

## The Sysadmin's Daily Grind: Checkinstall

# Against the tide

Most programs we have looked at in this column are only available as source code archives. There is nothing wrong with that, of course, as building and installing is normally quite easy. However, what most programmers tend to forget is a *make uninstall*. **BY CHARLY KÜHNAST**

Software that is distributed in the form of a source code tar archive is always installed in the same way: read the *README* and *INSTALL* file (assuming they exist), then say the magic words *./configure; make; make install* – and you're up and running.

Unfortunately, when I'm looking for software, I have a tendency to choose the wrong programs, and I should be very surprised if I were the only person to have that problem. What I need in this case is an easy way out. So how can I get rid of the unwanted programs? Where on earth did *make install* put those executables when I wasn't paying attention? Why didn't the developer include an *uninstall* in the Makefile? Checkinstall gives me the answers to all these questions. Lucky me!

### Installwatch

Instead of painstakingly removing the unwanted software from the disk, I use a tool called Installwatch [1] to tell me which components *make install* has put where in the filesystem. Although Installwatch is a useful tool to have, it doesn't remove the need to manually

delete whatever software I don't want to keep. Checkinstall [2] is based on Installwatch, and goes just one small, but important, step further.

The big difference is that Checkinstall not only monitors where executables are put during the install, but captures them on the way there, and makes neat little packages of them – the options are RPM or DEB packages, or even Slackware TGZs. I can use the default distribution tool (*rpm*, *dpkg*...) to install a package compiled in this way. Also, if I decide that enough is enough, I can use the same tool to cleanly remove the package.

### Practical Applications

For my next trick, and just to try this out, I am going to try out a program that follows the typical build and install steps of *./configure; make; make install*: DNRD, a DNS-Proxy [3]. After unpacking the tar archive, the next step is to change to the new *dnrd-2.13* directory and run

```
./configure
make
```

Now, instead of typing *make install*, it's time to run Checkinstall:

```
checkinstall --type=rpm
```

Checkinstall quickly asks me if I want to put the documentation in */usr/doc/* – why not! – and then runs *make install*. It captures the filesystem commands involved, *cp* and *mv*, and uses the captured data to build a RPM package. I can now go on to install the package, and if I want to get rid of DNRD at some time in the future, I can simply run *rpm --erase packagename* to do so.



What happens if the magic words are not *make install*, but *make all* instead, for example? Easy, just add them to the Checkinstall command:

```
checkinstall --type=rpm make all
```

As I typically confirm, when Checkinstall prompts me about the *doc* directories, I tend to use the *--default* parameter. This tells Checkinstall not to bother asking. Slackware users should watch out for a trap: if you have a recent version of Slackware (8.1 or newer), you need to specify the *--newslack* parameter to prevent Checkinstall from using a pre-historic package format. ■

### INFO

- [1] Installwatch: <http://asic-linux.com.mx/~izto/installwatch.html>
- [2] Checkinstall: <http://checkinstall.izto.org>
- [3] DNRD: <http://dnrd.sourceforge.net>

### SYSADMIN

#### Admin Workshop .....58

Unix systems have a huge range of functions that can be hard for an administrator to find. This article helps you to locate and specifically disable unwanted communication services.

#### Reiser 4 .....60

The new Reiser 4 file journaling system is about to be released. We take the first look at what should be the answer to our Linux needs.

### THE AUTHOR

Charly Kühnast is a Unix System Manager at the data-center in Moers, near Germany's famous River Rhine. His tasks include ensuring fire-wall security and availability and taking care of the DMZ (demilitarized zone).

