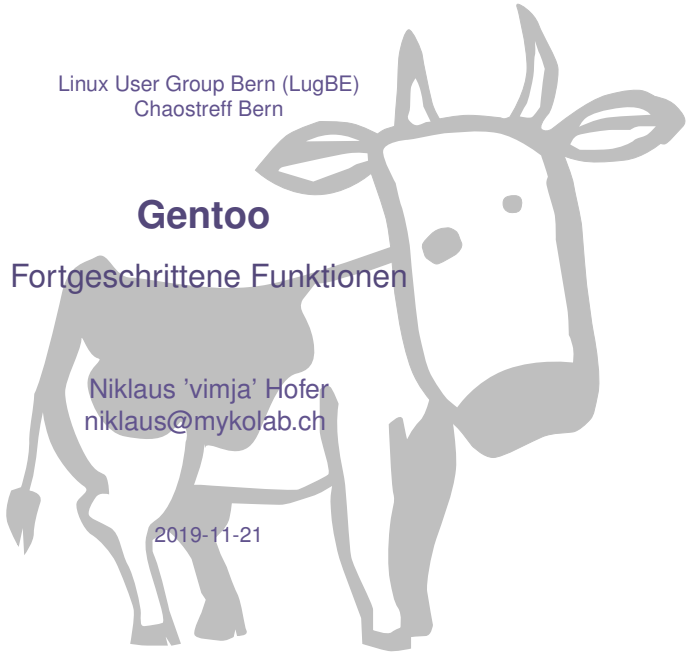
Linux User Group Bern (LugBE)
Chaostreff Bern

Gentoo

Fortgeschrittene Funktionen

Niklaus 'vimja' Hofer
niklaus@mykolab.ch

2019-11-21



/me

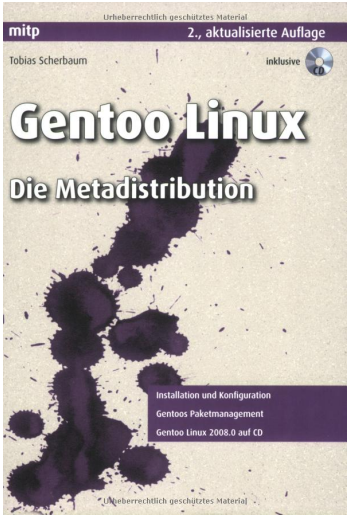
- Gentoo user seit 2012
  - beruflich seit 2016
- Mitglied der LugBE
- Mitglied im Chaostreff Bern
- Disclaimers

Content

Inhalt

Inhalt

Gentoo, Metadistribution



Was ist eine Metadistribution

- Werkzeug zum Bauen einer Distribution
- Kontrolle über ...
  - ... ausrichtung und Spezialisierung
  - ... Wahl zwischen Alternativen
  - ...
- Kurz: Kontrolle über alle Entscheidungen
- Jedes Gentoo ist anders
  - (oder doch nicht?)

Inhalt

Wie gehts das

- Portage
- Emerge
- ebuilds
  - Instruktionen zum Kompilieren, installieren von Software
  - Vergleichbar mit specfiles

Was macht Portage besonders

- Aus Qeallcode kompilieren
- Optimierungen
- Keywording
- use flags

use flags

- Global

```
# /etc/portage/make.conf
USE="${CPU_FLAGS_X86} -gnome -hal -aqua -bluetooth crypto cups ...
```

- Per Paket

```
# /etc/portage/package.use/firefox
www-client/firefox pgo lto hardened pulseaudio
```

use flags und Abhängigkeiten


```
# firefox-68.2.0.ebuild

CDEPEND="
...
        system-libevent? ( >=dev-libs/libevent-2.0:0=[threads] )
```

Inhalt

Zusammenfassung

- Flexbilität
- Jedes System ist anders
- Oftmals will man aber:
  - viele Systeme managen
  - diese sollen alle gleich sein
- Lasst uns sehen wie das geht...

Inhalt

Inhalt

ebuild repositories

- aka overlays
- git repository
- Kann weitere Element enthalten:
  - Profile
  - Paket Sets

management

- eselect repository
- layman
- Beide kennen eine grosse Zahl öffentlich verfügbarer Overlays

Inhalt

Sets

- Liste von Paketen
  - Eine Art Metapaket
- lässt sich mit einem Kommando installieren

System sets

- @system
- @world

```
# emerge --list-sets
changed-deps
deprecated-live-rebuild
downgrade
installed
live-rebuild
module-rebuild
perl-cleanup
```

Eigene Package Sets verwalten

- Dateien im Verzeichnis `/etc/portage/sets/`
- Via overlay
  - `sets.conf`

```
[vimja sets]
class = portage.sets.files.StaticFileSet
multiset = true
directory = ${repository:vimja-overlay}/sets/
```

Inhalt

Profile

- Wird bei der Installation gewählt
  - kann später geändert werden
- Enthält Konfigurationen für
  - eapi
  - packages
  - package.mask
  - make.defaults
  - package.use
  - use.force
  - use.mask
  - package.use.force
  - package.use.mask
  - ...

Standard Profile

- Standard Profile
  - Spezialisierungen

```
[1]   default/linux/amd64/17.0 (stable)
[2]   default/linux/amd64/17.0/selinux (stable)
[3]   default/linux/amd64/17.0/hardened (stable)
[4]   default/linux/amd64/17.0/hardened/selinux (stable)
[5]   default/linux/amd64/17.0/desktop (stable)
[6]   default/linux/amd64/17.0/desktop/gnome (stable)
[7]   default/linux/amd64/17.0/desktop/gnome/systemd (stable)
[8]   default/linux/amd64/17.0/desktop/plasma (stable)
[9]   default/linux/amd64/17.0/desktop/plasma/systemd (stable)
[10]  default/linux/amd64/17.0/developer (stable)
[11]  default/linux/amd64/17.0/no-multilib (stable)
[12]  default/linux/amd64/17.0/no-multilib/hardened (stable)
[13]  default/linux/amd64/17.0/no-multilib/hardened/selinux (stable)
```

Profile Vererbung

- parent
  - multi-parent

```
# default/linux/amd64/17.0/no-multilib/hardened/parent
..
../../../../../../features/hardened/amd64/no-multilib
```

Eigene Profile

- Vererbung nutzen, vorhandene Profile erweitern

```
# metadata/layout.conf
# permit usage of repo-names in parent-specification
profile-formats = portage-2

# profiles/vimja/parent
gentoo:default/linux/amd64/17.0/no-multilib/hardened
```

vimja's profile


```
# eselect profile list
  [35]   vimja-overlay:vimja/pc/laptop (stable)
  [36]   vimja-overlay:vimja/pc/desktop (stable)
  [37]   vimja-overlay:vimja/headless/server (stable)
  [38]   vimja-overlay:vimja/headless/router (stable)
```

Anwendung

- Zentrales Konfigurationsmanagement
- Spezialisierung von Profilen
- Versionierung

Inhalt

Inhalt

Abgrenzung

- Binäre Pakete
  - Binäre Pakete, vergleichbar mit anderen Distributionen
- `-bin` Pakete
  - Ein ebuild welcher eine vorkompilierte Software aus dem Upstream installiert

Paketaufbau

- xpak Format
- komprimiertes TAR Archiv mit Dateien
- Metadaten

Metadatenaufbau


```
# qtbz2 --split firefox-68.2.0.tbz2
# qxpak --list firefox-68.2.0.xpak
BUILD_TIME
CATEGORY
CBUILD
CC
CFLAGS
CHOST
CXX
CXXFLAGS
DEFINED_PHASES
DEPEND
...
firefox-68.2.0.ebuild
...
```

Binärpakete erstellen

- `quickpkg`

```
quickpkg www-client/firefox
```

- emerge mit `--buildpkg` oder `--buildpkgonly`

```
emerge --buildpkgonly www-client/firefox
```

Optionen beim Erstellen

- Kompressionsalgorithmus via `BINPKG_COMPRESS`
- Zielverzeichnis via `PKGDIR`
  - Standardmässig nach `/usr/portage/packages/`
  - Metadatenindex in der Datei `Packages`
- `FEATURES="binpkg-multi-instance"`

Anwendungsbeispiele für quickpkg

- FEATURES:
  - `unmerge-backup`
  - `downgrade-backup`

Installation


```
# /etc/portage/make.conf
FEATURES="getbinpkg"
PORTAGE_BINHOST="ssh://binpkg_host/usr/portage/packages"

# emerge --ask www-client/firefox
These are the packages that would be merged, in order:

Calculating dependencies... done!
[binary      U  ] www-client/firefox-68.2.0 [68.1.0]
```

- Option `--usepkg`

Installationsoptionen

- Übersteuern lokaler Einstellungen

```
!!! The following binary packages have been ignored due to non
    matching USE:
...
!!! The following binary packages have been ignored due to changed
    dependencies:
...
```

- Bildzeitabhängigkeiten entfernen?

```
emerge --ask --depclean --with-bdeps=n
```

Inhalt

Grundsätzliche Überlegungen

- x86 zu x86 geht immer
- Exakt gleiche CPU?
- Buildumgebung
  - nativ
  - chroot
  - VM
  - Crosscompile
- Rückwärtskompatible CPUs

Optimierung

- Wie viel Optimierung vs wie viel Aufwand

```
# gcc -march=native -E -v - </dev/null 2>&1 | grep cc1
/usr/libexec/gcc/x86_64-pc-linux-gnu/9.2.0/cc1 -E -quiet -v - -march=
    haswell -mmmx -mno-3dnow -msse -msse2 -msse3 -mssse3 -mno-sse4a -
    mcx16 -msahf -mmovbe -maes ...
```

- Gemeinsames vielfaches der Parameter?
- `-march=ivybridge`
  - Plus Cache sizes?

Ende

- Fragen?
- Install Party
  - Samstag, 2019-11-23 14:00
  - Chaostreff Bern, Kyburgstrasse 13, 3013 Bern