

# Proseminar im Sommersemester 2008: Linux für Umsteiger Paketverwaltung von RPM Paketen

Philipp Stekl

Universität Heidelberg

13. Mai 2008

Betreuer: J. Kunkel, O. Mordvinova

## Übersicht

- 1 Was ist RPM?
  - Definition
  - Aufbau eines Pakets
- 2 Arbeiten mit Paketen
  - rpm, yum
  - Repositories
  - Grafische Helfer
- 3 Erstellen von RPM Paketen
  - Aufbau der .spec Datei
  - Verzeichnisstruktur für rpmbuild
  - Beispiel Kernel als RPM
- 4 Quellen
  - ManPages
  - Webseiten

**Definition:** RPM Package Manager (RPM, Akronym, ursprünglich Red Hat Package Manager) ist ein freies (GPL) Software-Paketverwaltungssystem entwickelt von der Firma Red Hat. Es umfasst das Paketformat und alle nötigen Programme, um RPM-Pakete zu erstellen, zu installieren und zu verwalten. RPMs (Dateiendung „.rpm“) sind Archive, die ein Softwarepaket enthalten (grob ähnlich dem .zip-Format). Das RPM-Format ist Teil der Linux Standard Base.

## Aufbau eines Pakets

- \*.src.rpm Source-RPM Paket(enthält .spec datei und tarball mit sourcecode)
- \*.rpm Binary-RPM Paket
- \*.delta.rpm Delta-RPM Paket(enthält binäre diff)

## Aufbau eines Pakets - Aufbau eines BinaryRPM

BinaryRPM Pakete bestehen aus vier Teilen.

- 1. Der Vorspann (Lead) enthält in erster Linie Informationen zum Dateityp
- 2. Im zweiten Teil stehen die MD5-Summen und digitale Unterschriften (Signature)
- 3. Der dritte Teil (Header) liefert wichtige Metadateien und -verzeichnisse
- 4. Der vierte Teil enthält das CPIO-Archiv

## Aufbau eines Pakets - Das CPIO Archiv entpacken

- Beispiel Mplayer-\*.rpm
- rpm2cpio MPlayer-1.0rc2-4.pm.0.i586.rpm cpio -i  
-make-directories
- etc und usr Verzeichnisse mit enthaltenen Dateien werden entpackt

## Aufbau eines Pakets - Verwaltung von RPM'S

- RPM benötigt eine Datenbank, um die Informationen, welche zu jedem Paket gehören abzuspeichern
- /var/lib/rpm ist die Datenbank gespeichert
- Dadurch kann gewährleistet werden, das die nötigen Abhängigkeiten installiert werden.

## Arbeiten mit Paketen - rpm, yum

- (De-)Installation, Update
- Versionsabfrage
- Signatur überprüfen
- Pakete suchen

## Arbeiten mit Paketen - (De-)Installation, Update

- rpm -i „Paketname“
- yum install „Paketname“
- rpm -U „Paketname“
- yum update „Paketname“
- rpm -e „Paketname“
- yum remove „Paketname“

## Arbeiten mit Paketen - Versionsabfrage

- rpm -q „Paketname“
  - rpm -qa Alle installierten Pakete anzeigen
  - rpm -qf Ermitteln, zu welchem Paket eine Datei gehört
  - rpm -ql Anzeigen aller im (bereits installierten) Paket enthaltenen Dateien
- yum info „Paketname“

## Arbeiten mit Paketen - Signatur überprüfen

- Die Signatur wird in der RPM Datei gespeichert
- `rpm -v --checksig „Paketname“`
- Der passende Schlüssel muss im Keyring vorhanden sein

## Arbeiten mit Paketen - Pakete suchen

- Http, Ftp Server
  - z.B. rpm.livna.org, rpm.pbone.net ...
- andere Medien
- Repositories

Repositories muss man erst seinem Paketmanager hinzufügen.

## Arbeiten mit Paketen - Repositories

- `/etc/yum.repos.d/` sind die Repositories gespeichert, also Dateien in denen die Urls für die Paketsammlungen und ein paar andere Infos enthalten sind
- Evtl noch GPG keys laden
- Oft gibt es RPMs die die obigen beiden Schritte für einen erledigen
- `yum search „string“`
- Oder Yast/Yumex etc...

## Arbeiten mit Paketen - Grafische Helfer

- Yast, Yumex und KYum
- Demo/Bilder zeigen

## Erstellen von RPM Paketen

- Aufbau der .spec Datei
- Verzeichnisstruktur für rpmbuild
- Beispiel Kernel als RPM

## Erstellen von RPM Paketen - Aufbau der .spec Datei

- Distributionsabhängig
- Sektionen mit festgelegter Bedeutung
- Verwendung von Makros häufig

## Erstellen von RPM Paketen - Aufbau der .spec Datei

- Die Präambel: Für Benutzer relevante Informationen
- Die Prep-Sektion: Befehle die den Quellcode-Dateibaum aufbauen
- Die Build-Sektion: Befehle um den Quellcode zu kompilieren
- Die Install-Sektion: Befehle um die Dateien an die entsprechenden Positionen zu kopieren.
- Die Files-Sektion: Kein Shellscrip; Hier werden Dateien angegeben, die in das RPM-Paket übernommen werden.

## Erstellen von RPM Paketen - Verzeichnisstruktur zur Erstellung

- SOURCES/ - enthält die Sourcen, nach Installation von SourceRPM
- SPECS/ - enthält die Spec-Files
- BUILD/ - enthält die Source-Verzeichnisse aus denen die Software gebaut wird.
- RPMS/ - enthält die Binary-Pakete
- SRPMS/ - enthält die Source-Pakete

## Erstellen von RPM Paketen - Beispiel Kernel als RPM

- Verzeichnisstruktur anlegen, und Datei `~/rpmmacros` erstellen: `rpmdev-setuptree`
- Kernelsource downloaden: `yumdownloader --source kernel`
- Benötigte Programme um Kernel zu kompilieren installieren: `yum-builddep kernel-2.6.24.4-64.fc8.src.rpm`

## Erstellen von RPM Paketen - Beispiel Kernel als RPM

- Kernelsource nach `~/rpmbuild/SOURCES` und `.spec` Datei nach `~/rpmbuild/SPECS` kopieren:  
`rpm -Uvh kernel-2.6.24.4-64.fc8.src.rpm`
- ins SPEC Verzeichnis wechseln `cd ~/rpmbuild/SPECS`
- Source vorbereiten, in dem die Prepare Sektion der `.spec` Datei durchlaufen wird:  
`rpmbuild -bp --target='uname -m' kernel.spec`

## Erstellen von RPM Paketen - Beispiel Kernel als RPM

- in das Verzeichnis der Kernelsource wechseln:  
`cd ~/rpmbuild/BUILD/kernel-2.6.24/linux-2.6.24.i686`
- Kernel konfigurieren: `make menuconfig`
- `.config` Datei kopieren:  
`cp .config ~/rpmbuild/SOURCES/config-i686`
- in das Verzeichnis der `.spec` Datei wechseln:  
`cd ~/rpmbuild/SPECS`

## Erstellen von RPM Paketen - Beispiel Kernel als RPM

- Öffne die kernel.spec datei, und ändere die Zeile: define buildid .local in define buildid .irgendwas
- Falls man einen Patch anwenden will, diesen auch in die .spec Datei eintragen.
- Schließlich : rpmbuild -bb --target='uname -m' kernel.spec um den Kernel zu bauen.

## Zusammenfassung

- rpm dient der Softwarepaketverwaltung bei Linux und Unixartigen Systemen
- (De)Installation, Update, Entfernen mit rpm oder Frontend wie yast, yum etc
- Erstellung von rpm Paketen: Verzeichnisstruktur für rpmbuild einrichten, .spec Datei schreiben und anschließend mit rpmbuild das rpm Paket erzeugen.

## ManPages

- man rpm
- man rpm2cpio
- man cpio
- man yum

## Webseiten

- [docs.fedoraproject.org/](https://docs.fedoraproject.org/)
- [de.opensuse.org/](https://de.opensuse.org/)
- [de.wikipedia.org/wiki/RPM\\_Package\\_Manager](https://de.wikipedia.org/wiki/RPM_Package_Manager)
- [www.linuxhaven.de/dlhp/HOWTO-test/DE-RPM-HOWTO-8.html](http://www.linuxhaven.de/dlhp/HOWTO-test/DE-RPM-HOWTO-8.html)
- [www.rpm.org/](http://www.rpm.org/)
- [www.linux-praxis.de/](http://www.linux-praxis.de/)