

Gestion des logiciels

Table des matières

1. Généralités	1
2. RPM : RedHat Package Manager	2
2.1. Commande rpm	2
3. YUM : Yellow dog Updater Modified	5
3.1. Commande yum	5
3.2. Fonctionnement de YUM	6
4. Gérer son dépôt	7
5. Le dépôt EPEL	8
5.1. Installation	8

Chapitre 1. Généralités

Sur un système Linux, il est possible d'installer un logiciel de deux façons :

- en utilisant un paquet d'installation ;
- en compilant les fichiers sources.

Le paquet : Il s'agit d'un unique fichier comprenant toutes les données utiles à l'installation du programme. Il peut être exécuté directement sur le système à partir d'un dépôt logiciel.

Les fichiers sources : Certains logiciels ne sont pas fournis dans des paquets prêts à être installés mais via une archive contenant les fichiers sources. Charge à l'administrateur de préparer ces fichiers et de les compiler pour installer le programme.

Chapitre 2. RPM : RedHat Package Manager

RPM (RedHat Package Manager) est un système de gestion des logiciels. Il est possible d'installer, de désinstaller, de mettre à jour ou de vérifier des logiciels contenus dans des paquets.

RPM est le format utilisé par toutes les distributions à base RedHat (Fedora, CentOS, SuSe, Mandriva, ...). Son équivalent dans le monde de Debian est DPKG (Debian Package).

Le nom d'un paquet RPM répond à une nomenclature précise :

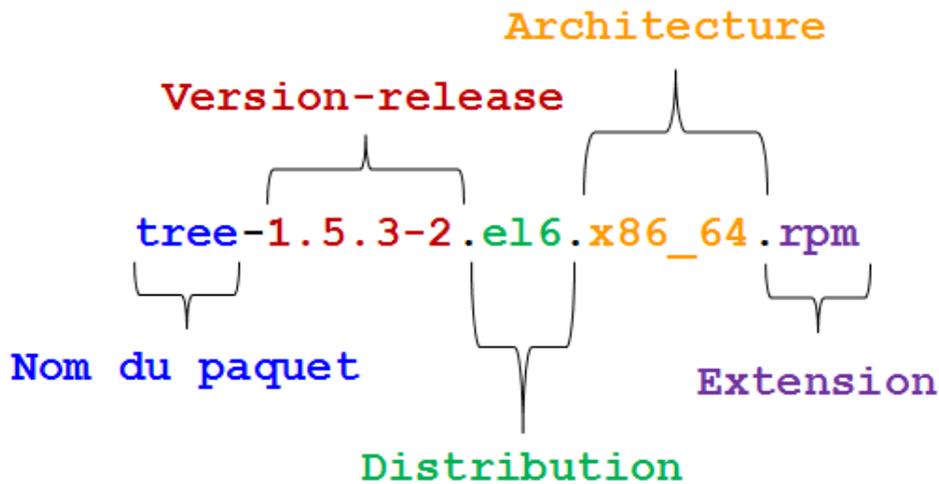


Figure 1. `nom-version-edition.architecture.rpm`

2.1. Commande rpm

La commande `rpm` permet d'installer un paquet.

Syntaxe de la commande rpm

```
rpm [-i][-U] paquet.rpm [-e] paquet
```

Exemple :

```
[root]# rpm -ivh paquet.rpm
```

Table 1. Options de la commande rpm

Option	Description
<code>-i paquet.rpm</code>	Installe le paquet.
<code>-U paquet.rpm</code>	Met à jour un paquet déjà installé.
<code>-e paquet.rpm</code>	Désinstalle le paquet.
<code>-h</code>	Affiche une barre de progression.

Option	Description
-v	Informe sur l'avancement de l'opération.
--test	Teste la commande sans l'exécuter.

La commande `rpm` permet aussi d'interroger la base de données des paquets du système en ajoutant l'option `-q`.

Il est possible d'exécuter plusieurs types de requêtes pour obtenir différentes informations sur les paquets installés. La base de donnée RPM se trouve dans le répertoire `/var/lib/rpm`.

Exemple :

```
[root]# rpm -qa
```

Cette commande interroge tous les paquets installés sur le système.

Syntaxe de la commande rpm pour interrogation

```
rpm -q [-a][-i][-l] paquet [-f] fichier
```

Exemple :

```
[root]# rpm -qil paquet
[root]# rpm -qf /chemin/fichier
```

Table 2. Options de la commande rpm pour interrogation

Option	Description
-a	Liste tous les paquets installés sur le système.
-i <i>paquet</i>	Affiche les informations du paquet.
-l <i>paquet</i>	Liste les fichiers contenus dans le paquet.
-f	Affiche le nom du paquet contenant le fichier précisé.
--last	La liste des paquets est donnée par date d'installation (les derniers paquetages installés apparaissent en premier).



Après l'option `-q`, le nom du paquet doit être exact. Les métacaractères ne sont pas gérés.



Il est cependant possible de lister tous les paquets installés et de filtrer avec la commande `grep`.

Exemple : lister les derniers paquets installés :

```
sudo rpm -qa --last | head
kernel-devel-3.10.0-514.21.2.el7.x86_64      mar. 27 juin 2017 14:52:21 CEST
webmin-1.850-1.noarch                        mar. 27 juin 2017 14:51:59 CEST
kernel-3.10.0-514.21.2.el7.x86_64          mar. 27 juin 2017 14:51:32 CEST
sudo-1.8.6p7-23.el7_3.x86_64               mar. 27 juin 2017 14:51:23 CEST
python-perf-3.10.0-514.21.2.el7.x86_64     mar. 27 juin 2017 14:51:22 CEST
kernel-tools-3.10.0-514.21.2.el7.x86_64    mar. 27 juin 2017 14:51:22 CEST
glibc-headers-2.17-157.el7_3.4.x86_64      mar. 27 juin 2017 14:51:22 CEST
glibc-devel-2.17-157.el7_3.4.x86_64        mar. 27 juin 2017 14:51:22 CEST
kernel-headers-3.10.0-514.21.2.el7.x86_64  mar. 27 juin 2017 14:51:21 CEST
kernel-tools-libs-3.10.0-514.21.2.el7.x86_64 mar. 27 juin 2017 14:51:20 CEST
```

Exemple : lister l'historique d'installation du kernel :

```
sudo rpm -qa --last kernel
kernel-3.10.0-514.21.2.el7.x86_64          mar. 27 juin 2017 14:51:32 CEST
kernel-3.10.0-514.21.1.el7.x86_64         mar. 20 juin 2017 10:31:38 CEST
kernel-3.10.0-514.16.1.el7.x86_64        lun. 17 avril 2017 09:41:52 CEST
kernel-3.10.0-514.10.2.el7.x86_64        lun. 27 mars 2017 07:07:10 CEST
kernel-3.10.0-514.6.1.el7.x86_64         dim. 05 févr. 2017 18:40:50 CET
```

Chapitre 3. YUM : Yellow dog Updater Modified

YUM est un gestionnaire de paquets logiciels. Il fonctionne avec des paquets RPM regroupés dans un dépôt (un répertoire de stockage des paquets) local ou distant.

La commande **yum** permet la gestion des paquets en comparant ceux installés sur le système à ceux présents dans les dépôts définis sur le serveur. Elle permet aussi d'installer automatiquement les dépendances, si elles sont également présentes dans les dépôts.

YUM est le gestionnaire utilisé par de nombreuses distributions à base RedHat (Fedora, CentOS, ...). Son équivalent dans le monde Debian est APT (Advanced Packaging Tool).

3.1. Commande yum

La commande yum permet d'installer un paquet en ne spécifiant que le nom court.

Syntaxe de la commande yum

```
yum [install][remove][list all][search][info] paquet
```

Exemple :

```
[root]# yum install tree
```

Seul le nom court du paquet est nécessaire.

Table 3. Options de la commande yum

Option	Description
<i>install</i>	Installe le paquet.
<i>remove</i>	Désinstalle le paquet.
<i>list all</i>	Liste les paquets déjà présents dans le dépôt.
<i>search</i>	Recherche un paquet dans le dépôt.
<i>provides */nom_cmde</i>	Recherche une commande.
<i>info</i>	Affiche les informations du paquet.

La commande **yum list** liste tous les paquets installés sur le système et présents dans le dépôt. Elle accepte plusieurs paramètres :

Table 4. Paramètres de la commande yum list

Paramètre	Description
<i>all</i>	Liste les paquets installés puis ceux disponible sur les dépôts.
<i>available</i>	Liste uniquement les paquets disponible pour installation.
<i>updates</i>	Liste les paquets pouvant être mis à jour.
<i>obsoletes</i>	Liste les paquets rendus obsolètes par des versions supérieures disponibles.
<i>recent</i>	Liste les derniers paquets ajoutés au dépôt.

Exemple de recherche de la commande semanage :

```
[root]# yum provides */semanage
```

3.2. Fonctionnement de YUM

Sur un poste client, le gestionnaire YUM s'appuie sur un ou plusieurs fichiers de configuration afin de cibler les dépôts contenant les paquets RPM.

Ces fichiers sont situés dans `/etc/yum.repos.d/` et se terminent obligatoirement par `.repo` afin d'être exploités par YUM.

Exemple :

```
/etc/yum.repos.d/MonDepotLocal.repo
```

Chaque fichier `.repo` se constitue au minimum des informations suivantes, une directive par ligne. Exemple:

```
[DepotLocal]    #Nom court du dépôt
name=Mon dépôt local    #Nom détaillé
baseurl=http://..... ou file:///.....    #Adresse http ou local
enabled=1    #Activation =1, ou non activé =0"
gpgcheck=1    #Dépôt demandant une signature
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-6    #Chemin de la clef publique GPG
```

Par défaut, la directive **enabled** est absente ce qui signifie que le dépôt est activé. Pour désactiver un dépôt, il faut spécifier la directive **enabled=0**.

Chapitre 4. Gérer son dépôt

La création d'un dépôt permet de disposer de sa propre banque de paquets. Celle-ci peut-être disponible par exemple par point de montage ou mise à disposition sur un serveur web.

Les étapes de la création d'un dépôt sur un serveur sont les suivantes :

- Créer un répertoire qui va accueillir tous les paquets rpm ;

```
[root]# mkdir /MonDepot
```

- Copier tous les paquets rpm nécessaires dans ce dossier ;

```
[root]# cp ../*.rpm /MonDepot/
```

- Créer la structure et générer le dépôt à l'aide de la commande **createrepo** ;

```
[root]# createrepo /MonDepot
```

- Configurer les fichiers **.repo** des clients afin qu'ils puissent installer les paquets depuis ce serveur (réinitialiser le cache des clients si besoin avec **yum clean all**).

Chapitre 5. Le dépôt EPEL

Le dépôt **EPEL** (**E**xtra **P**ackages for **E**nterprise **L**inux) est un dépôt contenant des paquets logiciels supplémentaires pour Enterprise Linux, ce qui inclut RedHat Enterprise Linux (RHEL), CentOS, etc.

5.1. Installation

Télécharger et installer le rpm du dépôt :

Si vous êtes derrière le proxy internet de l'école

```
[root]# export http_proxy=http://10.10.10.7:8080
```

- Pour une CentOS 6 :

```
[root]# rpm -ivh http://dl.fedoraproject.org/pub/epel/6/x86_64/epel-release-6-8.noarch.rpm
```

Après avoir installé le paquet RPM du dépôt :

```
[root]# yum update
```