How do I use apt-get to install some "testing" packages into a Debian "stable" system?

If you've found yourself wanting to run a Debian stable system but you have a burning need to use a particular testing package, then this is for you. (This might be the case if you want to run PHP5 or PostgreSQL 8.1 on Debian Sarge.)

Some notes:

- Don't try this unless you already understand the differences between the stable, testing, and unstable distributions of Debian — No, seriously! Don't even think about trying this if you don't have some clue about what you're getting yourself into
- The following works for Debian Sarge
- You can change the repository servers to your favorite ones... I just picked an arbitrary one as an example

Edit your apt sources list and add sources for testing (and unstable also if you're feeling adventurous):

Edit this file:
/etc/apt/sources.list:

```
# these entries were in the file already:deb http://mirrors.kernel.org/debian/
 stable main non-free contribdeb-src http://mirrors.kernel.org/debian/
 stable main non-free contribdeb http://security.debian.org/
 stable/updates main non-free contrib# add these entries for testing:deb
http://mirrors.kernel.org/debian/ testing main non-free contribdeb-src
http://mirrors.kernel.org/debian/
 testing main non-free contrib# add these entries for unstable:deb
http://mirrors.kernel.org/debian/ unstable main non-free contribdeb-src
http://mirrors.kernel.org/debian/ unstable main non-free contrib
```

The next step is to make sure that your system remains a "stable" system by default and will only pick up "testing" packages when you explicitly say so. You'll need to create a preferences file for apt:

Create this file:
/etc/apt/preferences:

```
Package: *Pin: release a=stablePin-Priority: 700Package: *Pin: release a=testingPin-Priority: 65
```

A higher number means that the distribution will be chosen before a distribution with a lower number.

This is what you do to force a testing package (such as PostgreSQL 8.1) to be installed:

First update your package listings:

```
apt-get update
```

Now install some bleeding-edge package:

```
apt-get -t testing install postgresql-8.1
```

IMPORTANT: Pay attention to what apt tells you after you enter this command! You will most likely need to install/upgrade a number of dependencies to get a testing package to install. Sometimes you will be upgrading shared libraries that are also used by other stable packages. Think about what you're doing and make sure you can live with these shared libraries being switched to testing. If the number of upgrades required fills up an entire page… you might want to reconsider what you're doing…

Revision #11
Created 2006-05-10 18:33:35 UTC by Fong, Jason
Updated 2006-05-10 18:42:11 UTC by Fong, Jason