
Meer over het installeren van software in Linux

Hans Luning

Inleiding

In SoftwareBus 2005/5 (september) heb ik u meegenomen op een tocht langs de diverse systemen voor softwarebeheer, die in de verschillende Linux-distributies worden toegepast. We zagen dat de twee belangrijkste basissystemen de Redhat Package Manager (RPM) en de Debian Package Manager (DPKG) zijn. Voor DPKG is de Advanced Packaging Tool (APT) ontwikkeld, waarmee gemakkelijk nieuwe software kan worden opgehaald en geïnstalleerd. Deze tool zorgt er voor dat - indien nodig - ook alle pakketten waar het gewenste pakket afhankelijk van is, worden opgehaald en geïnstalleerd. En om het nog gemakkelijker te maken is er een grafische gebruikersomgeving (GUI) voor APT met de naam Synaptic.

Was dit er aanvankelijk alleen voor Debian en daarvan afgeleide distributies, zoals Ubuntu, met APT4RPM is al dat moois ook ter beschikking gekomen van gebruikers van op RPM gebaseerde distributies, zoals SUSE, Fedora en Mandriva. Zo is APT met Synaptic een vrijwel universeel systeem voor softwarebeheer geworden, of het nu om RPM-pakketten (rpm) of om Debian-pakketten (deb) gaat. Ik heb u daarom wat uitgebreider kennis willen laten maken met APT en de wijze waarop APT met Synaptic kan worden aangestuurd.

Beide systemen beschikken echter ook over een tool voor het beheren van de

software die vanaf de opdrachtregel wordt bediend: rpm en dpkg. Zoals ik in het vorige artikel aankondigde, voelen we deze beide tools nu wat nader aan de tand.

De laatste jaren zijn nieuwe universele systemen voor pakketbeheer opgekomen en één daarvan, Autopackage, zou wel eens een belangrijke rol kunnen gaan spelen. Het is niet afhankelijk van een bestaand pakketformaat, en werkt in beginsel in elke distributie. Na rpm en dpkg gaan we ook wat dieper in op Autopackage.

Rpm en dpkg

Rpm en dpkg zijn de beheersprogramma's van de gelijknamige systemen voor pakketbeheer. Ze zijn eigenlijk het basisgereedschap, dat ook APT onder de motorkap gebruikt om zijn werk te doen. "APT?", zult u zich afvragen. Waarom zelf rpm en dpkg gebruiken, als APT het moeilijke werk al doet? Wel, stel dat u ergens een leuk programma hebt gevonden en het in het voor uw distributie geschikte formaat (rpm of deb) hebt gedownload. U zou het vervolgens met APT kunnen installeren, maar APT vereist wel dat u daarvoor een speciale repository maakt. Nu is dat niet zo heel moeilijk, maar het is wel zo simpel om het te vermijden door rechtstreeks rpm of dpkg te gebruiken. En als u dan toch in het gebruik van rpm en dpkg bent ingevoerd, kunt u het ook prima gebruiken voor het even updaten

of verwijderen van een pakket, of om er allerlei informatie over op te vragen. Het enige bezwaar bij installatie van een pakket is, dat afhankelijkheden niet automatisch worden opgelost. Mocht het niet lukken om een pakket om die reden te installeren, dan kunt u altijd nog terug naar Synaptic.

Goed, we gaan aan de slag. Beide pakketten werken vanaf de opdrachtregel, en dat zal wel even wennen zijn, als u alleen maar grafisch hebt leren werken! In KDE krijgt u een opdrachtregel door konsole te starten, en in Gnome vervult `gnome-terminal` die rol. U herkent ze in het paneel gemakkelijk aan hun icoontje, dat een computerscherm voorstelt. Log vervolgens in als root door de opdracht `su -` te geven. Na uw wachtwoord te hebben ingevoerd bent u systeembeheerder en kunt u `rpm` en `dpkg` volledig gebruiken. Ik zal nu eerst laten zien hoe `rpm` wordt gebruikt. Daarna komt `dpkg` aan de orde.

Hoe rpm te gebruiken?

Om goed met `rpm` te kunnen werken, is het handig te weten hoe de naam van een `rpm` pakket is samengesteld. Zo'n naam heeft het volgende formaat:

pakketnaam-versie-uitgave.architectuur.rpm

Bijvoorbeeld:

athena-devel-2.4.3-87.i586.rpm

Hierin is *athena-devel* de naam van het pakket, *2.4.3* het versienummer en *87* het uitgavenummer. Het pakket is geschikt voor *i586* (Intel-processor van de 586-familie: Pentium) en beter. Welke architectuur u hebt, kunt u achterhalen door het programma *arch* te draaien, of

uname met de optie *-m* (machine): *uname -m*. Op een 80486 kunt u geen *i586* of *i686* (Pentium IV en hoger) pakket installeren, maar wel een *i386* pakket. Het is ook mogelijk dat een pakket voor elke architectuur geschikt is. In dat geval wordt *noarch* als aanduiding gebruikt, bijvoorbeeld

athena-images-1.2-3.noarch.rpm

Met het programma *rpm* beheert u `rpm` pakketten. Het voorbeeldpakket *athena-devel* installeert u met de opdracht

rpm -ivh athena-devel-2.4.3-87.i586.rpm

Opdracht *-i* staat voor *install*, de optie *v* (voor *verbose*) laat `rpm` veel informatie over het installatieproces geven, en de optie *h* (voor *hash symbols*) zorgt voor een voortgangsbalkje, zodat u kunt zien hoe ver `rpm` met installeren is gevorderd. Het is mogelijk dat `rpm` constateert dat niet aan de afhankelijkheden is voldaan, hetgeen wil zeggen dat niet alle voor het pakket benodigde software is geïnstalleerd. Het geeft dan een lijst van ontbrekende pakketten. Dit kan twee dingen betekenen:

- De genoemde software is inderdaad niet beschikbaar
- De genoemde software is wel beschikbaar, maar niet met `rpm` geïnstalleerd, zodat het niet in de RPM database voorkomt.

In het laatste geval kan het pakket zonder problemen worden geïnstalleerd, omdat alle benodigde software er is, ook al weet RPM dat niet. Om het pakket dan toch geïnstalleerd te krijgen moet de optie *--nodeps* (voer geen check op dependencies, afhankelijkheden, uit) worden gebruikt:

```
rpm -ivh --nodeps athena-devel-2.4.3-87.i586.rpm
```

Als een pakket met deze naam, maar eventueel een ander versie- en/of uitgavenummer, al op uw systeem staat, zal RPM weigeren te installeren. Als het al aanwezige pakket een lager versienummer of - bij hetzelfde versienummer - een lager uitgavenummer heeft kunt u *upgraden* door de opdracht **-U** in plaats van **-i** te gebruiken:

```
rpm -Uvh athena-devel-2.4.3-87.i586.rpm
```

Wilt u een hoger versie en/of uitgavenummer overschrijven, dan zult u de optie **--force** moeten gebruiken:

```
rpm -Uvh --force athena-devel-2.4.3-87.i586.rpm
```

U kunt de upgrade opdracht **-U** ook gebruiken als het pakket nog niet op het systeem staat. Het zal dan worden geïnstalleerd alsof de installatie opdracht **-i** werd gebruikt.

Een pakket wordt gedeïnstalleerd met de **-e** (van *erase*) opdracht, alleen gevolgd door de naam van het pakket:

```
rpm -e athena-devel
```

Er kan veel meer met rpm. Een uitgebreide handleiding voor het werken met rpm vindt u op www.redhat.com/. Gebruikers van een op RPM gebaseerde distributie kunnen met het commando *man rpm* ook de rpm manual pages raadplegen.

Hoe dpkg te gebruiken?

Net als bij RPM is de belangrijkste informatie over een Debian-pakket in zijn naam opgenomen. Die naam is als volgt samengesteld:

```
pakketnaam_versie-uitgave_architectuur.deb
```

Bijvoorbeeld:

```
athena-devel_2.4.3-87_i586.deb
```

Hierin is *athena-devel* de naam van het pakket, *2.4.3* het versienummer en *87* het uitgavenummer. Het pakket is geschikt voor *i586* (Intel-processor van de 586-familie: Pentium) en beter. Welke architectuur u hebt, kunt u achterhalen door het programma *arch* te draaien. Op een 80486 kunt u bijvoorbeeld geen *i586* of *i686* (Pentium IV en hoger) pakket installeren, maar wel een *i386* pakket. Het is ook mogelijk dat een pakket voor elke architectuur geschikt is. In dat geval wordt *all* als aanduiding gebruikt, bijvoorbeeld

```
athena-images_1.2-3_all.deb
```

De architectuuraanduiding wordt echter niet altijd gebruikt, zij het wel door Debian zelf.

Met het programma *dpkg* beheert u Debian-pakketten. Het voorbeeldpakket *athena-devel* installeert u met de opdracht

```
dpkg -i athena-devel_2.4.3-87_i586.deb
```

Actie **-i** staat voor *install*. Als er al een oudere versie of een oudere uitgave van deze versie van het pakket op het systeem staat wordt het automatisch ge-upgrade. Het is mogelijk dat *dpkg* constateert dat niet aan de afhankelijkheden is voldaan, hetgeen wil zeggen dat niet alle voor het pakket benodigde software is geïnstalleerd. Het geeft dan een lijst van ontbrekende pakketten. Dit kan twee dingen betekenen:

- De genoemde software is inderdaad niet beschikbaar
- De genoemde software is wel be-

schikbaar, maar niet met of via dpkg geïnstalleerd, zodat het niet in de pakket-database voorkomt.

In het laatste geval kan het pakket zonder problemen worden geïnstalleerd, omdat alle benodigde software er is, ook al weet dpkg dat niet. Om het pakket dan toch geïnstalleerd te krijgen moet de optie `--force` met het argument `depends` (stop het commando niet als niet aan afhankelijkheden is voldaan) worden gebruikt:

```
dpkg --force depends -i athena-
devel_2.4.3-87_i586.deb
```

Wilt u een hoger versie- en/of uitgave-nummer overschrijven, dan zult u bij `--force` het argument `downgrade` moeten gebruiken:

```
dpkg --force downgrade -i athena-
devel_2.4.3-87_i586.deb
```

Een pakket wordt gedeïnstalleerd met de actie `--purge`, alleen gevolgd door de naam van het pakket:

```
dpkg --purge athena-devel
```

of als u de configuratiebestanden wilt behouden:

```
dpkg -r athena-devel
```

waarbij `-r` staat voor *remove*.

Er kan veel meer met dpkg. Een beknopte handleiding voor het pakketbeheer van Debian waarin ook de verschillende mogelijkheden van dpkg aan de orde komen, vindt u op www.debian.org/. Gebruikers van Debian of een er van afgeleide distributie kunnen met het commando `man dpkg` ook de dpkg manual pages raadplegen. Vooral voor beginners is het handig om het menugestuurde `dselect` te gebruiken. Een be-

ginnershandleiding voor het gebruik van `dselect` vindt u eveneens op de Debian website (5).

Opdrachtregel of niet?

Ik hoop dat deze ervaring op de opdrachtregel u goed bevallen is. In moderne Linux distributies is het eigenlijk niet meer nodig om de opdrachtregel te gebruiken. Als u echter eenmaal weet hoe het moet, gaan veel dingen via de opdrachtregel gewoon sneller en eenvoudiger. De echte Linux-kenner doet het op de opdrachtregel, maar laat u er door deze uitspraak niet van weerhouden Linux gewoon lekker comfortabel grafisch te gebruiken.

Dan is nu de beurt aan Autopackage.

Autopackage

Autopackage (6) is een vrij recent, veelbelovend universeel systeem voor verspreiding en installatie van Linux-software. Een autopackage-pakket, kortweg een autopackage, heeft een naam met extensie *package*. U kunt het op elke distributie probleemloos installeren. De manier waarop dat gaat, lijkt erg op het installeren van software onder Windows. Er zijn verschillende frontends beschikbaar: grafisch (GTK voor Gnome en Qt voor KDE) en de opdrachtregel. Autopackage ondersteunt meerdere talen. In tegenstelling tot software-pakketten voor Windows is het niet zo dat een autopackage altijd alle ondersteunende software bevat waarvan het afhankelijk is. Als u een autopackage installeert, kijkt het installatieprogramma of de benodigde software al op uw systeem staat, en downloadt en installeert het als dat niet zo is. Benodigde software

hoeft niet met autopackage zelf te zijn geïnstalleerd, als het er maar is. Jammer is dat autopackage nog niet met de eigen pakketbeheerder van de distributie (bijvoorbeeld RPM of DPKG) is geïntegreerd, maar het ligt in de bedoeling dit in een volgende uitgifte op te lossen.

Autopackage is niet bedoeld om de eigen pakketbeheersystemen van Linux-distributies te vervangen, maar om gemakkelijk extra software te kunnen verspreiden en te installeren ongeacht de distributie. Al voor verscheidene softwareprojecten, waaronder de browser Firefox en het beeldbewerkingsprogramma The Gimp, zijn inmiddels autopackages beschikbaar, en hun aantal neemt nog toe. Ook onze eigen Belastingdienst heeft autopackage ontdekt: het aangifteprogramma voor Linux verschijnt eveneens als autopackage.

De autopackage-website onderhoudt een lijst van beschikbare autopackages: <http://www.autopackage.org/packages/> Deze lijst is zeker niet compleet. Zo staat (vanzelfsprekend) het aangifteprogramma van de Belastingdienst er niet in.

Stel dat u een autopackage hebt gedownload en het wil installeren. Hoe gaat dat in zijn werk?

Laten we het eerst maar eens doen vanaf de opdrachtregel. U hebt daar toch al mee kennisgemaakt, en daar komt bij dat het gewoon de eenvoudigste manier is. Omdat een autopackage in een installatiescript is ingebed, kan het pakket vanaf de opdrachtregel worden geïnstal-

leerd door het script uit te voeren, bijvoorbeeld:

```
bash ./athena-devel-2.4.3-87.x86.package
```

Als u autopackage voor de eerste keer gebruikt, zal het eerst zijn ondersteu-



ningssoftware (met de gebruikersinterface) willen downloaden en installeren en vraagt u daar netjes toestemming voor (7). Als dat afgerond is, zal het gewenste pakket worden geïnstalleerd. Met welke gebruikersinterface u dan te maken krijgt, hangt af van de omgeving waarbinnen u werkt. Als u het pakket in een echte tekstterminal (tty) installeert, laat autopackage u op het scherm van de terminal de verschillende stappen van de installatie zien. Doet u het daarentegen in een grafische terminal (zoals konsole of gnome-terminal), dan wordt de passende grafische gebruikersinterface gestart en bent u weer van de opdrachtregel af. Volg eventuele instructies hoe dan ook gewoon op, en na korte

tijd zal het programma geïnstalleerd zijn.

Verwijderen van een pakket gaat vrijwel even eenvoudig met het speciale programma *package*. Met de opdracht:

```
package remove athena-devel
```

kunt u uw systeem weer van het zojuist geïnstalleerde pakket ontdoen. Let wel dat nu alleen de basisnaam wordt gegeven, zonder extensie, architectuur, en uitgifte- en versienummer.

Wilt u toch liever helemaal langs grafische weg installeren, geen nood, ook dat kan. U start dan bijvoorbeeld vanuit Konqueror (KDE) of Nautilus (Gnome). Ga naar de directory waarin het pakket staat, klik met de rechter muisknop op het icoontje van het pakket, en kies voor *Eigenschappen*. Er verschijnt dan een pop-up-venster met *Eigenschappen*. Ga naar het tabblad *Toegangsrechten* en zet een vinkje in de checkbox met de titel *is uitvoerbaar* of een tekst van gelijke strekking. Als er zo'n checkbox is voor zowel de eigenaar, de groep als voor anderen, vinkt u alleen die van de eigenaar aan. In afbeelding 1 ziet u hoe dit in Konqueror 3.5.2 gaat.

Vervolgens opent u het pakket via het rechter-muisknop-menu of door met de linker-muisknop op het icoontje te klikken. Als u dit voor de eerste keer doet, zal een terminal-venster (ja ja, u ontkomt er toch niet helemaal aan...) wor-

den geopend, waarin u wordt gevraagd of de ondersteuningssoftware met de juiste interface mag worden gedownload en geïnstalleerd. Wanneer dat na uw toestemming achter de rug is, begint de eigenlijke installatie van het pakket.



Daarbij kunt u de muis weer gebruiken. Bij volgende installaties start meteen de grafische interface en hebt u met de opdrachtregel niets meer te maken.

Volg verder eventuele instructies, en na korte tijd is het programma geïnstalleerd. In afbeelding 2 (hierboven) ziet u dat Autopackage u volledig op de hoogte houdt van de stand van zaken. Verwijderen van een eenmaal geïnstalleerd pakket is zo mogelijk nog eenvoudiger. Bij installatie van de ondersteuningssoftware wordt onder *Systeem* ook een menu-ingang in het KDE of Gnome menu gemaakt met de titel *Manage 3rd party software*. Klik daarop en u zal een lijst van met autopackage geïnstalleerde software worden getoond. Zie afbeelding 3 (hierna).

Klik op het te verwijderen pakket en vervolgens op de knop *Remove*. Dat is alles.



Uitleiding

In deze aflevering hebt u het basisge-reedschap van de belangrijkste Linux-systemen voor softwarebeheer, RPM en DPKG, leren gebruiken. In allerlei gevallen kan dit een handige aanvulling zijn op het gebruik van een grafische gebruikersinterface, zoals Synaptic voor APT.

Bovendien hebt u kennis gemaakt met Autopackage, een nieuw systeem voor het gemakkelijk verspreiden en installeren van extra software, zoals het aangifteprogramma van de Belastingdienst. In de media heeft het systeem al de aandacht getrokken. Zo verscheen er vorig jaar in het augustusnummer van het Engelse tijdschrift *Linux Format* een positief artikel over Autopackage. Ik ben benieuwd of het echt wortel schiet.

Een volgende keer kunnen we het eens hebben over een ander nieuw systeem

voor de verspreiding van software: *Klik*, te vinden op www.atekon.de. Het is geïnspireerd door MacOS. Bij aanklikken van een programma op de Klik-server

wordt a.h.w. een recept aangemaakt, waarmee het programma op uw systeem kan draaien. Op basis van dit recept wordt een Klik-pakket aangemaakt. Na downloaden wordt het op uw bureaublad geplaatst en is het direct te gebruiken zonder dat installatie nodig is. Als u de software niet meer nodig hebt, hoeft u alleen maar het Klik-pakket te verwijderen. Erg handig voor het even proberen van software!

Noten

1. www.redhat.com/docs/books/max-rpm/
2. www.linuxbase.org/
3. www.pathname.com/fhs/pub/fhs-2.3.html
4. www.debian.org/doc/manuals/quick-reference/ch-package.en.html
5. www.debian.org/doc/manuals/dselect-beginner/
6. www.autopackage.org
7. Zo u wilt kunt u die ondersteuningssoftware al van te voren installeren. In <http://ftp.sunsite.dk/projects/autopackage/1.0.9/>, een mirror van de autopackage site, vindt u het basispakket als rpm en als tarball en de GTK-gebruikersinterface als autopackage.

