

Linux som utvecklingsmiljö

Tema 13 Pakethantering

HT-13

Petter Lerenius, 7907030295

Innehållsförteckning

Uppgiftsbeskrivning.....	2
Genomförande.....	2
1.1 Elektrolib-paketet.....	2
1.2 Elektrotest-paketet.....	3
2 Elektrotestgtk-paketet.....	4
Slutsatser.....	4
Referenser.....	4

Uppgiftsbeskrivning

1. Skapa två debian-paket för den elektrottest-applikationen som skapades i lab 6. Paket 1, elektrolib, ska bestå av de 3 biblioteken som skapades i lab 6 och ska installera dessa på lämpligt ställe i filsystemet. Paket 2, elektrottest, ska bestå av själva applikationen. Paketerna ska vara komplett, följa standarden, inkludera beroenden osv. En användare ska kunna spara ner paketet elektrottest till sin dator, köra något av de vanliga pakethanteringssystemen för deb-paket och applikationen ska installeras med de beroenden som behövs.
2. Skapa motsvarande paket som ovan men för den grafiska variant av elektronikapplikationen i lab 11.

Se kurshemsidan för mer information.

Genomförande

För att utföra den här laborationen bestämde jag mig för att göra på två olika sätt. Elektrolib-paketet skapade jag genom att använda mig av make och olika paketverktyg som jag redovisar nedan, medan jag för de övriga två paketerna skapade paketerna på enklast möjliga sätt genom att kopiera in färdiga binärer.

1.1 Elektrolib-paketet

För att skapa Elektrolib-paketet använde jag en metod som beskrivs i Professional Linux Programmer. Jag kopierade in filerna från Tema 6, och sedan slängde jag allt som inte behövdes för att bygga lib:en. Jag modifierade sedan Makefilen så att target 'all' bygger biblioteken, medan target 'install' skapar katalogerna \$(DESTDIR)/usr/lib och \$(DESTDIR)/usr/include och kopierar biblioteken till lib och headerfilerna till include. Target 'clean' tar bort alla .o- och .so-filer. Variabeln \$(DESTDIR) definieras av programmet dpkg (en pakethanterare för Debian) när paketet byggs.

Nästa steg är att packa ihop källkodsfilerna till ett .tar.gz-arkiv, elektrolib-1.0.tar.gz. Detta görs i katalogen elektrolib-1.0, med kommandot:

```
tar -czvf ../elektrolib-1.0.tar.gz *
```

Nu är vi redo för att köra dh_make, vilket är ett verktyg som formaterar en katalog med källkodsfiler till ett format som överensstämmer med Debians paket policy. dh_make måste startas inifrån den katalog som innehåller källkodsfilerna och som är namngivna enligt formatet

```
<paketnamn>-<version>
```

Så stående i katalogen elektrolib-1.0 exekveras kommandot:

```
dh_make -e petter.lerenius@gmail.com -f ../elektrolib-1.0.tar.gz
```

där -e används för att ange en e-postadress som skall användas som adress, i debian/control-filen, till den person som är ansvarig för paketet. Flaggan -f anges för att peka ut en sökvägen till filen som innehåller källkodsfilerna.

När dh_make körs får man svara på vilken typ av paket man vill skapa, här valde jag 'l' (lilla L) för bibliotek (library). Nu skapas en underkatalog 'debian' samtidigt som ett antal filer genereras i denna katalog.

Filen changelog kan sedan editeras för att stämma med paketets innehåll. Även i control-filen tog jag bort ganska mycket. Den sista filen som jag behövde editera var filen copyright. Här valde jag att använda GPL-2+ för alla filer.

Nu är vi redo för skapandet av själva paketet. Detta görs med kommandot dpkg ett steg upp i katalogstrukturen, dvs i katalogen elektrolib-1.0, körs:

```
dpkg-buildpackage -rfakeroot
```

Om allt går som det ska nu, så skapas ett paket elektrolib_1.0-1_amd64.deb en nivå upp i katalogstrukturen. *-rfakefoot* används för att lura paketbyggmjukvaran att den körs medfullständiga root-behörigheter. Det nya paketet kan nu installeras med dpkg. Installationen görs med kommandot:

```
sudo dpkg -i elektrolib_1.0-1_amd64.deb
```

För att ange att det är en installation som ska göras, används växeln *-i*.

1.2 Elektrotest-paketet

För de andra två paketen valde jag ett enklare och mer manuellt sätt att skapa paketen. Först skapades en kataloger med lämpliga namn:

```
mkdir -p elektrotest_1.0-1/DEBIAN
mkdir -p elektrotest_1.0-1/usr/bin
```

Sedan kopierade jag in elektrotest-binären från tema6

```
cp ../tema6/elektrotest elektrotest_1.0-1/usr/bin/elektrotest
```

Sedan är det dags att skapa några av filerna som dh_make skapade åt oss I förra steget, nämligen control, changelog och copyright filerna. Jag utgick från mallarna som dh_make skapade. För att få till så att beroendet till Elektrolib hanterades lades beroendet till i control filen, som då ser ut enligt följande:

```
Package: elektrotest
Architecture: all
Maintainer: Petter Lerenius <petter.lerenius@gmail.com>
Depends: elektrolib (>= 1.0), debconf(>= 0.5.00)
Priority: optional
Version: 1.0
Description: An amzing electronics calculator
This app will blow you away! Install now!
```

Innan vi skapar paketet så måste alla filer byta ägare till root, så att inte ägandeskapet blir ett problem vid installationen. När detta är gjort kan vi skapa paketet med:

```
dpkg-deb --build
```

Sen är paketet redo att installeras.

2 Elektrotestgtk-paketet

Det här paketet skapades på samma sätt som paketet ovan (1.2). Det är egentligen bara namnet på

paketet och dess binär som ser annorlunda ut.

Slutsatser

Det var betydligt lättare att skapa ett eget paket med beroenden än vad jag trodde innan jag gjorde uppgiften. Det var intressant att se hur det är uppbyggt och hur smidigt det kan vara att sprida sin egen mjukvara genom att paketera den för distribution.

Med det sagt krävdes det ändå en hel del letande och provande för att få ihop det. Jag tyckte att det var svårt att hitta information som var enkel att följa.

Jag försökte också hitta något sätt så att beroendet skulle lösas automatiskt om elektrolib paketet låg i samma katalog, men jag lyckades inte hitta någon information om hur man skulle kunna åstadkomma det. Jag antar att det krävs att båda paketen finns i ett APT repository.

Hur som helst var det ännu en intressant uppgift att lösa!

Referenser

1. <http://www.moodle2.tfe.umu.se/mod/assignment/view.php?id=274>
2. <http://www.sj-vs.net/creating-a-simple-debian-deb-package-based-on-a-directory-structure/>
3. <http://ubuntuforums.org/showthread.php?t=910717>
4. <https://www.debian.org/doc/manuals/maint-guide/first.en.html>
5. Professional Linux Programmer