

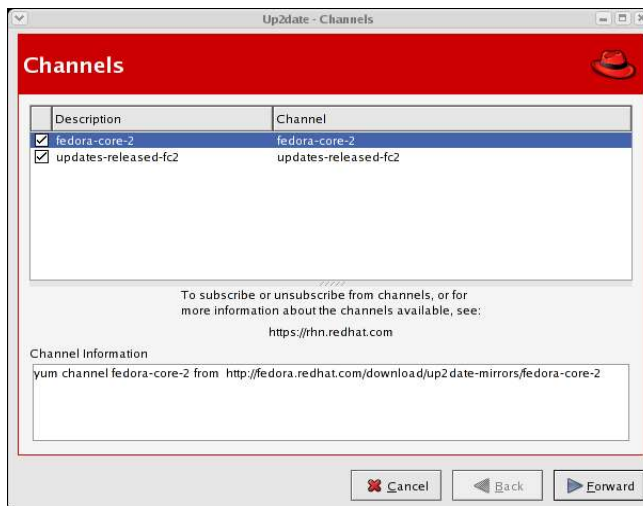


Fig. Up2date Channels Screen

Just click Next (make sure the Internet connection is enabled, if it is not enable it before proceeding further) – it will contact the online servers and find packages that are installed and need updating, and it will prompt you along the way. Once it is done, your system will be updated (and the blue icon with a tick will be displayed). If your system is not updated or the packages not kept up to date, a red icon with an exclamation mark will be displayed and it will be blinking.

On a Debian GNU/Linux system, a tool known as "apt-get" is available on the command line. On a default install, that is all that is provided, however, a good GUI front-end to it is Synaptic, which can be downloaded from the Internet via,

```
# apt-get install synaptic
```

Fedora also comes with another updating tool known as yum, and this can be invoked via the command line such as,

```
# yum update
```

To upgrade your current system,

```
# yum upgrade
```

can be invoked.

Keep in mind that keeping an updated system is very important, as when security holes or bugs are found in software and get fixed, you will always be kept abreast of such developments. A non-updated system can be an insecure system, and that is not good practice.

INSTALLING NEW PACKAGES

If a package is available on your Red Hat Linux or Fedora Core CDROM, there's an Add/Remove Applications application that is useful. It is invoked via,

```
Main Menu --> System Settings -->
Add/Remove Applications
```

It will ask you for the root password, and once that is provided, it will display all applications that may be installed. Once you have ticked the applications that you want installed, you just need to click "Update" to install. Change the discs as you are prompted, and once this is done, you will have the software installed.

However, in the open source world where applications change quite often, and fixes are posted, this method might mean you get out-dated software. This is where tools like yum and apt come into play.

To search the yum database for a piece of software, you can invoke,

```
# yum search xargs
```

where xargs is an example of an application that needs to be installed. Yum will report if it finds xargs, and if its successful, performing,

```
# yum install xargs
```

will be all that is required. If xargs calls for any dependencies, it will be resolved automatically, and those packages get pulled in automatically too.

This is similar with Debian and apt.

```
# apt-cache search xargs  
# apt-get install xargs
```

If you want to install a downloaded RPM or DEB file manually, it can be performed like,

```
# rpm -ivh xargs.rpm
```

or

```
# dpkg -i xargs.deb
```

And if you're manually upgrading a package, use,

```
# rpm -Uvh xargs.rpm
```

The above command will upgrade the package if it is already installed or install it if it is not. To perform an upgrade only if the package is currently installed, use,

```
# rpm -Fvh xargs.rpm
```

There are many more options to pass to the rpm, dpkg, yum, apt-get and apt-cache tools, and the best way to learn more, would be to read their manual pages. It is also worthy to note that apt-get is available for RPM-based systems, so versions for Red Hat Linux or Fedora Core (or even SuSE or Mandrake) are available as a download from the Internet.