

Building Debian Packages by Example: Packaging libgnupg-perl

Chris Fearnley cjf@LinuxForce.net <http://www.LinuxForce.net>

We will demonstrate the building of a Debian package by preparing a .deb file and source packages for libgnupg-perl, a Perl Interface for Gnu Privacy Guard (GnuPG.pm). Since this is a perl library module a few extra considerations are applicable. However, GnuPG.pm is still very simple so this session will be appropriate for those new to Debian development.

1. Unpack the upstream source into a pristine directory named libgnupg-perl-version
2. Debianize the directory by cd'ing into it and running
`dh_make -e your.maintainer@address -f ../pristine-source.tar.gz`
3. Examine the upstream source (<http://indev.insu.com/GnuPG/gnupg.html>)
4. Write a short description of the package
Perl Interface to Gnu Privacy Guard (GPG)
5. Write a long description for the package
GnuPG is a perl module that interfaces with the Gnu Privacy Guard using the coprocess hooks provided by gpg. The communication mechanism uses shared memory and a status file descriptor. There are other perl interfaces to PGP and GnuPG but none of them use the coprocess interface provided by gpg.
The module tries its best to map the rather interactive interface of gpg to a more programmatic API. The module offers a tied file handle interface to encryption and decryption making it easier to use.
6. Post an "ITP" (Intent To Package) e-mail announcement that you are working on the package to debian-devel@lists.debian.org
7. Verify the license is DFSG (Debian Free Software Guidelines) compliant
8. Write a man page if the package doesn't already provide one
9. Write a debian/control file
10. Write a debian/copyright file
11. Write a debian/rules file
12. Build the package
 - (a) (Skip this step for scripts) Build the executables `debian/rules build`
 - (b) Build the binary package (.deb) `debian/rules binary`
13. Test the package
 - (a) `ls -alR debian/tmp*`
 - (b) `dpkg -c ../libgnupg-perl-version.deb`
 - (c) `dpkg -I ../libgnupg-perl-version.deb`
 - (d) `dpkg -i ../libgnupg-perl-version.deb`
 - (e) `lintian ../libgnupg-perl-version.deb`
14. Install the package (`dpkg -i ../libgnupg-perl-version.deb`)
15. Test the package some more
 - (a) Does it work
 - (b) `dpkg -L libgnupg-perl`
 - (c) `dpkg -s libgnupg-perl`
 - (d) Install it on at least one system on which the package was not built (to ensure the package includes all components necessary to work).
 - (e) `--remove` the package and then reinstall. Any errors?
 - (f) Are the maintainer's scripts idempotent? Try running them multiple times.