



Paketmanagement mit XBPS bei Void Linux

Gut eingepackt

Die Linux-Distribution Void pflegt mit XBPS ein eigenes Paketmanagementsystem.

Es arbeitet rasend schnell und erlaubt den Bau eigener Pakete. Ferdinand Thommes

README

XBPS dient bei Void Linux nicht nur zum Verwalten des Paketbestands, sondern mit der Erweiterung Xbps-src zum Bau von Paketen. Durch Änderungen an den vorgegebenen Rezepten erstellen Sie abgewandelte Packages oder bringen durch eigene Vorlagen neue Pakete in die Distribution ein.

Paketverwaltungen bilden das Herzstück aller Linux-Distributionen. Die Paketverwaltung regelt bei der Installation von Software die Verteilung der Komponenten des Pakets in die Verzeichnisstruktur des Filesystem Hierarchy Standard (FHS) [↗](#), den fast alle Distributionen seit 1993 verwenden.

In den Aufgabenbereich einer Paketverwaltung fallen neben der Installation der eigentlichen Anwendungspakete meist auch das Nachziehen der benötigten Abhängigkeiten sowie das Aktualisieren und Entfernen von Paketen. Zu-

dem lässt sich der Paketbestand nach verschiedenen Kriterien durchsuchen.

Die am häufigsten anzutreffenden Paketverwaltungssysteme unter Linux bedienen die Formate DEB und RPM. Daneben gibt es neben Exoten wie Nix unter anderem Pacman bei Arch Linux, Portage bei Gentoo, Eopkg bei Solus oder XBPS bei Void. Mit Letzterem beschäftigt sich dieser Artikel.

XBPS steht für X Binary Package System und entstand ursprünglich als Ersatz für Pkgsrc [↗](#) bei NetBSD. Im Laufe der Zeit entwickelte sich aus diesem Vor-

haben die Rolling-Release-Distribution Void, über die wir in LU 11/2017 bereits berichteten [☞](#). Man darf XBPS also mit Fug und Recht als Herzstück dieser Distribution bezeichnen.

Die von XBPS verwalteten Pakete tragen die Endung `.xbps`. Aus technischer Sicht handelt es sich wie bei Debian-Paketen um mit Xz gepackte TAR-Archive. Die Basis von XBPS bilden die Repositories. Das System unterstützt neben lokal gespeicherten Repos auch via HTTP(S) oder FTP angebundene Paketquellen von entfernten Servern.

Die System-Repositories finden sich unter `/usr/share/xbps.d/`, von Anwendern hinzugefügte unter `/etc/xbps.d`. Die Inhalte der offiziellen Paketquellen sind mit RSA signiert, die enthaltenen Metadaten, Dateien und Binärpakete mit SHA256 gehashed. Subrepositories für Non-Free-Pakete oder Multi-Arch-Umgebungen lassen sich ebenfalls einbinden.

XBPS und Void

Void verwaltet nicht nur Repositories, deren Pakete die C-Standard-Bibliothek Glibc verwenden, sondern unterstützt in separaten Repositories auch die Alternative Musl [☞](#). XBPS beherrscht neben dem Paketmanagement aber noch mehr: Es stellt gleichzeitig auch das Buildsystem, mit dem die Void-Entwickler alle Pakete der Distribution bauen.

Derzeit liegt der von XBPS verwaltete Bestand in Void für die x86-Plattform bei über 8250 Paketen, die ARM-Varianten bieten jeweils über 5000 Pakete an. Um Zugriff zu bekommen, müssen Sie zunächst die Quellen aktualisieren. Im Zuge dieses Updates bringen Sie per `xbps-install -Su` auch gleich den Paketbestand auf den neuesten Stand [1](#).

Blitzschnell unterwegs

Als Paketmanager legt XBPS Wert auf Einfachheit und Geschwindigkeit. Letzteres zeigt sich umgehend, wenn man XBPS mit dem ebenfalls flotten Debian-Paketsystem vergleicht. XBPS geht etwa bei der Suche [2](#) und der Installation [3](#) von Paketen extrem flott zu Werke.

Zudem beherrscht XBPS ein paar nützliche Funktionen, die den großen Paketmanagern meist fehlen. So unterstützt XBPS beispielsweise Package Reverts: Damit können die Entwickler, falls einmal ein Fehler unbemerkt zu den Anwendern gelangt, ein Paket-Downgrade verteilen und damit das defekte Paket ersetzen. Zudem lässt sich per `xbps-pkgdb Paket` ein Paket auf Fehler überprüfen.

Ein paar nette Tricks

Zudem bietet XBPS die Option, Pakete mitsamt der anderweitig nicht benötigten Abhängigkeiten zu entfernen. Dazu



```
Shell Nr. 1
Datei Aktionen Bearbeiten Ansicht Hilfe
Shell Nr. 1 x
[test@voidlinux ~]$ sudo xbps-install -Su
[*] Updating `https://repo.voidlinux.eu/current/x86_64-repdata' ...
155 packages will be downloaded:
device-mapper-2.02.176_1 dhcpcd-6.11.5_5 diffutils-3.6_3
e2fsprogs-1.43.7_1 e2fsprogs-libs-1.43.7_1 efibootmgr-15_1
ethtool-4.13_1 eudev-3.2.5_1 eudev-libudev-3.2.5_1 expat-2.2.5_1
gawk-4.2.0_1 glib-2.54.2_1 glib-networking-2.54.1_1 glibc-2.26_3
glibc-locales-2.26_3 gnutls-3.5.16_1 gtk+3-3.22.26_1 gvfs-1.34.1_2
gvfs-afc-1.34.1_2 gvfs-mtp-1.34.1_2 gvfs-smb-1.34.1_2
kidletime-5.40.0_1 ksolid-5.40.0_1 kwayland-5.40.0_1
kwindowsystem-5.40.0_1 lame-3.100_1 lcms2-2.9_1 less-529_1
libEGL-17.3.0_1 libGL-17.3.0_1 libX11-1.6.5_2 libXcursor-1.1.15_1
libXfont2-2.0.3_1 libavcodec-3.4.1_1 libavformat-3.4.1_1
libavutil-3.4.1_1 libblkid-2.31_2 libbluray-1.0.2_1 json-c-0.12.1_1
libcryptsetup-2.0.0_1 libcups-2.2.6_1 libdrm-2.4.88_1
libfdisk-2.31_2 libffi-3.2.1_4 libfftw-3.3.7_1 libfm-qt-0.12.0_1
libgdm-17.3.0_1 libgcc-7.2.0_4 libgcrypt-1.8.2_1 libglapi-17.3.0_1
libgomp-7.2.0_4 libharfbuzz-1.7.2_1 libinput-1.9.4_1
libjpeg-turbo-1.5.3_1 libkscreen-5.11.3_1 liblqt-0.12.0_1
libmount-2.31_2 libnfs-2.0.0_1 libogg-1.3.3_1 libpciaccess-0.14_1
```

1 Als erste Amtshandlungen stehen das Update der Quellen sowie das Aktualisieren des gesamten Paketbestands an.

```
Shell Nr. 1
Datei Aktionen Bearbeiten Ansicht Hilfe
Shell Nr. 1 x
[test@voidlinux xbps]$ sudo xbps-install thunderbird
2 packages will be installed:
  libhunspell-1.6.2_1 thunderbird-52.5.0_1

Size required on disk:      114MB
Free space on disk:        9683MB

Do you want to continue? [Y/n] y

[*] Downloading binary packages

[*] Verifying package integrity
libhunspell-1.6.2_1: verifying RSA signature...
thunderbird-52.5.0_1: verifying RSA signature...

[*] Running transaction tasks
libhunspell-1.6.2_1: unpacking ...
thunderbird-52.5.0_1: unpacking ...

[*] Configuring unpacked packages
libhunspell-1.6.2_1: configuring ...
libhunspell-1.6.2_1: installed successfully.
thunderbird-52.5.0_1: configuring ...
Updating GTK+ icon cache for /usr/share/icons/hicolor...
Updating MIME database...
thunderbird-52.5.0_1: installed successfully.

0 downloaded, 2 installed, 0 updated, 2 configured, 0 removed.
[test@voidlinux xbps]$
```

2 Die Suche nach Paketen leiten Sie mit `xbps-query Paket` ein. Wie schon bei der Installation geht XBPS rasend schnell zu Werke.

dient der Befehl `xbps-remove -R Paket`. Lässt man das `-R` weg, löscht XBPS nur das referenzierte Paket.

Das in XBPS integrierte Buildsystem erlaubt zudem Unit-Tests mithilfe von Travis CI [☞](#) auszuführen und so die Integrität und Konsistenz der Repositories zu überprüfen.

Einen Überblick über die gebräuchlichsten Befehle von XBPS und deren Entsprechungen bei Debian vermittelt die Tabelle [Befehlsreferenz](#).

Pakete erstellen

Wie bereits erwähnt, ermöglicht Void Linux Ihnen über `Xbps-src`, Programme selbst aus dem Quellcode nach vordefi-

nieren Rezepten zu bauen. Die Vorlagen lassen sich dabei an die persönlichen Vorstellungen anpassen. Das Wiki bietet dazu eine Schnellanleitung [☞](#), auf Github findet sich eine technisch detaillierte Beschreibung [☞](#).

Quellcode von Github

Um es vereinfacht zu skizzieren, lädt das Tool zunächst von Github das Repository `Void-Packages` [4](#) mit den Paketvorlagen lokal auf den Rechner. Dann erstellt das System eine Bootstrap-Umgebung sowie eine Chroot, in der es schlussendlich die Pakete baut.

Anwender dürfen aber nicht nur Pakete nach den Void-Rezepten selbst bauen, sondern auch neue Vorlagen für Pakete erstellen. Nach eigenen Vorlagen selbst erstellte Pakete lassen sich dann über einen Pull-Request für das offizielle Repository bereitstellen.

Allerdings herrscht bei Void eine um einiges strengere Qualitätskontrolle als etwa für das – vom Prinzip her vergleichbare – AUR bei Arch Linux. Der Grund dafür liegt unter anderem darin, dass akzeptierte Pakete dann offiziell zur Distribution gehören.

Zum Bauen der Pakete benötigen Sie keine Root-Rechte, da der ganze Build-Prozess in einer Chroot abläuft. Auch Cross-Compiling, also das Erstellen für eine andere Architektur als der zum Bauen verwendeten, stellt kein Problem dar.

Fazit

Obwohl es bereits eine Reihe etablierter Systeme zum Verwalten von Software gibt, ist noch Platz für Neues: XBPS stellt eine umfassende Paketverwaltung dar, die nicht nur das Einrichten, Aktualisieren und Entfernen von Paketen beherrscht, sondern auch alle Pakete der Distribution erstellt. Für Anwender bietet XBPS die Möglichkeit, Vorlagen für eigene Pakete zu erstellen und diese zu bauen. Die daraus resultierenden Pakete eignen sich für den Einsatz auf den eigenen Rechnern, lassen sich aber auch den Entwicklern zum Einbinden in die offiziellen Repositories vorschlagen.

3 Für die Installation des E-Mail-Clients Thunderbird aus den Paketquellen benötigte XBPS im Test weniger als 10 Sekunden.

```

Shell Nr. 1
Datei Aktionen Bearbeiten Ansicht Hilfe
Shell Nr. 1 ✕
[test@voidlinux ~]$ sudo xbps-install thunderbird

Name      Action  Version      New version  Download size
libhunspell install -            1.6.2_1      151KB
thunderbird install -            52.5.0_1     42MB

Size to download: 42MB
Size required on disk: 156MB
Free space on disk: 10GB

Do you want to continue? [Y/n] y

[*] Downloading binary packages
libhunspell-1.6.2_1.x86_64.xbps: 151KB [avg rate: 6127MB/s]
libhunspell-1.6.2_1.x86_64.xbps.sig: 512B [avg rate: 8475KB/s]
thunderbird-52.5.0_1.x86_64.xbps: 42MB [avg rate: 6355KB/s]
thunderbird-52.5.0_1.x86_64.xbps.sig: 512B [avg rate: 7576KB/s]

[*] Verifying package integrity
libhunspell-1.6.2_1: verifying RSA signature...
thunderbird-52.5.0_1: verifying RSA signature...

[*] Running transaction tasks
libhunspell-1.6.2_1: unpacking ...
thunderbird-52.5.0_1: unpacking ...

[*] Configuring unpacked packages
libhunspell-1.6.2_1: configuring ...
libhunspell-1.6.2_1: installed successfully.
    
```

Befehlsreferenz		
	XBPS	Debian
Quellen aktualisieren	<code>xbps-install -S</code>	<code>apt update</code>
System aktualisieren	<code>xbps-install -u</code>	<code>apt dist-upgrade</code>
Paket installieren	<code>xbps-install Paket</code>	<code>apt install Paket</code>
Paket entfernen	<code>xbps-remove Paket</code>	<code>apt remove Paket</code>
Paket samt Abhängigkeiten entfernen	<code>xbps-remove -R Paket</code>	-
Paket samt Konfiguration entfernen	-	<code>apt purge Paket</code>
Bestandteile eines Pakets auflisten	<code>xbps-query -f Paket</code>	<code>dpkg -L Paket</code>
Pakete suchen	<code>xbps-query -Rs Text</code>	<code>apt search Text</code>

XBPS arbeitet schnell, der Unterschied zu anderen Paketverwaltungen ist nicht zu übersehen. Die Installation von Paketen dauert oft nur einen Wimpernschlag. XBPS bietet zudem Funktionen, von denen sich andere Paketmanager eine Scheibe abschneiden könnten. Die umfangreiche Dokumentation erklärt alle Funktionen [☞](#). Eine sehr ausführliche Dokumentation von XBPS in deutscher Sprache [☞](#) hilft bei der Erforschung des Paketmanagers weiter, falls die Englischkenntnisse zum Studium der Originaldokumentation nicht ausreichen.

Wer ein blitzschnelles Linux sucht, das selbst auf älteren Rechnern flott unterwegs ist, Run-it anstatt Systemd verwendet und neben XBPS noch weitere inno-

vative Ansätze bietet, der sollte sich Void Linux genau ansehen. Es ist der lebendige Beweis dafür, dass das Ökosystem Open Source immer wieder für eine innovative Überraschung gut ist. (cla) ■

```

Shell Nr. 1
Datei Aktionen Bearbeiten Ansicht Hilfe
Shell Nr. 1 x
[test@voidlinux ~]$ git clone https://github.com/voidlinux/void-packages
Klone nach 'void-packages' ...
remote: Counting objects: 432104, done.
remote: Total 432104 (delta 0), reused 0 (delta 0), pack-reused 432104
Empfange Objekte: 100% (432104/432104), 180.33 MiB | 3.60 MiB/s, Fertig.
Löse Unterschiede auf: 100% (246299/246299), Fertig.
[test@voidlinux ~]$ cd void-packages/
[test@voidlinux void-packages]$ ls -al
insgesamt 420
drwxr-xr-x  7 test test  4096 15. Dez 13:47 .
drwx----- 15 test test  4096 15. Dez 13:45 ..
drwxr-xr-x  7 test test  4096 15. Dez 13:47 .git
-rw-r--r--  1 test test   58 15. Dez 13:47 .gitattributes
drwxr-xr-x  2 test test  4096 15. Dez 13:47 .github
-rw-r--r--  1 test test   65 15. Dez 13:47 .gitignore
-rw-r--r--  1 test test  2027 15. Dez 13:47 .mailmap
-rw-r--r--  1 test test   835 15. Dez 13:47 .travis.yml
-rw-r--r--  1 test test  5158 15. Dez 13:47 CONTRIBUTING.md
-rw-r--r--  1 test test  1295 15. Dez 13:47 COPYING
-rw-r--r--  1 test test  53756 15. Dez 13:47 Manual.md
-rw-r--r--  1 test test  17941 15. Dez 13:47 README.md
drwxr-xr-x 12 test test  4096 15. Dez 13:47 common
drwxr-xr-x  2 test test  4096 15. Dez 13:47 etc
drwxr-xr-x 5198 test test 266240 15. Dez 13:47 srcpkgs
-rwxr-xr-x  1 test test  27743 15. Dez 13:47 xbps-src
[test@voidlinux void-packages]$

```

4 Um selbst Pakete zu erstellen, müssen Sie zunächst ein Github-Repository ins Home-Verzeichnis klonen.



Weitere Infos und interessante Links

www.linux-user.de/qr/40335

Chemnitzer
Linux-Tage
10. und 11. März 2018



www.linux-tage.de

Jeder fängt mal an.