

Génération de paquets Linux

Génération de paquets RPM

Documentation officielle

- <http://www.rpm.org/max-rpm/>

Spec file sections

Explication des différents mots-clés utilisables dans un fichier spec. Il est très probable que l'utilisation tel quel de ces exemples ne fonctionne pas. Tous les mots-clés ne sont pas utilisables en même temps. L'objectif de cette partie est de présenter les possibilités. Pour des exemples complets se reporter à la section [des fichiers spec exemples](#).

Preamble

```
#
# Example spec file
#
Summary: A CD player
Name: cdplayer
Version: 1.0
Release: 1
License: GPL
Group: Applications/Sound
Source: ftp://....
URL: http://siteofficiel.fr
Distribution: Red Hat Enterprise Linux
Icon: foo.xpm
Vendor: Societe
Packager: nom packager
%description
Le logiciel decrit sur plusieurs lignes sans soucis
pour peu qu'on a des choses a dire
```

Prep section

Dans la prep section, on peut utiliser une macro qui décompresse un tar.gz et faire un cd dans les sources.

```
%prep //first script rpm executes during a build
%setup //macro pour décompresser l'archive tar des sources
```

Build section

```
%build //execute when %prep script left off  
make
```

Install section

```
%install //to do whatever is necessary to actually install the newly built  
software  
make install
```

Verify script

```
%check //run test suite of the built software to ensure that the binaries  
work correctly  
make test or make check
```

Clean section

```
%clean //to clean up the software's build directory tree  
rm -rf $RPM_BUILD_ROOT
```

Install/uninstall scripts

```
%pre //execute before installation  
  
%post //executed after installation  
  
%preun //executed before the package has been removed  
  
%postun //executed after the package has been removed
```

File list

```
%files //which files on the build system are to be packaged  
  
%doc //flags the filename(s) that follow, as being documentation  
  
%config //flags the specified file as being a configuration file  
  
%attr //permits finer control over three key file attributes (permission,  
user and group)  
%attr(<mode, <user>, <group>) file  
%attr(755, root, root) foo.bar
```

```
%attr(755, -, root) foo.bar

%defattr //allows setting of default attributes for files and directives
%defattr(<file mode>, <user>, <group>, <dir mode>)

%ghost //files not add to the package
%verify //ability to verify the integrity of the software it has installed
is impressive
%docdir //add directory to the list of directories that will contain
documentation
%dir //add the content of that directory but just files, not others sub-
directories
```

Complete spec file example

Cette section comprend des exemples de fichiers spec fonctionnel à utiliser tel quel.

Fichier spec pour un paquet vide à utiliser pour marquer une installation dans la base RPM (fonctionnel).

```
Summary: Software for <name_projet> project
Name: <softname>
Version: <softversion>
Release: el5.6
License: GPLv3+
Group: Applications/System
Vendor: <provider_name>
Packager: system team
%description
<description_with_multi_lines_possibility>

%files
```

Fichier spec pour un paquet qui contient une archive qu'on souhaite décompresser et réaliser une action d'installation (fonctionnel).

```
Summary: Software for <name_projet> project
Name: <softname>
Version: <softversion>
Release: el5.6
License: GPLv3+
Group: Applications/System
BuildRoot: /usr/src/redhat/SOURCES/
Prefix: /<softname>V<softversion>
%description
Description de <softname>



```
%pre
```


```

```
%prep

%build

%install

%post
cd /<softname>V<softversion>/version
/bin/tar xfz <softname>V<softversion>.tgz

touch /<softname>V<softversion>/test

%postun
rm -rf /<softname>V<softversion>

%clean

%files
%defattr(-,root,root)
/<softname>V<softversion>
/<softname>V<softversion>/version
/<softname>V<softversion>/version/<softname>V<softversion>.tgz
```

Fichier spec présenté dans max-rpm mais non testé par mes soins.

```
#
# Example spec file
#
Summary: A CD player
Name: cdplayer
Version: 1.0
Release: 1%{?dist}
License: GPL
Group: Applications/Sound
Source: ftp://....
URL: http://siteofficiel.fr
Distribution: Red Hat Enterprise Linux
Icon: foo.xpm
Vendor: Societe
Packager: nom packager
BuildRoot: %[_tmppath]/%{name}-%{version}-%{release}-root-%(%{__id_u} -n)
%description
Le logiciel decrit sur plusieurs lignes sans soucis
pour peu qu'on a des choses a dire
```

spec complet qui m' inspiré et a l'air fonctionnel mais à tester.

```
Name: npm
Version: 2.7.0.100
Release: 1%{?dist}
Summary: The "NPM-MPS System"
```

```
Group: Applications/Text
License: GPLv3+
Source0: npm_2.7.0-100.tar.gz
BuildRoot: %{_tmppath}/%{name}-%{version}-%{release}-root-%(%{__id_u} -n)

%description
this is only to test for mb and mpe

%files
/usr/bin/mb
/usr/bin/mpe
```

rpmrc file

Afficher les valeurs retenues dans le contexte dans lequel est exécuté la commande.

```
rpm --showrc
```

Ordre hiérarchique de prise en compte des différents fichiers de configuration.

- /usr/lib/rpmrc : file always read, do not edit.
- /etc/rpmrc : is fair game for modifications and additions
- ~/.rpmrc
- In the command line : -rcfile

Changer les valeurs des chemins par défaut.

```
buildroot //default : none
topdir //the path of the top-level directory tree. The following entries
base their default values on the value of topdir.
* sourcedir //default : <topdir>/SOURCES
* specdir //default : <topdir>/SPECS
* builddir //default : <topdir>/BUILD
* rpmdir //default : <topdir>/RPMS
* srcrpmdir //default : <topdir>/SRPMS
```

Génération RPM

Pour générer des RPM, utiliser un compte utilisateur de base. L'utilisation du compte root est à proscrire. Néanmoins pour cela, l'arborescence de création de paquet (forge) doit disposer du bon propriétaire. Par défaut sous Red Hat, c'est le dossier /usr/src/redhat qui est apporté par rpm-build qui est utilisé pour générer.

```
chown -R <user_de_base:<user_de_base> /usr/src/redhat
```

Exécuter les instructions %prep, %build, %install, %check et génération du paquet rpm binaire et rpm source.

```
cd /usr/src/redhat
rpmbuild -ba <chemin fichier spec>
```

Exécuter les instructions %prep, %build, %install, %check et génération uniquement du paquet rpm binaire.

```
cd /usr/src/redhat
rpmbuild -bb <chemin fichier spec>
```

Extraction du contenu d'un paquet rpm

Avec un paquet RPM construit, il peut être intéressant d'en extraire le contenu sans l'installer.

Extraire le contenu du paquet php-common. Pour extraire le contenu, les dossiers de destinations doivent être créés en local au préalable.

```
mkdir extraction
cd extraction
mkdir etc
rpm2cpio php-common-5.3.3-1.el5.remi.x86_64.rpm | cpio --extract
```

Avec ceci, on obtient que le contenu du dossier etc du paquet, les autres dossiers vont émettre des erreurs. Dans notre cas, on ne souhaite que la configuration, le reste ne nous intéresse pas.

On peut aussi utiliser la commande rpm2tgz pour convertir le paquet RPM en archive tar.gz et l'exploiter ensuite avec la commande tar. S'il s'agit d'un paquet RPM binaire, on ne pourra pas consulter les opérations effectuées lors de l'installation ou la désinstallation (l'équivalent des scripts d'installation ou de désinstallation dans d'autres systèmes de gestion de paquet) : pour cela, vous devez extraire le fichier SPEC du paquet RPM source et y lire les sections {pre,post}-install et {pre,post}-uninstall.

From:
<https://wiki.ouieuhoutca.eu/> - **kilsufi de noter**

Permanent link:
https://wiki.ouieuhoutca.eu/generation_paquets_linux

Last update: **2021/01/21 21:42**

