

# Paketmanagement

Alexander Günther  
mailto:alex.guenther@gmail.com  
gpg:0xEBE4A38A

UnFUG  
<http://unfug.org>

16. Dezember 2006

# Übersicht

- 1 Einführung in das Paketmanagement
- 2 Werkzeuge
- 3 Konfiguration
- 4 Pakete intern
- 5 Lokalen Mirror erstellen
  - Lokaler Arch Mirror

# Aufgaben des Paketmanagements

- Installieren von Paketen
- Deinstallieren von Paketen
- Update
- Abhängigkeiten von Paketen auflösen
- Konflikte von Paketen auflösen

# Klassische Lösungsansätze

## Application Layer

- Jedes Program hat eigene (De-)Installations-Tools
- Wenn das Programm das nich sauber macht, ist Handarbeit angesagt...

## Built in System Utilities

- System V
  - Systemweite DB welches Paket welche Dateien installiert hat
  - keine Unterstützung beim compilieren der Pakete
- BSD ports
  - Automatisierung beim Download, entpacken, konfigurieren und kompilieren der Pakete
  - keine Systemweite DB

# Debian basierte Systeme

- dpkg – Werkzeug auf unterster Ebene (installieren, konfigurieren der Pakete)
- apt-get – holt Pakete, löst Abhängigkeiten auf, benutzt dpkg
- aptitude – apt-get++ mit ncurses-gui

# PRM basierte Systeme

- rpm – war gedacht als ein Kommando für alles
- yum – ein rpm Wrapper, kann Abhängigkeiten auflösen

# ArchLinux

- pacman – Standardsachen holen, installieren,...
- qpkg – query tool

# Gentoo

- emerge – das Standardtool
- eix/esync – query tools

# Debian Konfigurationsdateien

- /etc/apt/sources.list
- ...

# Fedora Konfigurationsdateien

- `/etc/yum.conf`
- `/etc/yum.repos.d/*` – Mirrorlists
- `/etc/yum/yum-updates.conf` – Autoupdates einstellungen
- `/etc/yum/pluginconf.d/*`
- ...

# Arch Konfigurationsdateien

- /etc/pacman.conf – PaketmanagerKonfiguration
  - /etc/pacman.d/current
  - /etc/pacman.d/extra
  - /etc/pacman.d/community
  - ...
- /etc/makepkg.conf – Paketerstellung

# Gentoo Konfigurationsdateien

- /etc/make.conf
- /etc/portage/package.keywords
- /etc/portage/package.unmask
- /etc/portage/package.use
- ...

## .deb Paketinhalt

### centericq-utf8\_4.21.0-16\_i386.deb

- debian-binary
  - Version (meist 2.0)
- control.tar.gz
  - Metadaten
- data.tar.gz
  - das eigentliche Programm, etc.

## Namenstruktur der Debian Pakete

`centericq-utf8_4.21.0-16_i386.deb`

- centericq-utf8
  - Softwarename mit Prä- und Suffix (lib\*, \*-doc, \*-common)
- 4.21.0
  - Version
- -16
  - Release Nummer Debianintern  
(entfällt bei debianinternen Paketen)
- i386
  - Architektur (ppc, hppa, all)

## .rpm Paketinhalt

centericq-4.21.0-7.fc5.i386.rpm

- INFO/
  - Paketinfos
  - SCRIPTS/
    - ALL
    - PREIN
    - POSTIN
    - PREUN
    - POSTUN
- CONTENTS.cpio – Das gepackte Programm
- HEADER
- INSTALL
- UPGRADE

## .pkg.tar.gz (ArchLinux) Paketinhalt

centericq-4.21.0-3.pkg.tar.gz

- .PKGINFO
- .FILELIST
- .INSTALL (optional)
- Programmdateien

## .ebuild Paketinhalt

```
/usr/portage/net-im/centericq/
```

- centericq-4.21.0-r2.ebuild
- ChangeLog
- files/
  - diffs, patches
- Manifest
- metadata.xml

# Aufbau eines Mirros

- Paketliste
  - Packages.gz, current.db.tar.gz,
  - Portage tree
- Pakete, geordnet in Kategorien

# Paketkategorien

## Debian

- main – DFSL lizenzierte Pakete
- contrib – freie Pakete, die aber auf nichtfreie aufbauen
- non-free – nicht freie im sinne der DFSL

## Arch

- current
- extra
- community
- unstable
- testing

# Archmirror

- /var/cache/pacman/pkg per nfs freigeben

/etc/pacman.conf anpassen:

```
[current]
Server = file:///var/cache/pacman/pkg
...
[extra]
Server = file:///var/cache/pacman/pkg
...
```

Skript zum Mirror DB update, an Clients verteilen

```
cd /var/cache/pacman/pkg
wget -N ftp://ftp.archlinux.org/current/os/i686/current.db.tar.gz
wget -N ftp://ftp.archlinux.org/extra/os/i686/extra.db.tar.gz
```

# Links

- Debian Anwenderhandbuch
- ebuild HOWTO
- Apt HOWTO
- Debian Backports
- RPM Guide
- Archlinux Pacman