



AUR-Packages bauen und testen ohne ArchLinux im eigenen Docker-Container

Augsburger Linux Informationstag

6. April 2019

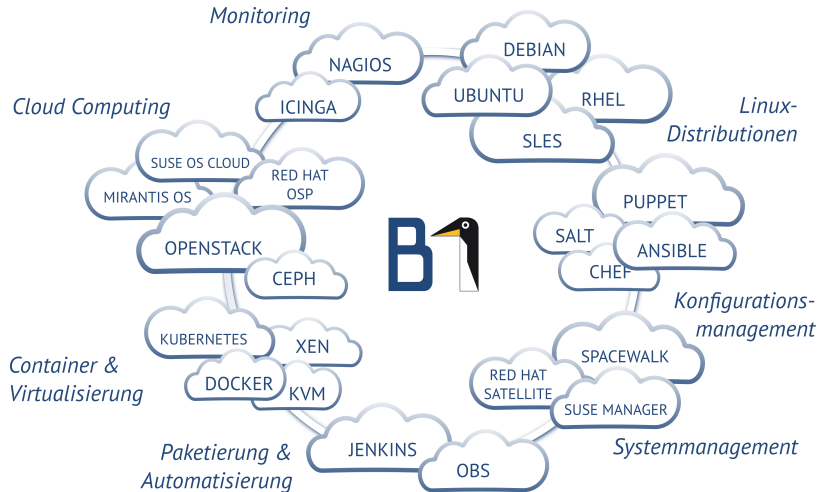


Anton Müller
Linux Consultant & Trainer
B1 Systems GmbH
anmueller@b1-systems.de

Vorstellung B1 Systems

- gegründet 2004
- primär Linux/Open Source-Themen
- national & international tätig
- über 100 Mitarbeiter
- unabhängig von Soft- und Hardware-Herstellern
- Leistungsangebot:
 - Beratung & Consulting
 - Support
 - Entwicklung
 - Training
 - Betrieb
 - Lösungen
- Standorte in Rockolding, Köln, Berlin & Dresden

Schwerpunkte





Vorwort/Disclaimer

Vorwort/Disclaimer

- Warum gerade ArchLinux?
 - ArchWiki
 - Lerneffekt
- Theorieteil am Anfang verkürzt
- Links verweisen auf englische Seiten
- Fragen -> *Verständnisfragen* sofort/am Ende der Folie



Grundlagen

Grundlagen

- ArchLinux
- Docker
- Noch Fragen?

Grundlagen

- ArchLinux
- Docker
- Noch Fragen?

Grundlagen

- ArchLinux
- Docker
- Noch Fragen?



ArchLinux

ArchLinux

- 2002 – Neuentwicklung aus Crux nach dem KISS-Prinzip
- KISS = Keep it simple stupid → *Mach es so einfach wie möglich*
- Rolling Release
- vergleichbar mit openSUSE Tumbleweed
- AUR = ArchLinux User Repository,
<https://aur.archlinux.org> – ähnlich wie PPAs in Ubuntu
- Pacman – Paket-Manager
- AUR-Helper – für einfache AUR-PKG Installation
- AUR-Helper: <https://github.com/Jguer/yay>
- PKGBUILD:
<https://wiki.archlinux.org/index.php/PKGBUILD>
- mehr im ArchLinux-Wiki unter
https://wiki.archlinux.org/index.php/Arch_Linux

ArchLinux

- 2002 – Neuentwicklung aus Crux nach dem KISS-Prinzip
- KISS = Keep it simple stupid → *Mach es so einfach wie möglich*
- Rolling Release
- vergleichbar mit openSUSE Tumbleweed
- AUR = ArchLinux User Repository,
<https://aur.archlinux.org> – ähnlich wie PPAs in Ubuntu
- Pacman – Paket-Manager
- AUR-Helper – für einfache AUR-PKG Installation
- AUR-Helper: <https://github.com/Jguer/yay>
- PKGBUILD:
<https://wiki.archlinux.org/index.php/PKGBUILD>
- mehr im ArchLinux-Wiki unter
https://wiki.archlinux.org/index.php/Arch_Linux

ArchLinux

- 2002 – Neuentwicklung aus Crux nach dem KISS-Prinzip
- KISS = Keep it simple stupid → *Mach es so einfach wie möglich*
- Rolling Release
- vergleichbar mit openSUSE Tumbleweed
- AUR = ArchLinux User Repository,
<https://aur.archlinux.org> – ähnlich wie PPAs in Ubuntu
- Pacman – Paket-Manager
- AUR-Helper – für einfache AUR-PKG Installation
- AUR-Helper: <https://github.com/Jguer/yay>
- PKGBUILD:
<https://wiki.archlinux.org/index.php/PKGBUILD>
- mehr im ArchLinux-Wiki unter
https://wiki.archlinux.org/index.php/Arch_Linux

ArchLinux

- 2002 – Neuentwicklung aus Crux nach dem KISS-Prinzip
- KISS = Keep it simple stupid → *Mach es so einfach wie möglich*
- Rolling Release
- vergleichbar mit openSUSE Tumbleweed
- AUR = ArchLinux User Repository,
<https://aur.archlinux.org> – ähnlich wie PPAs in Ubuntu
- Pacman – Paket-Manager
- AUR-Helper – für einfache AUR-PKG Installation
- AUR-Helper: <https://github.com/Jguer/yay>
- PKGBUILD:
<https://wiki.archlinux.org/index.php/PKGBUILD>
- mehr im ArchLinux-Wiki unter
https://wiki.archlinux.org/index.php/Arch_Linux

ArchLinux

- 2002 – Neuentwicklung aus Crux nach dem KISS-Prinzip
- KISS = Keep it simple stupid → *Mach es so einfach wie möglich*
- Rolling Release
- vergleichbar mit openSUSE Tumbleweed
- AUR = ArchLinux User Repository,
<https://aur.archlinux.org> – ähnlich wie PPAs in Ubuntu
- Pacman – Paket-Manager
- AUR-Helper – für einfache AUR-PKG Installation
- AUR-Helper: <https://github.com/Jguer/yay>
- PKGBUILD:
<https://wiki.archlinux.org/index.php/PKGBUILD>
- mehr im ArchLinux-Wiki unter
https://wiki.archlinux.org/index.php/Arch_Linux

ArchLinux

- 2002 – Neuentwicklung aus Crux nach dem KISS-Prinzip
- KISS = Keep it simple stupid → *Mach es so einfach wie möglich*
- Rolling Release
- vergleichbar mit openSUSE Tumbleweed
- AUR = ArchLinux User Repository,
<https://aur.archlinux.org> – ähnlich wie PPAs in Ubuntu
- Pacman – Paket-Manager
- AUR-Helper – für einfache AUR-PKG Installation
- AUR-Helper: <https://github.com/Jguer/yay>
- PKGBUILD:
<https://wiki.archlinux.org/index.php/PKGBUILD>
- mehr im ArchLinux-Wiki unter
https://wiki.archlinux.org/index.php/Arch_Linux

ArchLinux

- 2002 – Neuentwicklung aus Crux nach dem KISS-Prinzip
- KISS = Keep it simple stupid → *Mach es so einfach wie möglich*
- Rolling Release
- vergleichbar mit openSUSE Tumbleweed
- AUR = ArchLinux User Repository,
<https://aur.archlinux.org> – ähnlich wie PPAs in Ubuntu
- Pacman – Paket-Manager
- AUR-Helper – für einfache AUR-PKG Installation
- AUR-Helper: <https://github.com/Jguer/yay>
- PKGBUILD:
<https://wiki.archlinux.org/index.php/PKGBUILD>
- mehr im ArchLinux-Wiki unter
https://wiki.archlinux.org/index.php/Arch_Linux

ArchLinux

- 2002 – Neuentwicklung aus Crux nach dem KISS-Prinzip
- KISS = Keep it simple stupid → *Mach es so einfach wie möglich*
- Rolling Release
- vergleichbar mit openSUSE Tumbleweed
- AUR = ArchLinux User Repository,
<https://aur.archlinux.org> – ähnlich wie PPAs in Ubuntu
- Pacman – Paket-Manager
- AUR-Helper – für einfache AUR-PKG Installation
- AUR-Helper: <https://github.com/Jguer/yay>
- PKGBUILD:
<https://wiki.archlinux.org/index.php/PKGBUILD>
- mehr im ArchLinux-Wiki unter
https://wiki.archlinux.org/index.php/Arch_Linux

ArchLinux

- 2002 – Neuentwicklung aus Crux nach dem KISS-Prinzip
- KISS = Keep it simple stupid → *Mach es so einfach wie möglich*
- Rolling Release
- vergleichbar mit openSUSE Tumbleweed
- AUR = ArchLinux User Repository,
<https://aur.archlinux.org> – ähnlich wie PPAs in Ubuntu
- Pacman – Paket-Manager
- AUR-Helper – für einfache AUR-PKG Installation
- AUR-Helper: <https://github.com/Jguer/yay>
- PKGBUILD:
<https://wiki.archlinux.org/index.php/PKGBUILD>
- mehr im ArchLinux-Wiki unter
https://wiki.archlinux.org/index.php/Arch_Linux

ArchLinux

- 2002 – Neuentwicklung aus Crux nach dem KISS-Prinzip
- KISS = Keep it simple stupid → *Mach es so einfach wie möglich*
- Rolling Release
- vergleichbar mit openSUSE Tumbleweed
- AUR = ArchLinux User Repository,
<https://aur.archlinux.org> – ähnlich wie PPAs in Ubuntu
- Pacman – Paket-Manager
- AUR-Helper – für einfache AUR-PKG Installation
- AUR-Helper: <https://github.com/Jguer/yay>
- PKGBUILD:
<https://wiki.archlinux.org/index.php/PKGBUILD>
- mehr im ArchLinux-Wiki unter
https://wiki.archlinux.org/index.php/Arch_Linux



PKGBUILD

PKGBUILD 1/2

- gute Infos im ArchWiki
<https://wiki.archlinux.org/index.php/PKGBUILD>
- `makepkg` im `{base-devel}`
- obligatorische Variablen:
 - `pkgname` – Packagename
 - `pkgver` – Packageversion
 - `pkgrel` – Packagerelease
 - `arch` – Architektur (i686, x86_64, any)
- empfohlene Variablen:
 - `license` (Warning im Build)
 - `checksums` (md5 oder sha256)
 - `pkgdesc` – Beschreibung

PKGBUILD 1/2

- gute Infos im ArchWiki
<https://wiki.archlinux.org/index.php/PKGBUILD>
- makepkg im {base-devel}
- obligatorische Variablen:
 - pkgname – Packagename
 - pkgver – Packageversion
 - pkgrel – Packagerelease
 - arch – Architektur (i686, x86_64, any)
- empfohlene Variablen:
 - license (Warning im Build)
 - checksums (md5 oder sha256)
 - pkgdesc – Beschreibung

PKGBUILD 1/2

- gute Infos im ArchWiki
<https://wiki.archlinux.org/index.php/PKGBUILD>
- makepkg im {base-devel}
- obligatorische Variablen:
 - pkgname – Packagename
 - pkgver – Packageversion
 - pkgrel – Packagerelease
 - arch – Architektur (i686, x86_64, any)
- empfohlene Variablen:
 - license (Warning im Build)
 - checksums (md5 oder sha256)
 - pkgdesc – Beschreibung

PKGBUILD 1/2

- gute Infos im ArchWiki
<https://wiki.archlinux.org/index.php/PKGBUILD>
- makepkg im {base-devel}
- obligatorische Variablen:
 - pkgname – Packagename
 - pkgver – Packageversion
 - pkgrel – Packagerelease
 - arch – Architektur (i686, x86_64, any)
- empfohlene Variablen:
 - license (Warning im Build)
 - checksums (md5 oder sha256)
 - pkgdesc – Beschreibung

PKGBUILD 1/2

- gute Infos im ArchWiki
<https://wiki.archlinux.org/index.php/PKGBUILD>
- makepkg im {base-devel}
- obligatorische Variablen:
 - pkgname – Packagename
 - pkgver – Packageversion
 - pkgrel – Packagerelease
 - arch – Architektur (i686, x86_64, any)
- empfohlene Variablen:
 - license (Warning im Build)
 - checksums (md5 oder sha256)
 - pkgdesc – Beschreibung

PKGBUILD 1/2

- gute Infos im ArchWiki
<https://wiki.archlinux.org/index.php/PKGBUILD>
- makepkg im {base-devel}
- obligatorische Variablen:
 - pkgname – Packagename
 - pkgver – Packageversion
 - pkgrel – Packagerelease
 - arch – Architektur (i686, x86_64, any)
- empfohlene Variablen:
 - license (Warning im Build)
 - checksums (md5 oder sha256)
 - pkgdesc – Beschreibung

PKGBUILD 1/2

- gute Infos im ArchWiki
<https://wiki.archlinux.org/index.php/PKGBUILD>
- makepkg im {base-devel}
- obligatorische Variablen:
 - pkgname – Packagename
 - pkgver – Packageversion
 - pkgrel – Packagerelease
 - arch – Architektur (i686, x86_64, any)
- empfohlene Variablen:
 - license (Warning im Build)
 - checksums (md5 oder sha256)
 - pkgdesc – Beschreibung

PKGBUILD 1/2

- gute Infos im ArchWiki
<https://wiki.archlinux.org/index.php/PKGBUILD>
- makepkg im {base-devel}
- obligatorische Variablen:
 - pkgname – Packagename
 - pkgver – Packageversion
 - pkgrel – Packagerelease
 - arch – Architektur (i686, x86_64, any)
- empfohlene Variablen:
 - license (Warning im Build)
 - checksums (md5 oder sha256)
 - pkgdesc – Beschreibung

PKGBUILD 1/2

- gute Infos im ArchWiki
<https://wiki.archlinux.org/index.php/PKGBUILD>
- makepkg im {base-devel}
- obligatorische Variablen:
 - pkgname – Packagename
 - pkgver – Packageversion
 - pkgrel – Packagerelease
 - arch – Architektur (i686, x86_64, any)
- empfohlene Variablen:
 - license (Warning im Build)
 - checksums (md5 oder sha256)
 - pkgdesc – Beschreibung

PKGBUILD 1/2

- gute Infos im ArchWiki
<https://wiki.archlinux.org/index.php/PKGBUILD>
- makepkg im {base-devel}
- obligatorische Variablen:
 - pkgname – Packagename
 - pkgver – Packageversion
 - pkgrel – Packagerelease
 - arch – Architektur (i686, x86_64, any)
- empfohlene Variablen:
 - license (Warning im Build)
 - checksums (md5 oder sha256)
 - pkgdesc – Beschreibung

PKGBUILD 1/2

- gute Infos im ArchWiki
<https://wiki.archlinux.org/index.php/PKGBUILD>
- makepkg im {base-devel}
- obligatorische Variablen:
 - pkgname – Packagename
 - pkgver – Packageversion
 - pkgrel – Packagerelease
 - arch – Architektur (i686, x86_64, any)
- empfohlene Variablen:
 - license (Warning im Build)
 - checksums (md5 oder sha256)
 - pkgdesc – Beschreibung

PKGBUILD 2/2

- `Maintainer` – # Maintainer name `<e@mail.adresse>`
- `build()`
- `package()`
- `.SRCINFO` wird aus der PKGBUILD mit `makepkg` erzeugt
 - `$ makepkg -printsrcinfo > .SRCINFO`
 - erster Commit = PKGBUILD & `.SRCINFO`
- guter Vortrag in der Mediathek des CCC:
<https://media.ccc.de/v/gpn18-10-aur-pakete-bauen-fr-anfnger>

PKGBUILD 2/2

- `Maintainer` – # Maintainer name `<e@mail.adresse>`
- `build()`
- `package()`
- `.SRCINFO` wird aus der PKGBUILD mit `makepkg` erzeugt
 - `$ makepkg -printsrcinfo > .SRCINFO`
 - erster Commit = PKGBUILD & `.SRCINFO`
- guter Vortrag in der Mediathek des CCC:
<https://media.ccc.de/v/gpn18-10-aur-pakete-bauen-fr-anfnger>

PKGBUILD 2/2

- `Maintainer` – # Maintainer name `<e@mail.adresse>`
- `build()`
- `package()`
- `.SRCINFO` wird aus der PKGBUILD mit `makepkg` erzeugt
 - `$ makepkg -printsrcinfo > .SRCINFO`
 - erster Commit = PKGBUILD & `.SRCINFO`
- guter Vortrag in der Mediathek des CCC:
<https://media.ccc.de/v/gpn18-10-aur-pakete-bauen-fr-anfnger>

PKGBUILD 2/2

- `Maintainer` – # Maintainer name `<e@mail.adresse>`
- `build()`
- `package()`
- `.SRCINFO` wird aus der PKGBUILD mit `makepkg` erzeugt
 - `$ makepkg -printsrcinfo > .SRCINFO`
 - erster Commit = PKGBUILD & `.SRCINFO`
- guter Vortrag in der Mediathek des CCC:
<https://media.ccc.de/v/gpn18-10-aur-pakete-bauen-fr-anfnger>

PKGBUILD 2/2

- Maintainer – # Maintainer name <e@mail.adresse>
- build()
- package()
- .SRCINFO wird aus der PKGBUILD mit makepkg erzeugt
 - \$ makepkg -printsrcinfo > .SRCINFO
 - erster Commit = PKGBUILD & .SRCINFO
- guter Vortrag in der Mediathek des CCC:
<https://media.ccc.de/v/gpn18-10-aur-pakete-bauen-fr-anfnger>

PKGBUILD 2/2

- `Maintainer` – # Maintainer name `<e@mail.adresse>`
- `build()`
- `package()`
- `.SRCINFO` wird aus der PKGBUILD mit `makepkg` erzeugt
 - `$ makepkg -printsrcinfo > .SRCINFO`
 - erster Commit = PKGBUILD & `.SRCINFO`
- guter Vortrag in der Mediathek des CCC:
<https://media.ccc.de/v/gpn18-10-aur-pakete-bauen-fr-anfnger>

PKGBUILD 2/2

- `Maintainer - # Maintainer name <e@mail.adresse>`
- `build()`
- `package()`
- `.SRCINFO` wird aus der PKGBUILD mit `makepkg` erzeugt
 - `$ makepkg -printsrcinfo > .SRCINFO`
 - erster Commit = PKGBUILD & `.SRCINFO`
- guter Vortrag in der Mediathek des CCC:
<https://media.ccc.de/v/gpn18-10-aur-pakete-bauen-fr-anfnger>



Docker

Docker

- 2013 – erstes Release
- Apache License 2.0
- Architektur: x86_64, ARM, s390x, ppc64le
- Binaries: alle 3 „großen“ OS
- Virtualisierung ohne VM
- chroot + cgroups + NetworkNS in „beautiful & Noobfreundlich“
- DockerHub – *Docker Container Repository*

Docker

- 2013 – erstes Release
- Apache License 2.0
- Architektur: x86_64, ARM, s390x, ppc64le
- Binaries: alle 3 „großen“ OS
- Virtualisierung ohne VM
- chroot + cgroups + NetworkNS in „beautiful & Noobfreundlich“
- DockerHub – *Docker Container Repository*

Docker

- 2013 – erstes Release
- Apache License 2.0
- Architektur: x86_64, ARM, s390x, ppc64le
- Binaries: alle 3 „großen“ OS
- Virtualisierung ohne VM
- chroot + cgroups + NetworkNS in „beautiful & Noobfreundlich“
- DockerHub – *Docker Container Repository*

Docker

- 2013 – erstes Release
- Apache License 2.0
- Architektur: x86_64, ARM, s390x, ppc64le
- Binaries: alle 3 „großen“ OS
- Virtualisierung ohne VM
- chroot + cgroups + NetworkNS in „beautiful & Noobfreundlich“
- DockerHub – *Docker Container Repository*

Docker

- 2013 – erstes Release
- Apache License 2.0
- Architektur: x86_64, ARM, s390x, ppc64le
- Binaries: alle 3 „großen“ OS
- Virtualisierung ohne VM
- chroot + cgroups + NetworkNS in „beautiful & Noobfreundlich“
- DockerHub – *Docker Container Repository*

Docker

- 2013 – erstes Release
- Apache License 2.0
- Architektur: x86_64, ARM, s390x, ppc64le
- Binaries: alle 3 „großen“ OS
- Virtualisierung ohne VM
- chroot + cgroups + NetworkNS in „beautiful & Noobfreundlich“
- DockerHub – *Docker Container Repository*

Docker

- 2013 – erstes Release
- Apache License 2.0
- Architektur: x86_64, ARM, s390x, ppc64le
- Binaries: alle 3 „großen“ OS
- Virtualisierung ohne VM
- chroot + cgroups + NetworkNS in „beautiful & Noobfreundlich“
- DockerHub – *Docker Container Repository*



Dockerfile

Dockerfile

```
# version:  z.y
# git:      https://github.com/username/repo
# description: was mache ich hier bloss...
# docs: docs.url

FROM base/image

ENV VARIABLE="content"

RUN befehle -switch \  
&& befehle

USER username
WORKDIR /home/username

RUN befehle -swich

RUN befehle
```



Aufgaben

Aufgaben

- 1 Outdated Packages
- 2 eigenes PKGBUILD schreiben
- 3 Dockerfile aus Ansible

Vielen Dank für Ihre Aufmerksamkeit!

Bei weiteren Fragen wenden Sie sich bitte an info@b1-systems.de
oder +49 (0)8457 - 931096