



- ✓ Vendor Independent
- ✓ Community Supported
- ✓ Respected Worldwide
- ✓ Distribution Neutral



1.102.5 Red Hat Package Manager (RPM und YUM)

Sie müssen in der Lage sein, Paket-Management auf Linux-Distributionen, die RPM / YUM für die Paketverwaltung verwenden, durchzuführen. Die Kenntnisse betreffen die (Neu-)Installation, das Update und das Entfernen von Paketen sowie das Abfragen von Status- und Versions-Informationen. Auch das Abfragen von Paket-Informationen wie Abhängigkeiten, Integrität und Signaturen sollte man beherrschen. Sie sollten auch in der Lage sein, zu bestimmen, welche Dateien von einem Paket zur Verfügung gestellt werden, und auch das Paket finden können, dem eine bestimmte Datei entstammt.

Die wichtigsten Dateien, Bezeichnungen und Anwendungen:

`rpm`

`rpm2cpio`

`/etc/yum.conf`

`/etc/yum.repos.d/`

`yum`

`yumdownloader`

Die meisten heutigen Linux-Distributionen nutzen das Paketformat von RedHat, weil es eine sehr weitreichende Verwaltung der installierten Software ermöglicht.



RPM-Verwaltung

Paketaufbau: **Programmname-Versionnummer.Architektur.rpm**
(Paketname-Paketversion.Architektur.rpm)

Jedes RPM-Paket enthält neben den zu installierenden Dateien (wie Debian-Pakete) vier Scripts, die jeweils vor und nach der Installation und Deinstallation ausgeführt werden. Zusätzlich existieren Informationen über das Paket und die Abhängigkeiten, die die darin enthaltenen Programme besitzen.

Im Verzeichnis `/usr/lib/rpm` liegen die verschiedenen Hilfsmittel, die das rpm-Programm benötigt und unter `/var/lib/rpm` finden sich die verschiedenen Datenbanken über die installierten Pakete.

Der RPM Package Manager ist ein freies (GPL) Software-Paketverwaltungssystem entwickelt von der Firma Red Hat.



Installieren und deinstallieren von Paketen

Installieren eines Paketes:

Um eine *.rpm-Datei zu installieren, wird folgender Befehl verwendet:

```
rpm -i | --install [Optionen] Paketdatei
```

`--nodeps` [Prüfung von Abhängigkeiten wird abgeschaltet.]

`--noscripts` [Es werden keine pre- oder postinstall Scripts abgearbeitet.]

`--test` [Die Installation wird nur simuliert, keine Dateien werden installiert.]

`--excludedocs` [Die im Paket enthaltenen Dokumentationen werden nicht mitinstalliert.]

`--replacepkgs` [Die Installation wird durchgeführt, auch wenn schon Teile installiert sind.]

`--replacefiles` [Die Installation wird auch durchgeführt, auch wenn dabei Dateien überschrieben werden, die von anderen Paketen stammen.]

`--oldpackage` [Erlaubt ein Upgrade von älteren Versionen als die bereits installierten.]

`--force` [Entspricht der Kombination von `--replacepkgs --replacefiles --oldpackage`.]



Deinstallieren und Upgrade von Paketen

Ein Paket wird mit folgendem Befehl deinstalliert:

```
rpm -e | --uninstall [Optionen] Paketname
```

`--nodeps` [keine Prüfung von Abhängigkeiten]

`--noscripts` [Es werden keine pre- oder postinstall Scripts abgearbeitet.]

`--test` [Die Deinstallation wird nur simuliert, keine Dateien werden gelöscht.]

`--allmatches` [Deinstalliert alle Pakete, auf die das Namensmuster zutrifft. In diesem Fall ist Paketname also ein Muster und nicht ein Paketnamen.]

Upgrade

```
rpm -U | --upgrade [Optionen] Paketdatei
```

[Optionen siehe Installieren]. Befehl wie beim Installieren benutzt, nur daß eventuell vorher existierende Versionen des Paketes vor der Installation entfernt werden.

```
rpm -F | --freshen [Optionen] Paketdatei
```

Installiert die angegebene Paketdatei nur, wenn eine ältere Version des selben Programms vorher schon installiert war. Dann Ablauf wie (-U)



Übung: Abfrage von Paketinformationen (rpm)

- ➔ Versionsnummer installierter Pakete erfahren...

```
rpm -qi Paketname
```

```
rpm --query --info Paketname
```

```
rpm -qa | sort | less (alle inst. Pakete sortiert)
```

- ➔ Versionsinformation eines nicht installierten RPM-Paketes...

```
rpm -qi -p Paketdatei
```

```
rpm --query --info --package Paketdatei
```

- ➔ Es kann auch der eingebaute Variablensubstitutionsmechanismus benutzt werden...

```
rpm -q Paketname --queryformat "%{VERSION}\n" (siehe auch: rpm --querytags)
```

- ➔ Welche Dateien liefert ein Paket?

```
rpm -ql Paketname
```

```
(falls nicht installiert: rpm -ql -p Paketdatei)
```

- ➔ Signaturen von RPM-Dateien prüfen...

```
rpm -Va
```

```
rpm --check-sig paket.rpm
```



rpm2cpio

Jemand möchte ein rpm-Paket untersuchen oder nur bestimmte Inhalte des Pakets verwenden

Vorgehensweise:

- ➔ `mkdir /tmp/rpm-package-name`
- ➔ `cd /tmp/rpm-package-name`

```
rpm2cpio /path/to/file-to-extract.rpm | cpio -dim
```

Ergebnis:

- ➔ Alle Dateien des rpm-Archives liegen nun im angelegten Verzeichnis zur weiteren Untersuchung / Verwendung vor.



YUM

YUM (Yellow dog Updater, Modified) ist ein RPM-Paketmanager der, wie apt(itude), automatisch Abhängigkeiten auflösen kann.

- ➔ Pakete installieren/updates/entfernen:
`yum install paket`
- ➔ Upgrade einzelner Pakete (nur bereits installierte Pakete):
`yum update paket`
- ➔ Upgrade des kompletten Systems:
`yum upgrade`
- ➔ Deinstallation von Paketen:
`yum remove paket`
- ➔ Paketsuche: Welches Paket enthält eine bestimmte Datei?
`yum provides dateiname`
- ➔ Gibt es ein Paket?:
`yum list paketname`



YUM (2)

- ➔ Informationen über verfügbare Pakete anzeigen:
`yum list available`
- ➔ Updatebare Pakete anzeigen:
`yum list updates`
- ➔ Heruntergeladene Pakete wieder löschen:
`yum clean packages`
- ➔ **yumdownloader**
rpm-Pakete aus YUM-Repositories herunterladen.
- ➔ Repositories verwalten
in **/etc/yum.conf**



Literatur und Quellen

- F. Kahlhammer, LPI Study-Guide, 2001
- Jefferey Dean, LPI in a Nutshell, O'Reilly 2001
- Steven Pritchard. »Hardware-HOWTO«, <http://www.tldp.org/HOWTO/Hardware-HOWTO>
- Anselm Lingnau u.a., LPI – Level 1, mitp 2004
- Linup Front, Linux-Grundlagen für Anwender und Administratoren, Linup Front 2005
- wikipedia
- Allgemein: man-Pages, HOWTOs, info-Pages,