



COMMUNITY-EDITION  
Frei kopieren und beliebig weiter verteilen!

McFly: Bash-History tunen  
mit maschinellem Lernen S. 82

07.2019

# linuxUSER

Alltagssysteme im großen Vergleich, Spezialisten mit innovativen Konzepten

## DISTRIBUTIONEN

**Ubuntu 19.04: Konsequenter auf bewährtem Kurs** S. 10

**Clear Linux: Atomic Updates und Applikations-Bündel** S. 36

**Solus & EasyOS: Exoten mit spannenden Konzepten** S. 30, 42

**Vergleichstest: So schlagen sich OpenSuse 15.1, Fedora 30 und Ubuntu 19.04 im Alltagseinsatz** S. 20

**Arch User Repository komfortabel nutzen** S. 96

Topaktuelle Software perfekt ins System einbinden, Gefahren in Build-Dateien sofort erkennen, mit ausgefeilten AUR-Helfern viel Zeit und Nerven sparen

**Ethernet und WLAN**  
Mit wenigen Zeilen Konfiguration bringt Systemd den PC ans Netz S. 90

**Spiele einfach installieren**  
Dank Lutris reichen oft wenige Klicks und der Game-Spaß kann beginnen S. 48



**Infotainment**  
Datenträger enthält nur Lehr- oder Infoprogramme  
[www.linux-user.de](http://www.linux-user.de)



EUR 8,50 Deutschland    EUR 9,35 Österreich    sfr 17,00 Schweiz    EUR 10,85 Benelux    EUR 11,05 Spanien    EUR 11,05 Italien



# Reality Check

Sehr geehrte Leserinnen und Leser,

oft hilft einem unverhofft ein kurzer Blick über den Tellerrand dabei, eine durch Betriebsblindheit aus der Balance geratene Perspektive wieder zurechtzurücken. Als Anwender, der privat wie beruflich nur Linux einsetzt, finde ich immer wieder das ein oder andere am Betriebssystem im Allgemeinen oder einer Distribution im Besonderen auszusetzen. Systemd nervt, manche Anwendung läuft nicht wie gewünscht, Wayland macht Fernzugriffe unnötig kompliziert, die Software ist zu alt (oder zu neu), und überhaupt war früher alles besser. So scheint es zumindest – bis besagter Blick über den Tellerrand einen wieder auf den Boden der Tatsachen zurückholt.

Ein solcher Aha-Effekt stellte sich bei mir vor einigen Wochen ein, als einer meiner Söhne sich einen neuen Laptop kaufte. Mit dem Gerät unter dem Arm, stolzeschwellter Brust und einer Freundin als Begleitung trudelte er auf der heimischen Terrasse ein, wo ich gerade den Feierabend genoss. Er ließ sich gemütlich nieder, packte seine Neuerwerbung aus, und bootete den Rechner. Nun stand das Einrichten von Windows 10 an, wohlgemerkt: nur das Einrichten, das Microsoft-Betriebssystem war ja bereits vorinstalliert. Ich lehnte mich derweil entspannt im Gartenstuhl zurück – den familiären Windows-Support habe ich bereits vor Jahren offiziell eingestellt – und wartete mit Neugier darauf, was nun geschehen würde.

Über das, was in der nächsten halben Stunde folgte, würde man eigentlich den gnädigen Mantel des Schweigens breiten wollen. Windows 10 ließ das jedoch nicht zu, denn dessen „Assistent“ Cortana plapperte während der gesamten Zeit ungezügelt und wasserfallartig vor sich hin. Mit honigsüßer Frauenstimme und der Beharrlichkeit einer Dampfwalze versuchte die Konfigurationsroutine im Minutentakt, meinem Filius persönliche Daten zu entlocken. Ich habe mitgezählt: Allein acht Mal wollte Cortana partout die Standortdaten wissen, obwohl der Junior deren Herausgabe bereits beim ersten Mal abgelehnt hatte. Nach rund 30 Minuten war Windows 10 dann eingerichtet, mein Sohn schwer und seine (durchaus computeraffine) Begleitfee leicht entnervt, und ich um eine Erfahrung reicher.


Sollte ich an dieser Stelle jemals wieder an einer Distribution herummäkeln, kommen Sie doch bitte digital oder persönlich vorbei und zupfen Sie mich kräftig am Ohr: In 30 Minuten könnte ich mindestens zwei Rechner mit Linux einrichten, das Setup würde mich nur ein paar Mausclicks kosten, und es ließe so still ab, dass es sich sogar in einer Kirche während des Gottesdiensts erledigen ließe. Private Daten blieben dabei komplett außen vor.

Klar: Auch Linux hat seine Macken, und man kann und darf sie auch ansprechen. Über eines sollte man sich dabei aber im-

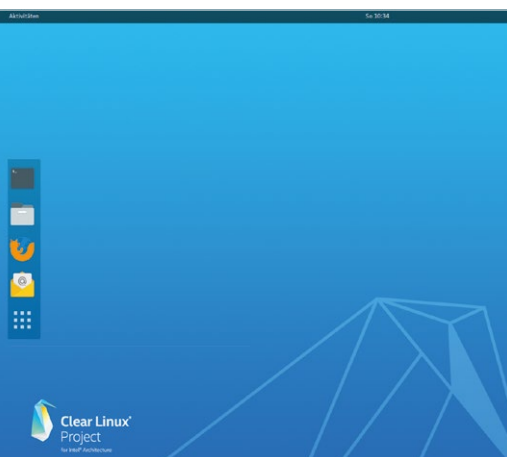
mer im Klaren sein: Letztlich ist das nichts anderes als Jammern auf hohem Niveau.

Wenn Sie die Probe aufs Exempel machen möchten, dann installieren Sie doch einfach mal eine der sechs Distributionen, die Sie auf der Heft-DVD dieser Ausgabe finden. Drei davon zählen zu den „Großen“, die drei anderen könnte man als mehr oder weniger experimentell bezeichnen. Sie werden feststellen, dass ein Windows egal welcher Versionsnummer keiner davon das Wasser reichen kann, weder hinsichtlich der Installation noch im Betrieb. Versprochen.

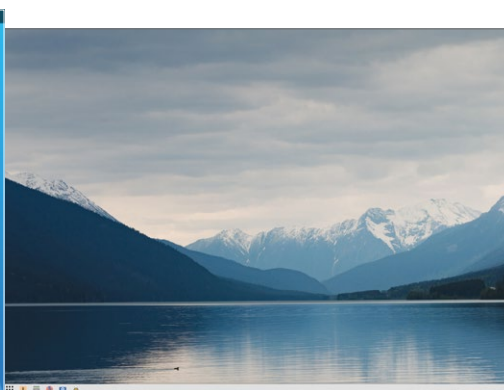
Herzliche Grüße




Jörg Luther  
Chefredakteur



**36** Mit ihrem Fokus auf den Einsatz im Server- und IoT-Bereich ist die Distribution **Clear Linux OS** ein Geheimtipp für den Desktop. Wir zeigen, warum.



**42** Mit **Solus Budgie** zeigt sich wieder einmal, wie sich die Offenheit freier Software positiv auf Innovationen auswirkt. Mit einem geschickten Mix aus bewährten Konzepten und frischen Ideen setzt das System spannende Akzente.



**48** Gaming unter Linux gehört noch immer zu den heiklen Themen. Dabei vereinfachen heute pfiffige Helfer wie **Lutris** den Spielespaß enorm.

## Aktuelles

### News: Software ..... 8

Server-Zertifikate überprüfen mit Cert 0.13.0, DNS-Abfragen verfolgen mit Dnstrace 0.1.1, Quellcode-Syntax hervorheben mit Highlight 3.51, Videos aus Portalen herunterladen mit Piko 0.4.8

## Schwerpunkt

### Ubuntu 19.04 ..... 10

Obwohl nur ein Routine-Release, hat die aktuelle Version der Distribution doch einiges zu bieten: Offenbar bewährten sich die Weichenstellungen der letzten drei Fassungen so gut, dass es bei Bugfixes oder einem aufpolierten Icon-Design blieb.

### Distributionsvergleich ..... 20

Wir vergleichen die aktuellen Versionen von Ubuntu, Fedora und OpenSuse hinsichtlich ihrer Einsteigertauglichkeit, Aktualität sowie ihrer Eignung für die alltägliche Arbeit.

## Schwerpunkt

### EasyOS ..... 30

Die experimentelle Distribution des Puppy-Erfinders Barry Kauler ermöglicht es, auf einfache Weise Container aufzusetzen, Snapshots zu erstellen und das System mit atomaren Updates zu aktualisieren.

### Clear Linux OS ..... 36

Das von Intel entwickelte Clear Linux OS dient dem Chip-Giganten als Vorzeigeprojekt für den Container- und IoT-Bereich. Es eignet sich aber auch als innovatives Betriebssystem für Entwickler und Enthusiasten.

### Solus 4.0 ..... 42

Die Mehrzahl der Distributionen steht auf den Schultern von Debian, Arch oder Fedora. Solus OS tanzt aus der Reihe und macht sein eigenes Ding – und das gar nicht schlecht.

**20** In unserem Vergleichstest geht **OpenSuse Leap 15.1** als eine der drei populärsten Alltagsdistributionsen ins Rennen um die Gunst der Anwender. Mit Fedora und Ubuntu als Konkurrenten kämpft das Feld in verschiedenen Disziplinen um den Spitzenplatz.





**58** Tickt's noch richtig? Dem gehen Sie mit **Timegrapher** auf den Grund und finden per digitaler Zeitwaage heraus, ob ein Uhrwerk noch sauber läuft.

**86** Der anonyme Datenaustausch gerät zunehmend komplizierter. Mit der Software **OnionShare** haben Sie einen einfachen Weg an der Hand, um Transfers über das Netz zu verschleiern.

**90** Übernimmt **Systemd-networkd** die Kontrolle über die Netzwerkschnittstellen, genügen schon wenige Zeilen Konfiguration, um online zu gehen.

### Praxis

**Lutris** ..... 48  
Wer häufiger unter Linux spielt, der musste sich bisher mit zahlreichen verschiedenen Installern und Konfigurationen herumschlagen. Damit macht Lutris jetzt Schluss.

**PDF Arranger** ..... 54  
Mit dem PDF Arranger und Densify stellen wir zwei wichtige Anwendungen im Umgang mit PDF-Dateien vor.

**Timegrapher** ..... 58  
Mechanische Armbanduhren unterliegen natürlichem Verschleiß. Wir zeigen, wie Sie mithilfe von Linux und dem Programm Timegrapher frühzeitig auf Unregelmäßigkeiten aufmerksam werden.

**Baobab** ..... 62  
Nicht nur die Massenspeicher werden immer größer, sondern dank Internet und Multimedia auch die Datenbestände darauf. Baobab hilft dabei, die Datenflut im Griff zu behalten.

### easyLINUX

**Gimp-Tipps** ..... 66  
Mit unseren Tipps und Tricks zu Gimp wird jeder Anfänger zum Bildbearbeitungsprofi.

**OpenSuse-Tipps** ..... 70  
Das Login auf einen OpenSuse-Rechner per SSH aus der Ferne eröffnet viele Möglichkeiten – unter anderem, per VNC auf dem KDE-Desktop zu arbeiten.

**Qmmp** ..... 76  
Audioplayer für Linux gibt es zuhauf – nur kleine, schlanke mit ausreichend Funktionsumfang sind eher Mangelware. Der Winamp-Klon Qmmp bietet einen guten Kompromiss.



**76** Mit Qmmp haben Sie immer den richtigen Beat auf den Ohren. Der schlanke Audio-Player bringt alles mit, was der tägliche Einsatz erfordert.

### Netz&System

**McFly** ..... 82  
McFly bohrt die Bash-History mit KI-Funktionen auf, um Befehle schneller zu finden.

**OnionShare 2.0** ..... 86  
Der einfache Dienst aus dem Tor-Projekt erlaubt es, Daten anonym auszutauschen.

### Know-how

**Systemd-networkd** ..... 90  
Systemd bringt einen eigenen Netzwerk-Daemon mit, der den Rechner per Ethernet oder WLAN ans Netz anbindet.

**Arch User Repository** ..... 96  
Bei der Installation von Paketen über das Arch User Repository helfen ein paar Regeln und die richtigen Helfer.

### Service

**Editorial** ..... 3  
**Impressum** ..... 6  
**Events/Autoren/Inserenten** ..... 7  
**IT-Profimarkt** ..... 102  
**Vorschau** ..... 104  
**Heft-DVD-Inhalt** ..... 105

# COMPUTEC

## MARQUARD MEDIA GROUP

Ein Unternehmen der MARQUARD MEDIA GROUP AG  
Verleger Jürg Marquard

|  |  |  |
|--|--|--|
| Redaktion/Verlag                           | Redaktionsanschrift:<br>Redaktion LinuxUser<br>Putzbrunner Straße 71<br>81739 München<br>Telefon: (0911) 2872-110<br>E-Mail: <a href="mailto:redaktion@linux-user.de">redaktion@linux-user.de</a><br>Web: <a href="http://www.linux-user.de">www.linux-user.de</a> | Verlagsanschrift:<br>Computec Media GmbH<br>Dr. -Mack-Straße 83<br>90762 Fürth<br>Telefon: (0911) 2872-100 |
| Geschäftsführer                            | Rainer Rosenbusch, Christian Müller  |  |
| Chefredakteur,<br>Brand/Editorial Director | Jörg Luther (jlu, v.i.S.d.P.),<br><a href="mailto:jluther@linux-user.de">jluther@linux-user.de</a>   |  |
| Stellv. Chefredakteur                      | Andreas Bohle (agr), <a href="mailto:abohle@linux-user.de">abohle@linux-user.de</a>  |  |
| Redaktion                                  | Christoph Langner (cla), <a href="mailto:clangner@linux-user.de">clangner@linux-user.de</a><br>Thomas Leichtenstern (tle), <a href="mailto:tleichtenstern@linux-user.de">tleichtenstern@linux-user.de</a>  |  |
| Linux-Community                            | Andreas Bohle (agr), <a href="mailto:abohle@linux-community.de">abohle@linux-community.de</a>  |  |
| Datenträger                                | Thomas Leichtenstern (tle), <a href="mailto:cdredaktion@linux-user.de">cdredaktion@linux-user.de</a>   |  |
| Ständige Mitarbeiter                       | Erik Bärwaldt, Karsten Günther, Frank Hofmann, Mandy Neumeyer,<br>Tim Schürmann, Ferdinand Thommes, Uwe Vollbracht, Harald Zisler  |  |
| Titel & Layout                             | Elgin Grabe; Titelmotiv: alexlmx, 123RF<br>Bildnachweis: 123RF, Freemages und andere   |  |
| Sprachlektorat                             | Astrid Hillmer-Bruer   |  |
| Produktion                                 | Martin Clossmann (Ltg.), <a href="mailto:martin.clossmann@computec.de">martin.clossmann@computec.de</a>  |  |
| Vertrieb, Abonnement                       | Werner Spachmüller (Ltg.), <a href="mailto:werner.spachmueller@computec.de">werner.spachmueller@computec.de</a>  |  |
| Anzeigen                                   | Verantwortlich für den Anzeigenteil: Judith Gratias-Klamt<br>Es gilt die Anzeigenpreisliste vom 01.01.2019.  |  |
| Mediaberatung D/A/CH                       | Judith Gratias-Klamt, <a href="mailto:judith.gratias-klamt@computec.de">judith.gratias-klamt@computec.de</a><br>Tel.: (0911) 2872-252, Fax: (0911) 2872-241  |  |
| Mediaberatung UK/USA                       | Brian Osborn, <a href="mailto:bosborn@linuxnewmedia.com">bosborn@linuxnewmedia.com</a>   |  |
| Abo  | Die Abwicklung (Rechnungsstellung, Zahlungsabwicklung und Versand) erfolgt über unser Partnerunternehmen:<br>DPV Deutscher Pressevertrieb GmbH<br>Leserservice Computec<br>20080 Hamburg<br>Deutschland  |  |
| Einzelhefte und<br>Abo- Bestellung         | <a href="http://shop.computec.de">http://shop.computec.de</a>  |  |
| Leserservice<br>Deutschland                | Ihre Ansprechpartner für Reklamationen und Ersatzbestellungen<br>E-Mail: <a href="mailto:computec@dpv.de">computec@dpv.de</a><br>Tel.: (0911) 99 39 90 98<br>Fax: (01805) 861 80 02*<br>(*0,14 €/min aus dem Festnetz, max. 0,42 €/min aus dem Mobilnetz)          |  |
| Österreich, Schweiz<br>und weitere Länder  | E-Mail: <a href="mailto:computec@dpv.de">computec@dpv.de</a><br>Tel.: +49 911 99399098<br>Fax: +49 1805 8618002  |  |
| Supportzeiten                              | Montag 07:00 – 20:00 Uhr, Dienstag – Freitag: 07:30 – 20:00 Uhr,<br>Samstag 09:00 – 14:00 Uhr  |  |
| Pressevertrieb                             | DPV Deutscher Pressevertrieb GmbH<br>Am Sandtorkai 74, 20457 Hamburg<br><a href="http://www.dpv.de">http://www.dpv.de</a>  |  |
| Druck                                      | LSC Communications Europe, ul. Obr. Modlina 11,<br>30-733 Kraków, Polen  |  |
| ISSN                                       | 1615-4444  |  |



MARQUARD MEDIA  
GROUP

Deutschland:

4PLAYERS, AREAMOBILE, BUFFED, GAMESWORLD, GAMESZONE, GOLEM, LINUX-COMMUNITY,  
LINUX-MAGAZIN, LINUXUSER, MAKING GAMES, N-ZONE, GAMES AKTUELL, PC GAMES, PC GAMES HARDWARE,  
PC GAMES MMORE, PLAY 4, RASPBERRY PI GEEK, SFT, VIDEOGAMESZONE, WIDESCREEN

Marquard Media Polska:

CKM, COSMOPOLITAN, ESQUIRE, HARPER'S BAZAAR, JOY, KOZACZEK, PAPILOT, PLAYBOY, ZEBERKA

Marquard Media Hungary:

APA, ÉVA, GYEREKLÉLEK, INSTYLE, JOY, MEN'S HEALTH, PLAYBOY, RUNNER'S WORLD, SHAPE

### ABONNEMENT

| Mini-Abo (3 Ausgaben)                                    | Deutschland | Österreich | Ausland  |
|--|-------------|------------|----------|
| No-Media-Ausgabe <sup>1</sup>                            | 11,90 €     | 11,90 €    | 11,90 €  |
| DVD-Ausgabe  | 16,90 €     | 16,90 €    | 16,90 €  |
| Jahres-Abo (12 Ausgaben)                                 | Deutschland | Österreich | Ausland  |
| No-Media-Ausgabe <sup>1</sup>                            | 60,60 €     | 68,30 €    | 81,00 €  |
| DVD-Ausgabe  | 86,70 €     | 95,00 €    | 99,30 €  |
| Jahres-DVD zum Abo <sup>2</sup>                          | 6,70 €      | 6,70 €     | 6,70 €   |
| Preise Digital   | Deutschland | Österreich | Ausland  |
| Heft-PDF Einzelausgaben Digital                          | 5,99 €      | 5,99 €     | 5,99 €   |
| Digital-Abo (12 Ausgaben)                                | 48,60 €     | 48,60 €    | 48,60 €  |
| Kombi Digital + Print<br>(No-Media-Ausgabe, 12 Ausgaben) | 72,60 €     | 80,30 €    | 93,00 €  |
| Kombi Digital + Print<br>(DVD-Ausgabe, 12 Ausgaben)      | 98,70 €     | 107,00 €   | 111,30 € |

- (1) Die **No-Media-Ausgabe** erhalten Sie ausschließlich in unserem Webshop unter <http://shop.computec.de>, die Auslieferung erfolgt versandkostenfrei.
- (2) Nur erhältlich in Verbindung mit einem Jahresabonnement der Printausgabe von LinuxUser.

Internet <http://www.linux-user.de>  
News und Archiv <http://www.linux-community.de>  
Facebook <http://www.facebook.com/linuxuser.de>

Schüler- und Studentenermäßigung: 20 Prozent gegen Vorlage eines Schülerausweises oder einer aktuellen Immatrikulationsbescheinigung. Der aktuelle Nachweis ist bei Verlängerung neu zu erbringen. Andere Abo-Formen, Ermäßigungen im Ausland etc. auf Anfrage. Adressänderungen bitte umgehend beim Kundenservice mitteilen, da Nachsendeaufträge bei der Post nicht für Zeitschriften gelten.

### Rechtliche Informationen

COMPUTEC MEDIA ist nicht verantwortlich für die inhaltliche Richtigkeit der Anzeigen und übernimmt keinerlei Verantwortung für in Anzeigen dargestellte Produkte und Dienstleistungen. Die Veröffentlichung von Anzeigen setzt nicht die Billigung der angebotenen Produkte und Service-Leistungen durch COMPUTEC MEDIA voraus. Haben Sie Beschwerden zu einem unserer Anzeigenkunden, seinen Produkten oder Dienstleistungen, dann bitten wir Sie, uns das schriftlich mitzuteilen. Schreiben Sie unter Angabe des Magazins, in dem die Anzeige erschienen ist, inklusive der Ausgabe und der Seitennummer an:  
CMS Media Services, Annett Heinze, Verlagsanschrift (siehe oben links).

Linux ist ein eingetragenes Warenzeichen von Linus Torvalds und wird von uns mit seiner freundlichen Genehmigung genutzt. »Unix« verwenden wir als Sammelbegriff für die Gruppe der Unix-ähnlichen Betriebssysteme (wie beispielsweise HP/UX, FreeBSD, Solaris, u.a.), nicht als Bezeichnung für das Trademark »UNIX« der Open Group. Der Linux-Pinguin wurde von Larry Ewing mit dem Pixelgrafikprogramm »The GIMP« erstellt. Eine Haftung für die Richtigkeit von Veröffentlichungen kann – trotz sorgfältiger Prüfung durch die Redaktion – vom Verlag nicht übernommen werden.

Mit der Einreichung von Manuskripten oder Leserbriefen gibt der Verfasser seine Einwilligung zur Veröffentlichung in einer Publikation der COMPUTEC MEDIA. Für unverlangt eingesandte Manuskripte wird keine Haftung übernommen. Autoreninformationen finden Sie unter <http://www.linux-user.de/Autorenhinweise>. Die Redaktion behält sich vor, Einsendungen zu kürzen und zu überarbeiten. Das exklusive Urheber- und Verwertungsrecht für angenommene Manuskripte liegt beim Verlag. Es darf kein Teil des Inhalts ohne schriftliche Genehmigung des Verlags in irgendeiner Form vervielfältigt oder verbreitet werden.

### LinuxUser Community Edition

LinuxUser gibt es auch als Community Edition: Das ist eine rund 32-seitige PDF-Datei mit Artikeln aus der aktuellen Ausgabe, die kurz vor Veröffentlichung des gedruckten Heftes erscheint.

Die kostenlose Community-Edition steht unter einer Creative-Commons-Lizenz, die es erlaubt, „das Werk zu vervielfältigen, zu verbreiten und öffentlich zugänglich machen“. Sie dürfen die LinuxUser Community-Edition also beliebig kopieren, gedruckt oder als Datei an Freunde und Bekannte weitergeben, auf Ihre Website stellen – oder was immer ihnen sonst dazu einfällt. Lediglich bearbeiten, verändern oder kommerziell nutzen dürfen Sie sie nicht. Darum bitten wir Sie im Sinn des „fair use“. Weitere Informationen finden Sie unter: <http://linux-user.de/CE>

### Probleme mit den Datenträgern

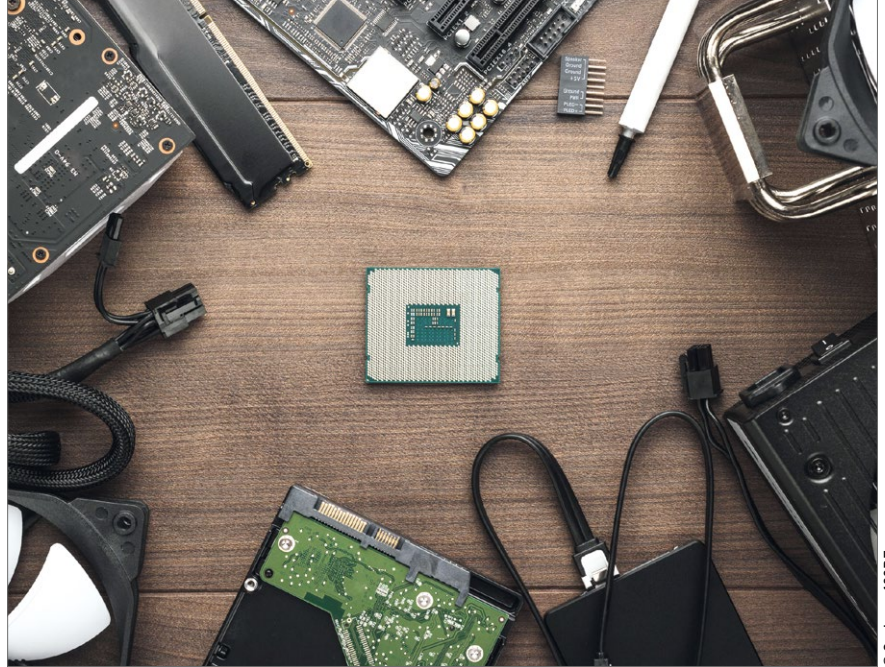
Falls es bei der Nutzung der Heft-DVDs zu Problemen kommt, die auf einen defekten Datenträger schließen lassen, dann schicken Sie bitte eine E-Mail mit einer genauen Fehlerbeschreibung an die Adresse [computec@dpv.de](mailto:computec@dpv.de). Wir senden Ihnen dann umgehend kostenfrei einen Ersatzdatenträger zu.

# Vorschau auf 08/2019

Die nächste Ausgabe erscheint am 18.07.2019

## Aktuelle Hardware

Dass Linux nicht gut mit Hardware zusammenspielt, gehört zu den längst widerlegten Mythen. Heute läuft das freie Betriebssystem auf topaktuellen Laptops, unterstützt direkt ab Werk mehr Peripherie als kaum ein anderes System und spielt – entsprechendes Know-how bei der Installation vorausgesetzt – selbst auf Geräten seine Stärken aus, die eigentlich für Windows gedacht sind. Das demonstrieren wir in der kommenden Ausgabe etwa an einem Surface Pro 6.



© Garfoon, 123RF

## Unsichtbare Tinte

Moderne Drucker markieren jeden Ausdruck mit mikroskopisch kleinen Punkten, um seinen Weg jederzeit bis zum Ersteller zurückverfolgen zu können. Wir zeigen, wie das funktioniert, wie weit die Möglichkeiten reichen, und wie Sie mit dem Deda-Toolkit die Spur verwischen.

## Ablenkungsfreies Schreiben

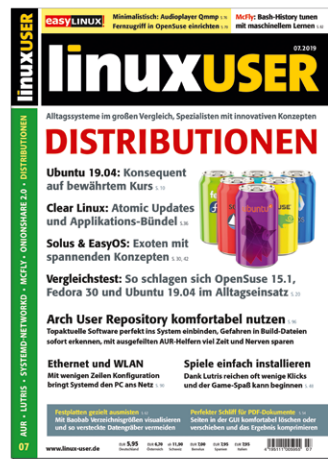
Doch ein anderer Font, oder ein größerer? Moderne Textverarbeitungen verleiten zur Prokrastination. Mit dem Focuswriter setzen Sie an der Wurzel an: Das Programm bietet kaum Möglichkeiten, sich beim Schreiben abzulenken. Wir prüfen, wie gut das in der Praxis klappt.

Die Redaktion behält sich vor, Themen zu ändern oder zu streichen.



## Heft als DVD-Edition

- 108 Seiten Tests und Workshops zu Soft- und Hardware
- 2 DVDs mit Top-Distributionen sowie der Software zu den Artikeln. Mit bis zu 18 GByte Software das Komplettpaket, das Unmengen an Downloads spart



## Heft als No-Media-Edition

- Preisgünstige Heftvariante ohne Datenträger für Leser mit Breitband-Internet-Anschluss
- Artikelumfang identisch mit der DVD-Edition: 108 Seiten Tests und Workshops zu aktueller Soft- und Hardware



## Community-Edition-PDF

- Über 30 Seiten ausgewählte Artikel und Inhaltsverzeichnis als PDF-Datei
- Unter CC-Lizenz: Frei kopieren und beliebig weiter verteilen
- Jeden Monat kostenlos per E-Mail oder zum Download



DVD-Edition (8,50 Euro) oder No-Media-Edition (5,95 Euro)  
Einfach und bequem versandkostenfrei bestellen unter:

<http://www.linux-user.de/bestellen>



Jederzeit gratis heruntergeladen unter:

<http://www.linux-user.de/CE>

## Kontrollleur

Mithilfe von **Cert 0.13.0** überprüfen Sie die Validität von TLS-Zertifikaten.

```
Terminal - root@LULab1804: /home/vollbracht/extract/angetestet/072019/arrons - + x
vollbracht@vmhost:~/extract/LU072019$ cert/cert www.linux-user.de
DomainName: www.linux-user.de
IP: 195.122.146.141
Issuer: Let's Encrypt Authority X3
NotBefore: 2019-04-02 06:15:11 +0200 CEST
NotAfter: 2019-07-01 06:15:11 +0200 CEST
CommonName: www.linux-user.de
SANS: [www.linux-user.de]
Error:

vollbracht@vmhost:~/extract/LU072019$ cert/cert -f md www.linux-user.de
DomainName | IP | Issuer | NotBefore | NotAfter | CN | SANS | Error
--- | --- | --- | --- | --- | --- | --- | ---
www.linux-user.de | 195.122.146.141 | Let's Encrypt Authority X3 | 2019-04-02 06:15:11 +0200 CEST | 2019-07-01 06:15:11 +0200 CEST | www.linux-user.de | www.linux-user.de<br/> |

vollbracht@vmhost:~/extract/LU072019$ cert/cert -f json www.linux-user.de
[{"domainName": "www.linux-user.de", "ip": "195.122.146.141", "issuer": "Let's Encrypt Authority X3", "commonName": "www.linux-user.de", "sans": ["www.linux-user.de"], "notBefore": "2019-04-02 06:15:11 +0200 CEST", "notAfter": "2019-07-01 06:15:11 +0200 CEST", "error": ""}]vollbracht@vmhost:~/extract/LU072019$
```

Nicht zuletzt aufgrund der breiten Akzeptanz von Let's Encrypt liefern viele Webseiten die Daten nur noch HTTPS-verschlüsselt aus, die Verbreitung selbst signierter Zertifikate nimmt dagegen stetig ab. Mit dem in Go geschriebenen Tool Cert prüfen Sie, ob eine Webseite ein Zertifikat und damit verschlüsselte Kommunikation anbietet. Sie beziehen Cert als Binärpaket von der Github-Seite des Projekts. Für eine einfache Abfrage übergeben Sie ihm beim Aufruf die zu prüfende Domain als FQDN, die Angabe einer IP-Adresse führt zu einer Fehlermeldung. Läuft der Webserver auf der zu prüfenden Seite nicht auf dem Standard-Port 443, müssen Sie den verwendeten Port beim Aufruf mit einem Doppelpunkt an den FQDN anfügen.

Cert ermittelt die zugehörige IP-Adresse und zeigt sämtliche

Zertifikatsdaten an, darunter auch Gültigkeitszeitraum, Aussteller und Common Name. Unter dem Eintrag SANS können Sie sehen, für welche weiteren FQDNs (meist Subdomains) das Zertifikat ebenfalls gilt. Um mehrere Webseiten zu prüfen, übergeben Sie diese beim Aufruf als Liste. Cert prüft standardmäßig nicht nur den Hostnamen, sondern auch die Zertifizierungskette. Die fehlt selbst signierten Zertifikaten, sodass das Tool hier mit einer Fehlermeldung die Arbeit einstellt. Mit dem Parameter `-k` weisen Sie Cert an, diese Prüfung zu überspringen, sodass es auch für solche Zertifikate den Aussteller und die Gültigkeitsdauer liefert.

Die Ausgabe der Informationen erfolgt zeilenweise; für eine tabellarische Ausgabe geben Sie mit dem Parameter `-f md` Markdown als Ausgabeformat vor. Zur automatisierten Weiterverarbeitung stellt Cert die Daten auf Wunsch alternativ auch im JSON-Format bereit.

Lizenz: MIT



Quelle: <https://github.com/genkiroid/cert>

## Spurensucher

Fehlern bei der DNS-Abfrage kommen Sie mit **Dnstrace 0.1.1** schnell auf die Spur.

```
Terminal - root@LULab1804: /home/vollbracht/extract/angetestet/072019/arrons - + x
vollbracht@vmhost:~/extract/LU072019/dnstrace$ ./dnstrace www.linux-user.de
Query name: "www.linux-user.de", type: A, nameserver IP: 192.203.230.10, nameserver label: e.root-servers.net.
Query options: recurse: false, EDNS: true, client-subnet: "", fallback: true, TCP: false, ignore sub-queries: false

Querying about "www.linux-user.de."/A @ 192.203.230.10(e.root-servers.net., "." authority)
Answer RTT: 23.796166ms from 192.203.230.10
+ Matched delegated NS and glue in additional section: 194.0.0.53(a.nic.de.)

Querying about "www.linux-user.de."/A @ 194.0.0.53(a.nic.de., "de." authority)
Answer RTT: 32.719464ms from 194.0.0.53
- No NS/glue match, we need extra lookups for ns3.routing.net.

Querying about "ns3.routing.net."/A @ 199.7.91.13(d.root-servers.net., "." authority)
Answer RTT: 27.223097ms from 199.7.91.13
+ Matched delegated NS and glue in additional section: 192.5.6.30(a.gtld-servers.net.)

Querying about "ns3.routing.net."/A @ 192.5.6.30(a.gtld-servers.net., "net." authority)
Answer RTT: 47.752117ms from 192.5.6.30
```

Ließen sich in der guten alten Zeit noch alle Systeme in der `hosts`-Datei des Rechners hinterlegen, kommt man heutzutage ohne DNS-Server definitiv nicht mehr aus. Stockt die Namensauflösung einmal, dann kommen Sie der Ursache mit Dnstrace schnell auf die Spur, das Sie als einsatzbereites Binärpaket von der Github-Seite des Tools laden.

Um den DNS-Eintrag zu einem vollqualifizierten Domain-Namen zu ermitteln, geben Sie den FQDN beim Aufruf als

Parameter an. Dnstrace klopft dann als Erstes bei einem der Root-DNS-Server im Internet an, der die Anfrage zu einem DNS-Server des jeweiligen NIC delegiert. Von dort aus geht sie dann an den zuständigen DNS-Server weiter, der schließlich den zugehörigen DNS-Eintrag ausgibt.

Dabei unterstützt Dnstrace Anfragen im Extended-DNS-Format EDNS, was größere Datenpakete ermöglicht. Die Kommunikation erfolgt normalerweise via UDP; über den Schalter `-tcp` stellen Sie die Anfrage stattdessen via TCP. Wollen Sie der Übersichtlichkeit halber nicht alle Abfrageschritte nachvollziehen, unterdrücken Sie mit `-ignore sub` die Unterabfragen.

Sofern Sie nichts anderes angeben, ermittelt Dnstrace immer den A-Eintrag zur jeweiligen Adresse. Suchen Sie dagegen beispielsweise den MX-Eintrag des Mail-servers zu einer Domain, dann geben Sie Dnstrace beim Aufruf den Query-Befehl MX mit. Eine Übersicht aller verfügbaren Parameter erhalten Sie mit dem Parameter `-h` über die Online-Hilfe. Eine kurze Beschreibung der Einsatzmöglichkeiten samt Anwendungsbeispielen liefert darüber hinaus die Github-Seite des Tools.

Lizenz: MIT



Quelle: <https://github.com/dkorunic/dnstrace>

Highlight geht Ihnen beim Aufbereiten von Quellcode in zahlreiche Ausgabeformate wie RTF, LaTeX, HTML oder XHTML zur Hand. Um den aufbereiteten Code nicht als Satzformat, sondern auf der Konsole auszugeben, verwenden Sie als Ausgabeformat `ansi` oder `xterm256`. Die aktuelle Version kommt mit über 200 Programmiersprachen und deren Dialekten zurecht, identifiziert die jeweiligen Schlüsselwörter der verwendeten Sprache und hebt sie farblich hervor. Das Tool erkennt sogar, wenn der übergebene Quellcode eine Mischung verschiedener Programmiersprachen enthält, und wählt jeweils das passende Highlighting. Bei Sprachen wie C oder Java formatiert Highlight auf Wunsch den Quellcode neu und rückt dabei die Zeilen passend ein.

Lizenz: GPLv3



Quelle: <http://www.andre-simon.de/doku/highlight/en/highlight.php>

Strukturen für neue Sprachen lassen sich ebenso leicht hinzufügen wie alternative Farbgebungen. In den Farbmustern von Highlight geben Sie Schlüsselwörter als feste Zeichenkette oder regulären Ausdruck an. Auf der Projektseite finden Sie eine Anleitung und diverse Beispiele für den Aufbau eigener Sprachdefinitionen.

Die meisten Distributionen führen ältere Highlight-Versionen im Repository, sodass Sie das Tool nur dann aus den Quellen kompilieren müssen, wenn sie Unterstützung für neuere Programmiersprachen benötigen. Die Konfiguration von Highlight erfolgt komplett über Kommandozeilenparameter, deren Bedeutung die Manpage näher erläutert.

```
Terminal - root@LULab1804: /home/vollbracht/extract/angestestet/072019/arrona - + x
vollbracht@vmhost:~/extract/LU072019$ highlight -h
USAGE: highlight [OPTIONS]... [FILES]...

General options:
-B, --batch-recursive=<wc>    convert all matching files, searches subdirs
                              (Example: -B '*.cpp')
-D, --data-dir=<directory>   set path to data directory
--config-file=<file>         set path to a lang or theme file
-d, --outdir=<directory>    name of output directory
-h, --help[=topic]          print this help or a topic description
                              <topic> = [syntax, theme, plugin, config, test]
-i, --input=<file>          name of single input file
-o, --output=<file>         name of single output file
-P, --progress              print progress bar in batch mode
-q, --quiet                 suppress progress info in batch mode
-S, --syntax=<type>         specify type of source code
--syntax-by-name=<name>    specify type of source code by given name
                              will not read a file of this name, useful for stdin
-v, --verbose               print debug info
--force                     generate output if input syntax is unknown
--list-scripts=<type>      list installed scripts
```

## Textmarker

Mit **Highlight 3.51** bereiten Sie Ihren Quellcode übersichtlich auf.

Der einfache Media-Downloader Piko lädt Videos und Bilder von Portalen wie Twitter, Imgur, Facebook oder 4Chan herunter. Sie beziehen das Tool als 64-Bit-Binärpaket direkt von Github. Um etwa einen Clip von Youtube herunterzuladen, geben Sie Piko dessen URL in einfache Anführungszeichen gesetzt beim Aufruf mit. Die Übertragung startet sofort, die Datei landet im aktuellen Verzeichnis. Ein einfacher Fortschrittsbalken zeigt dabei Datendurchsatz und geschätzte Übertragungsdauer an. Sofern Sie nicht mit `--format` etwas anderes bestimmen, wählt Piko als Dateinamen den Titel und als Dateierweiterung das Medienformat.

Lizenz: LGPLv3



Quelle: <https://github.com/mlvzk/piko>

Mit `--discover` aufgerufen, ermittelt Piko nur die Metadaten zur Datei und zeigt mögliche Optionen sowie die aktuelle Standardeinstellung an. Über den Parameter `--option` passen Sie bei Bedarf die Einstellungen von Piko an, verändern die Qualität des Downloads oder übertragen nur die Tonspur eines Videos. Die Dokumentation des Tools fällt allerdings spärlich aus: Neben der Github-Seite mit nur wenigen Beispielen steht lediglich über `-h` eine recht kurze Online-Hilfe zur Verfügung; eine Manpage sucht man vergebens. (jlu) ■

## Saugroboter

Videos und Bilder laden Sie mit **Piko 0.4.8** schnell und einfach aus Portalen herunter.

```
Terminal - root@LULab1804: /home/vollbracht/extract/angestestet/072019/arrona - + x
vollbracht@vmhost:~/extract/LU072019$ ./piko -h
piko alpha

USAGE:
piko [urls...]
piko 'https://www.youtube.com/watch?v=dQw4w9WgXcQ'
piko 'https://www.youtube.com/watch?v=dQw4w9WgXcQ' --stdout | mpv -

OPTIONS:
--help, -h          Prints this page
--format, -f <value> File path format, ex: --format %[id].%[ext]. "id" and "ext" are meta tags(see --discover).
                    Use %[default] to fill with default format, ex: download/%[default]
--option, -o <value> Download options, ex: --option quality=best (accepts multiple)
--discover, -d      Discovery mode, doesn't download anything, only outputs information
--stdout           Output download media to stdout

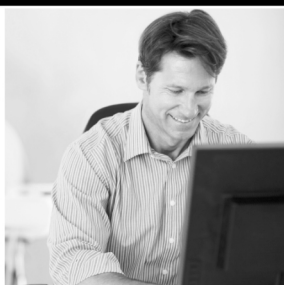
Author: mlvzk
vollbracht@vmhost:~/extract/LU072019$
vollbracht@vmhost:~/extract/LU072019$
vollbracht@vmhost:~/extract/LU072019$
```

# Werden Sie geprüfter Linux-Administrator LPI



Aus- und Weiterbildung zum Linux-Administrator. Ein Beruf mit sehr guten Zukunftsaussichten. Kostengünstiges und praxiserfahrenes Studium ohne Vorkenntnisse zur Vorbereitung auf die LPI-Prüfungen. Beginn jederzeit.

FERNSCHULE WEBER - Techn. Lehrinstitut seit 1959  
Neerstedter Str. 8 - 26197 Großenkneten - Abt. X23  
Tel. 0 44 87 / 2 63 - Fax 0 44 87 / 2 64



Weitere Studiengänge:

- ▶ Computer-Techniker
- ▶ Netzwerk-Technik
- ▶ Fachkraft Online-Marketing
- ▶ IT-Security SSCP/CISSP

**Teststudium  
ohne Risiko!**

GRATIS-Infomappe gleich anfordern!

[www.fernschule-weber.de](http://www.fernschule-weber.de)



Mit EasyOS experimentelle Funktionen erkunden

# Klein, aber fein

© Lassdesigner, 123RF

Mit EasyOS entwarf der Puppy-Erfinder Barry Kauler eine experimentelle Distribution, die Container innerhalb des Systems ebenso ermöglicht wie Snapshots und atomare Updates mit Rollback und das Sie in vielen Fällen über eine GUI administrieren. Ferdinand Thommes

## README

Das von Barry Kauler entwickelte Puppy Linux glänzt in erster Linie durch minimalen Ressourcenbedarf, gepaart mit einfachem Handling. In eine ähnliche Kerbe schlägt die Weiterentwicklung EasyOS, die neben mehr Applikationen einige Zusatzfunktionen bietet, wie etwa das Betreiben von Programmen in Containern oder das simple Update durch den bloßen Austausch dreier Dateien.

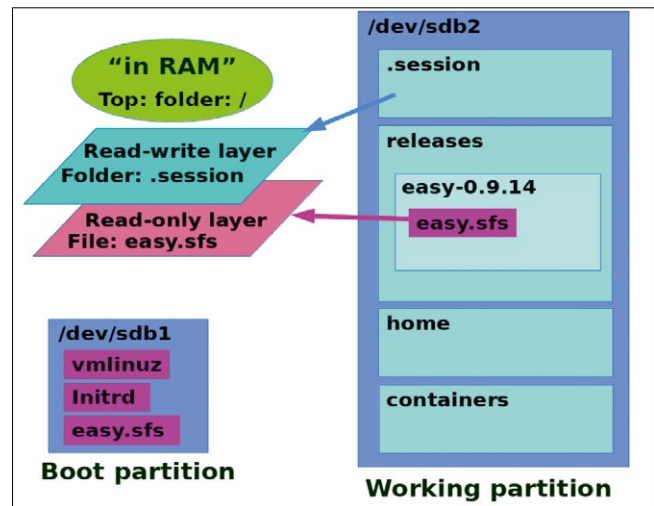
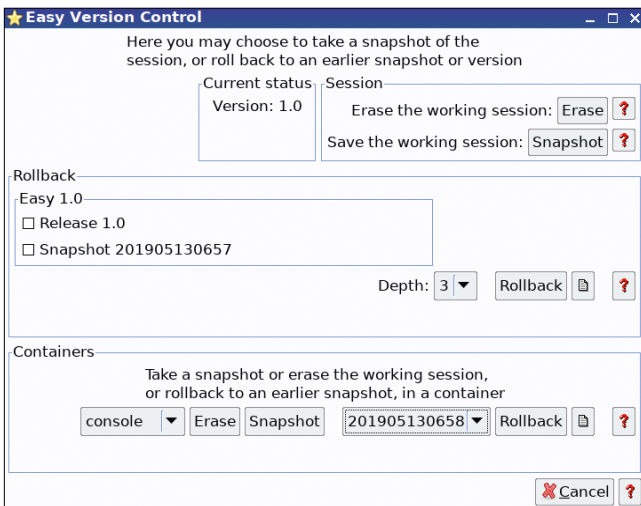
**Was macht der Entwickler** einer Distribution, wenn er sich langweilt? Barry Kauler [🔗](#), australischer Erfinder der leichtgewichtigen Distribution Puppy Linux [🔗](#), beantwortete diese Frage 2014: Er legte das 2003 erstmals veröffentlichte Puppy Linux in die Hände der Community und widmet sich seither der Zusammenstellung experimenteller Distributionen wie Quirky [🔗](#) oder EasyOS [🔗](#). Quirky wurde Ende 2018 offiziell eingestellt, EasyOS ist Kaulers aktuelles Projekt.

Er betont dabei, dass er mit EasyOS keine speziellen Ambitionen verfolgt, was wohl heißt, dass er es weiterentwickelt, solange es ihm Spaß macht. Er bezeichnet das System als „neues Paradigma“ für eine Distribution, eine Mischung der besten Funktionen aus Puppy Linux und Quirky, und ein „fundamentales Um-

denken in Sachen Sicherheit, Wartbarkeit und Benutzerfreundlichkeit“.

Er baute EasyOS von Grund auf neu. Als Build-System kam Woof aus der Puppy-Entwicklung zum Einsatz. Das fütterte er direkt mit Binärpaketen, die er wiederum mit einem eigenen Werkzeug, das seinerseits auf dem Build-Framework OpenEmbedded [🔗](#) beruht, aus dem jeweiligen Quellcode baut.

So vermeidet Kauler den Einsatz des von ihm nicht geschätzten Systemd und setzt stattdessen auf das minimale Init-Script von Busybox. Alle Pakete optimierte er für EasyOS, womit es dem leichtgewichtigen und wieselflinken Puppy nach-eifert. Eines der Paradigmen von EasyOS lautet, dem Anwender die Kommandozeile zu ersparen und für alle Aufgaben eine einfache GUI anzubieten.



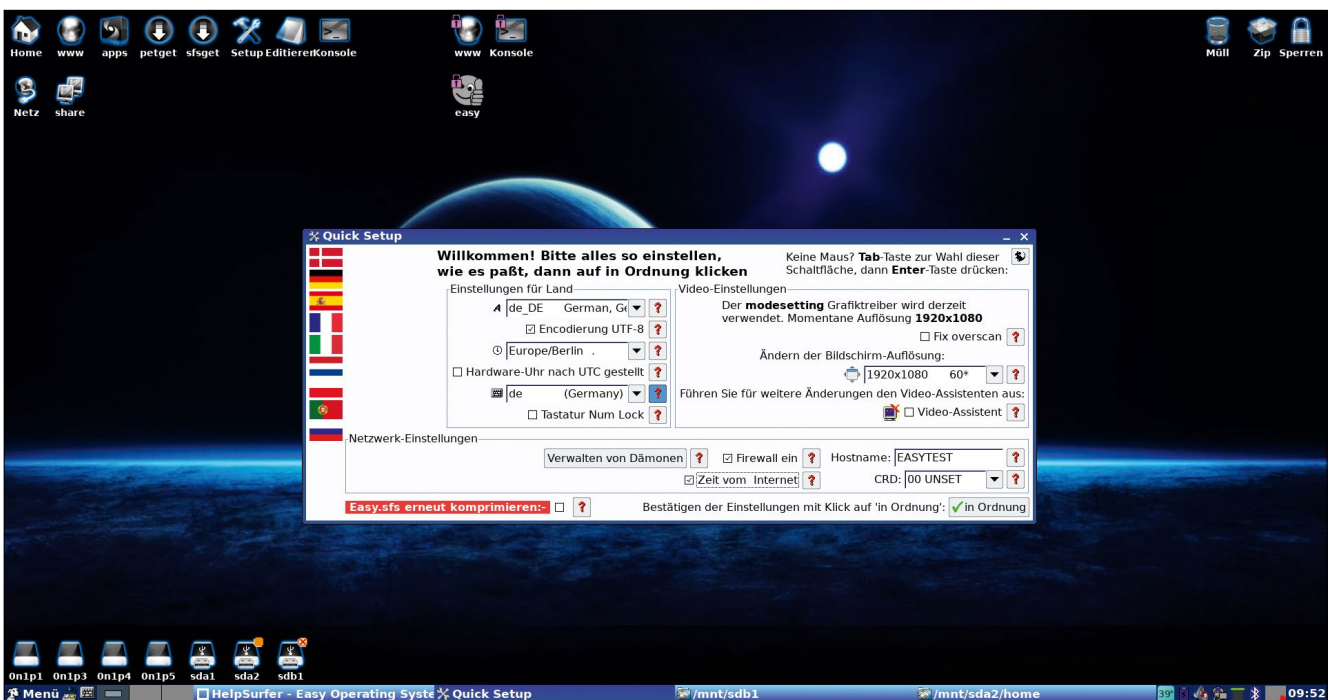
**1** Easy Version Control ist eine der einfachen, aber effektive grafische Oberfläche zum Verwalten verschiedener Versionen des Systems inklusive eines bereits erstellen Snapshots.

**2** Schematische Darstellung der beiden Partitionen eines USB-Sticks mit EasyOS. Das Verzeichnis .session/ enthält die aktuelle Sitzung und wird durch AuFS beschreibbar. (Quelle: bkhome.org)

Auf der Download-Seite [finden Sie](#) zwei Image-Dateien mit den Endungen .img.gz und .iso, die verschiedenen Zwecken dienen. Möchten Sie EasyOS live oder in einer virtuellen Maschine ausprobieren, greifen Sie zur ISO-Datei. Das gepackte Image von EasyOS, das auf Deutsch bereitsteht, eignet sich dagegen zum Booten von einem USB-Stick.

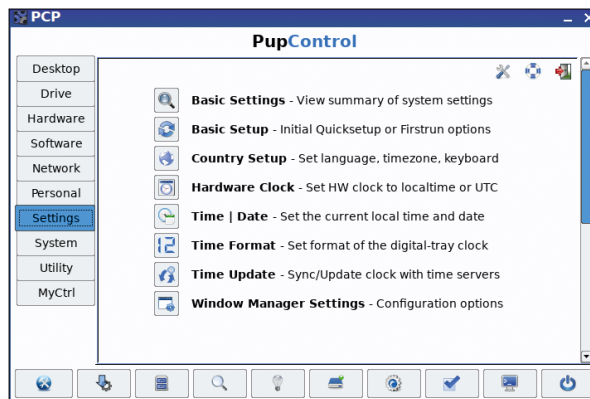
Für den Transfer beider Abbilder auf externe Datenträger nutzen Sie entweder Dd oder das grafische Tool Etcher [Für den alternativen Live-Betrieb brennen Sie das ISO-Image auf eine CD oder DVD und booten den Rechner damit. Auch diese Version lässt sich auf einer Festplatte installieren \[Für das nicht dem eigentlichen Verwendungszweck von\]\(#\)](#)

  
EasyOS 1.0.8  
bootfähig auf Heft-DVD  
EasyOS 1.0.14  
[LU/easyos/](http://LU/easyos/)



**3** Das Startfenster Quick Setup erlaubt einige grundsätzliche Einstellungen im Bezug auf Lokalisierung, Grafiktreiber und Netzwerk.

**4** Das von Puppy Linux ausgeliehene *PupControl* fasst die verschiedenen Einstellungsdialoge unter einer Oberfläche zusammen.



EasyOS entspricht, lässt die Dokumentation nur erahnen. Wie auch Puppy dient EasyOS hauptsächlich dazu, von Wechselmedien wie USB-Sticks oder SD-Karten zu starten, und bietet sich daher als Zweitsystem für die Hosentasche an.

### Besonderheiten

Beim Einrichten auf der Festplatte handelt es sich um eine sogenannte frugale Installation, die das System nicht nach den Regeln des Filesystem Hierarchy Standard (FHS) ausrollt, sondern alles in einem Ordner auf der gewählten Partition belässt. Weitere eventuell auf dieser Partition gespeicherte Inhalte bleiben unangetastet. Frugale Installationen bieten unter anderem Knoppix, MX Linux und Damn Small Linux an.

Eines der Alleinstellungsmerkmale von EasyOS besteht darin, Container innerhalb des Systems zu verwenden, um Anwendungen zu isolieren. Kauler setzt hier nicht auf Docker oder LXC, sondern verwendet eine Eigenentwicklung mit nur wenigen KByte kleinen Containern, die kaum Overhead erzeugen.

Zudem erlaubt EasyOS das Erstellen eines System-Snapshots, der das Zurückrollen des Systems erlaubt. Diese Funktion schließt auch mögliche Container ein. Auf diese Weise halten Sie bei Bedarf zwei Systemzustände vor, zwischen denen Sie wechseln können. Analog zu atomaren Paket-Updates wie etwa bei Fedora Silverblue genügt bei EasyOS das Ersetzen von nur drei Dateien, um es zu aktualisieren.

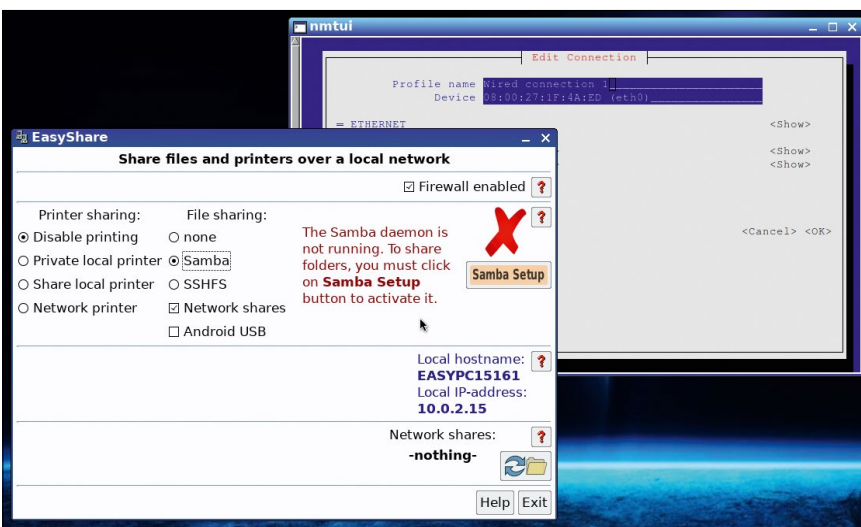
Die Bootpartition enthält die Dateien `mlinuz`, `initrd` und `easy.sfs`. Dabei handelt es sich um den Kernel, die Bootkonfiguration und um EasyOS selbst. Erscheint eine neue Version des Systems, genügt es dann, lediglich diese drei Dateien gegen die aus dem neuen Abbild auszutauschen. Anschließend klicken Sie auf die neue Initrd, um sie an die Gegebenheiten der Hardware anzupassen. Ein Bestätigen der Abfrage nach der Korrektheit der Einträge schließt die Aktualisierung dann ab.

Nach einem Reboot läuft dann die neue Version. Unter *Filesystem | Easy Version Control* sehen Sie nun die alte und die neue Version sowie einen eventuell von Ihnen erstellten Snapshot **1**. Der lässt sich alternativ auch in einem Container erstellen.

In der GUI rollen Sie bei Bedarf auf eine ältere Version zurück oder wieder vor auf die neuere. Über den Schalter *Depth* bestimmen Sie die Zahl der gespeicherten Releases; voreingestellt sind drei, maximal möglich fünf.

### Paketbündel

Ein von Intels Clear Linux OS bekanntes Merkmal sind Paketbündel statt einzeln installierter Pakete. Bei EasyOS enden deren Dateinamen auf `.sfs`. Bei den bereits von Puppy her bekannten Mega-Paketen handelt es sich um schreibge-



**5** Das Einrichten des Netzwerks übernimmt das Curses-basierte Werkzeug Nmtui, über *EasyShare* richten Sie das Print- und Filesharing für das System ein.

geschützte SquashFS-Archive, deren Pakete man nicht installiert, sondern bei Bedarf zur Laufzeit einhängt. Das hält die installierte Paketbasis klein und spart Platz. EasyOS setzt das mittels des Aufs-Overlay-Dateisystems um, das auch zum scheinbaren Beschreiben von Live-CDs zum Einsatz kommt **2**.

Startet EasyOS das erste Mal vom USB-Stick, lädt es zunächst das Image vom Stick in den Hauptspeicher. Anschließend legt es auf dem Stick zwei Partitionen an: eine mit FAT32 formatierte Bootpartition mit einer Größe von 640 MByte und eine Ext4-Partition, die den Rest des Sticks einnimmt. Das System beansprucht nach der Initialisierung etwa 700 MByte davon.

In dieser „Working Partition“ speichert die Distribution persistent die mit dem System erstellten Daten. Im Verlauf des Setups fragt der Installer Sie nach einem Passwort, mit dem Sie später auf verschlüsselte Teile des Systems zugreifen.

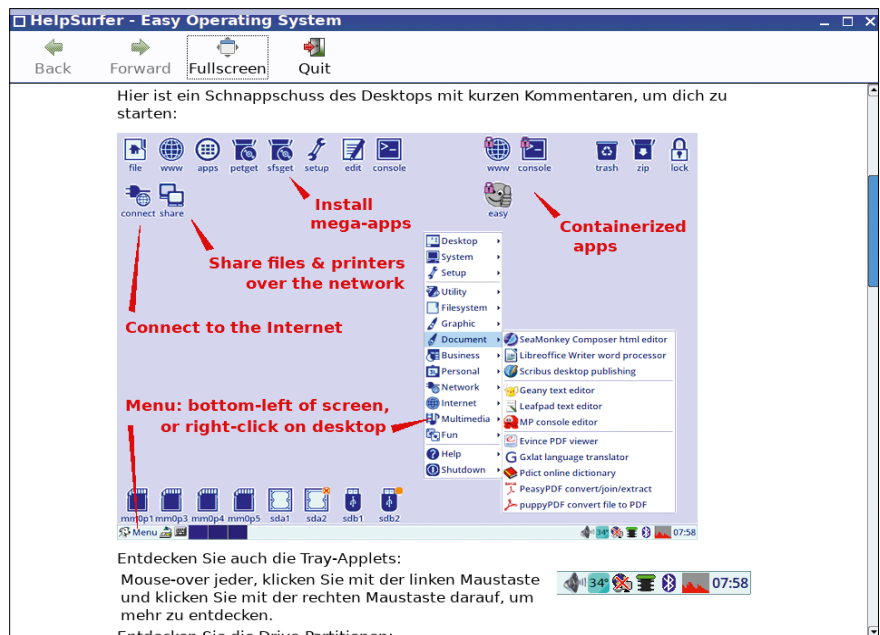
## Root als Standard

Wenn Sie den Bootscreen beim ersten Start aufmerksam verfolgen, dann stellen Sie fest, dass EasyOS Sie automatisch als Benutzer *root* einloggt. Diese Eigenart, die Kauler bei vielen der Varianten von Puppy Linux nutzte, beruht darauf, dass er die Distribution als Einzelplatzsystem konzipierte. So lassen sich darin ohne Passwordeingabe sämtliche Systemveränderungen vornehmen.

Für Webapps und ähnlich exponierte Anwendungen steht bei Bedarf der in seinen Rechten beschränkte User *spot* zur Verfügung, in Containern tritt *zeus* an seine Stelle. Dort läuft generell ein sogenannter crippled root, dessen Rechte EasyOS über diverse Kernel-Mechanismen stark reduziert.

## Einstellungssache

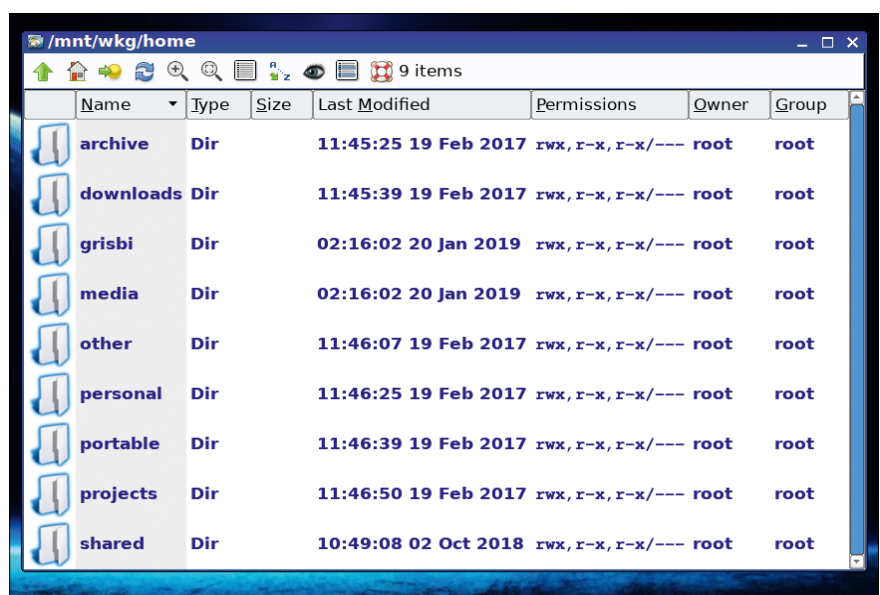
Nach dem ersten Einrichten erwartet Sie zunächst das *Quick Setup* **3**. Die Einstellungen zum Lokalisieren erklären sich selbst, die Videoeinstellungen rechts im Fenster waren bei Verwendung einer Intel-Grafikkarte von Haus aus korrekt.



**6** Der *HelpSurfer* erläutert die auf dem Desktop befindlichen Icons. Die drei Exemplare mit Vorhängeschloss führen zu Containern. Hinter *Easy* verbirgt sich ein containergeladertes, komplettes EasyOS.

Beim Versuch, den Intel-Treiber gegen den Kernel-Modeseitig-Treiber aus X.org zu ersetzen, erwies sich die Routine dann allerdings als nicht sehr stabil. Wir mussten in diesem Fall von vorne beginnen, da wir nicht mehr in eine grafische Umgebung gelangen konnten.

Kauler sieht im Quick-Setup die praktische Möglichkeit vor, EasyOS aus dem laufenden System heraus neu zu komprimieren. Nach einem Reboot wurde das System somit neu aufgesetzt. Beim zweiten Versuch gelang die Umstellung dann jedoch problemlos. Generell emp-



**7** Als Dateimanager kommt der schlanke ROX-Filer aus dem gleichnamigen Desktop zum Einsatz, der für alltägliche Operationen in der Regel passende Funktionen enthält.

fehlen wir aber, diese Einstellung nur im absoluten Notfall zu verändern.

## Schlanker Desktop

Ein Klick auf das Desktop-Icon *Setup* öffnet den aus Puppy Linux stammenden *PupControl*, der die wichtigsten Einstellungsdialoge in einer Oberfläche zusammenfasst **4**. Klicken Sie im Reiter *Network* auf den untersten Schalter *Network Wizard*, dann öffnet sich der *Internet Connection Wizard*, mit dem Sie die Verbindung ins Internet einrichten.

Den Curses-basierten Netzwerkmanager *Nmtui* erreichen Sie im Menü unter *Network | Nmtui NetworkManager configuration*. Die Schaltfläche *EasyShare network file and printer sharing* im selben Abschnitt öffnet den Einstellungsdialog *EasyShare*, in dem Sie das Print- und File-sharing (de-)aktivieren **5**.

Nach der Grundkonfiguration erscheint der *Helpsurfer*, der ins System einführt und unter anderem die Desktop-Icons erläutert. Das ist auch nötig,

da Kauler nicht nur keinen Wert auf zeitgemäßes Design legt, sondern einige der Icons ihren Sinn auf den ersten Blick nur Puppy-Kennern offenbaren **6**.

Das ungewohnte Design liegt zu großen Teilen am verwendeten Fenstermanager *JWM* (Joe's Window Manager) **6**. Er kam bereits bei Puppy, Damn Small Linux, Tiny Core Linux und Simplix zum Zug. Damit setzt EasyOS auf einen Desktop, der zunächst fremd und altmodisch anmutet. Durch seinen geringen Ressourcenverbrauch geht EasyOS außergewöhnlich flott zu Werk, was die Arbeit mit dem Desktop und das Starten von Programmen angeht. Als Dateimanager verwendet die Distribution den ROX-Fileer, einen Teil des ROX-Desktops **7**.

In der traditionellen Taskleiste am unteren Rand finden Sie links das Menü, das Zugriff auf alle installierten Anwendungen und die von Kauler erstellten GUIs zur Handhabung von Containern, zum Entfernen von Paketen aus SFS-Dateien oder für die vielen weiteren Konfigurationsmöglichkeiten bietet. Sie

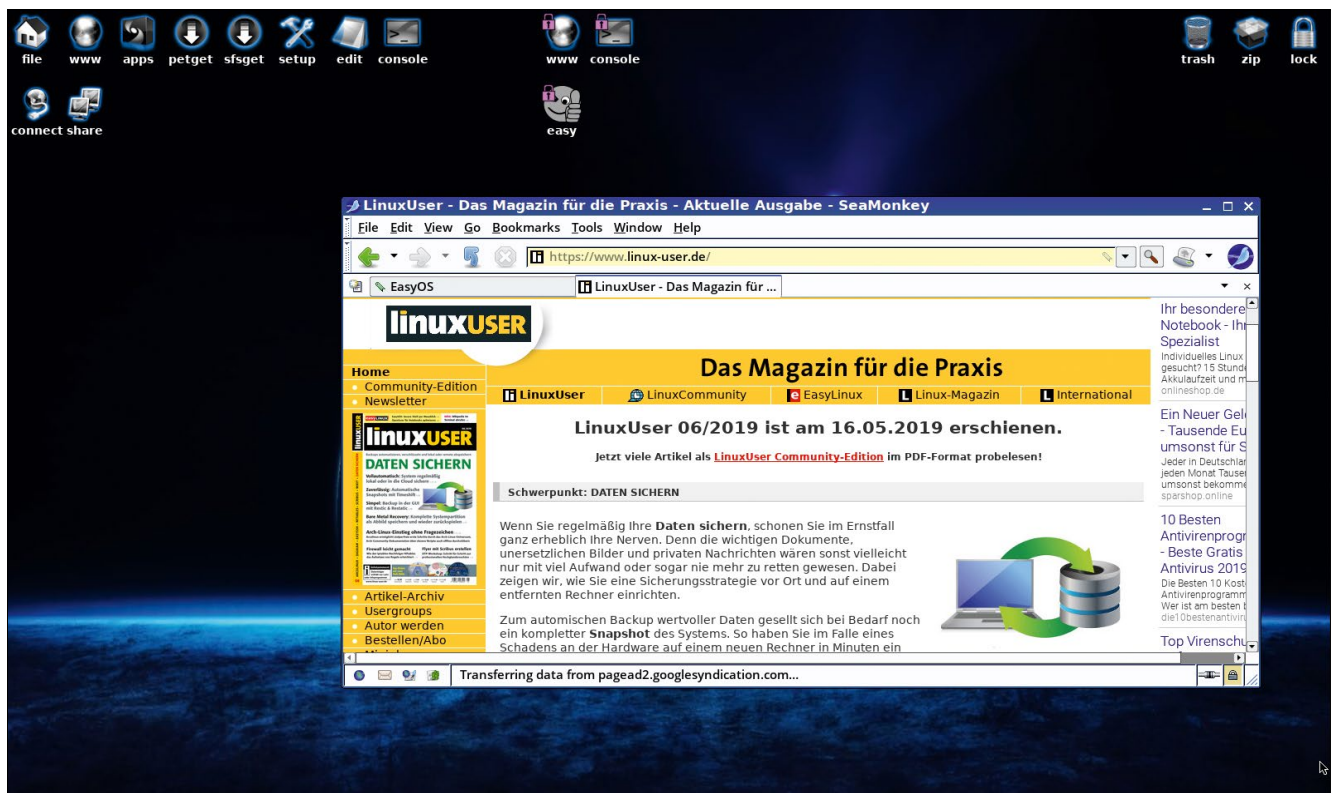
erreichen dieses Menü auch durch einen Rechtsklick auf den Desktop.

## Umfangreiche Auswahl

Was die auf dem relativ kleinen Image von knapp über 400 MByte Platz versammelten Anwendungen betrifft, sammelt EasyOS richtig Punkte. Kauler zeigt bei der Auswahl viel Erfahrung, und durch die SFS-Archive spart er viel Platz bei trotzdem blitzschnell startenden Anwendungen.

Bei der Auswahl setzte der Entwickler auf eher leichtgewichtige Alternativen zu den üblichen Standardanwendungen. So kommt etwa die Seamonkey-Suite zum Zug, die neben einem Webbrowser einen E-Mail- und IRC-Client mitbringt **8**. Auf Wunsch lassen sich aber Firefox und Chromium nachinstallieren.

In der grafischen Abteilung glänzt EasyOS neben vielen kleinen Werkzeugen mit Gimp, Inkscape und LibreOffice Impress. Für Büroarbeiten bieten sich LibreOffice Writer und Scribus an, er-



**8** Die Browsersuite Seamonkey, ehemals Mozilla Application Suite, umfasst neben einem Webbrowser unter anderem einen E-Mail- und IRC-Client, einen HTML-Editor, ein Adressbuch und diverse Hilfsprogramme für den Einsatz im Web.

gänzt durch mehrere Texteditoren und Tools. Die Sparte *Business* bringt Home Bank, LibreOffice Calc und Base mit. In der Rubrik *Multimedia* finden sich neben Xine und Audacious wiederum eine größere Anzahl an kleinen Helferlein für jede erdenkliche Aufgabe.

## Easy Containers

Das Aufsetzen und Konfigurieren von Containern übernimmt *Easy Containers*, das Sie im Menü unter *Filesystem | Easy Container Management* finden. Das Tool enthält zusätzlich einen Expertenmodus, der unter anderem die Kontrolle über Sicherheitsmerkmale wie Kernel-Capabilities und Namespaces für den zu erstellenden Container erlaubt [9](#).

Standardmäßig existiert neben Containern für den Browser und das Terminal bereits ein Container, der einen kompletten Klon des Betriebssystems enthält. Wenn Sie aus dem Easy-Container heraus arbeiten möchten, klicken Sie auf dem Desktop das Icon mit dem Titel *easy* und dem Vorhängeschloss an.

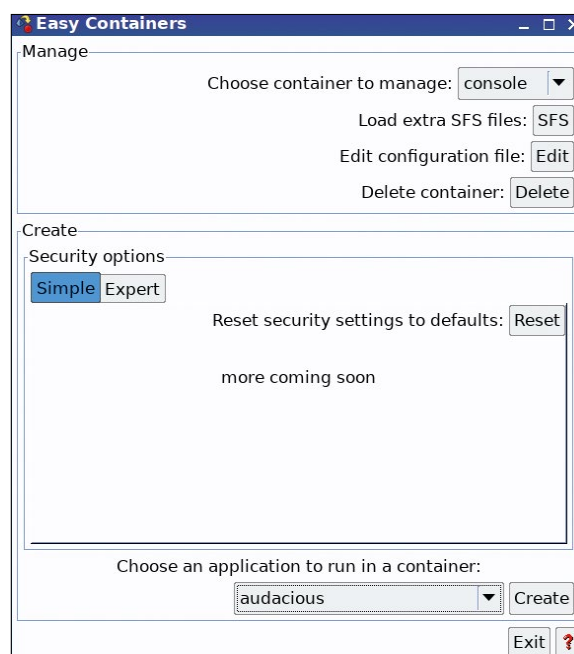
Hier lassen sich Pakete installieren, die Sie mit eingeschränkten Rechten starten möchten. Um wieder ins Hauptsystem zu wechseln, drücken Sie [Alt]+[F6]. Wir konnten aus der GUI heraus problemlos einen Container mit dem zusätzlichen Betriebssystem Xenial-Pup einrichten und es innerhalb von EasyOS nutzen [10](#).

## Fazit

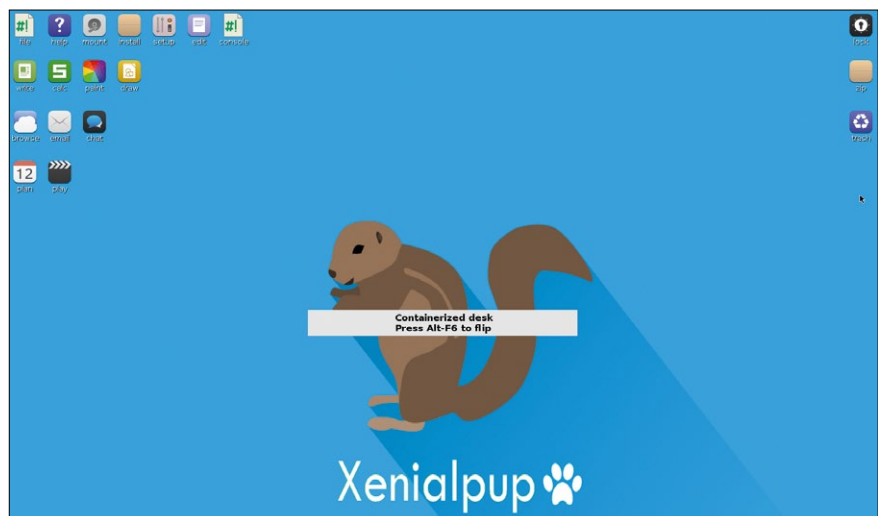
Stellt die Distribution EasyOS mehr als ein Experiment dar, und lässt sie sich sinnvoll im Alltag einsetzen? Nach unserer Einschätzung: ja, allerdings nicht für jeden. Sicher gibt es Anwender, die den Funktionsumfang als wie für ihren Arbeitsfluss gemacht empfinden, andere dürften eher nur ein Experiment darin sehen. EasyOS setzt aber in jedem Fall voraus, dass Sie sich an das überholte

Design gewöhnen. Kauler geht pragmatisch vor und gibt der Funktion Vorrang vor Form sowie Design.

Die Dokumentation auf der Webseite des Projekts führt sehr gut in die Materie ein und hilft beim Finden der Funktionen [8](#). Der einzige grobe Fehler, den wir im Test entdeckten, war die Umstellung des Grafiktreibers im Quick-Setup. Weitere Stabilitätsprobleme traten dagegen nicht auf, kleine Fehler und unfertige Übersetzungen trüben den positiven Gesamteindruck ein wenig. (tle) ■



[9](#) Eine weitere einfache GUI verwaltet die Container innerhalb des Systems. Im *Expert*-Modus lassen sich gezielt Kernel-Funktionen zum Absichern der Container nutzen.



[10](#) Innerhalb eines Containers von EasyOS lässt sich bei Bedarf eine andere Distribution, im Beispiel Xenialpup, erstellen und nutzen. Das Setup erfolgt über die passende GUI.



Weitere Infos und interessante Links

[www.linux-user.de/qr/42936](http://www.linux-user.de/qr/42936)



Mit Timegrapher  
Uhrwerke mechanischer Armbanduhren  
prüfen und bewerten

# Tickst du richtig?

© alexanderdn, 123RF

Mechanische Armbanduhren unterliegen natürlichem Verschleiß. Wir zeigen, wie Sie mithilfe des Programms Timegrapher frühzeitig auf Unregelmäßigkeiten aufmerksam werden. Erik Bärwaldt

## README

Sogenannte Zeitwaagen für die Prüfung von mechanischen Uhrwerken kosten meist mehrere Hundert Euro. Mit einem Linux-Notebook und dem Programm Timegrapher können Sie sich diese Ausgaben ersparen.

Der Markt für mechanische Armbanduhren boomt. Dabei haben es die kleinen Wunderwerke der Technik nicht nur passionierten Uhrenliebhabern angetan: Sie erfreuen sich auch bei Investoren wachsender Beliebtheit, die in Zeiten niedrigster Zinsen nach lukrativen Anlagemöglichkeiten Ausschau halten.

Doch die Freude wird zunehmend getrübt, teils durch meist fernöstliche Fälschungen hochwertiger Uhren und teils durch schlecht gewartete und kaum noch funktionstüchtige Gebrauchtuhren namhafter Hersteller. Um die Spreu vom Weizen zu trennen, leistet neben einschlägiger Sachkenntnis eine sogenannte Zeitwaage wertvolle Hilfe.

Doch solche Geräte kosten teilweise vierstellige Euro-Beträge und lohnen sich daher eigentlich nur für den Einsatz in professionellen Uhrmacherwerkstätten. Doch es geht auch günstiger: Mit einem Linux-PC und dem Programm Timegrapher kommen Sie Mängeln am Uhrwerk auf die Spur, ohne dazu gleich ein kleines Vermögen investieren zu müssen.

## Technik

Eine Zeitwaage ermittelt anhand der typischen Geräusche, die Armbanduhren mit Ankerhemmung erzeugen, die Ganggenauigkeit und andere Parameter. Diese geben über den Zustand des Uhrwerks Aufschluss.

Üblicherweise montiert man dazu die Armbanduhr auf einem kleinen Sockel oder einem Gestell in unmittelbarer Nähe eines Mikrofons, das die vom Uhrwerk ausgehenden Geräusche aufnimmt und an die Zeitwaage weiterleitet. Diese berechnet daraus die Amplitude sowie die Gangabweichung und zeigt bei modernen Modellen die ermittelten Werte samt entsprechenden Diagrammen auf einem Display an.

Das Betriebsgeräusch einer Uhr mit Ankerhemmung besteht aus drei periodisch auftretenden Komponenten, von denen sich zwei für die Auswertung durch die Zeitwaage eignen: Trifft im Uhrwerk der Hebelstein der Rolle auf die Gabel des Ankers, resultiert daraus das

erste Geräusch. Es ist für den Beginn der Diagrammaufzeichnung der Zeitwaage und somit auch der Berechnung der Gangabweichung relevant.

Beim zweiten Ganggeräusch trifft ein Zahn des Ankerrads auf die Impulsfläche einer Palette, wobei die Anker gabel den Hebelstein berührt. Dieses Geräusch zeichnet die Zeitwaage ebenfalls auf (in aller Regel grafisch), es dient jedoch nicht zur Bestimmung von Ganggenauigkeit oder Amplitude.

Fällt schließlich ein Zahn des Ankerrads auf die Palette, entsteht ein drittes Geräusch, das man zur Bestimmung der Amplitude heranziehen kann. Die Drehung der Unruh zwischen diesen drei Impulsen bezeichnet man im Jargon der Uhrmacher als Hebungswinkel. Die Zeitwaage misst jeweils die Zeit zwischen zwei solchen Gangintervallen und vergleicht sie mit einem Sollwert, der die Ganggenauigkeit angibt.

Die einzelnen Werte visualisiert die Zeitwaage auf einem Display. Hochwertige Zeitwaagen präsentieren dort in der Regel gleich mehrere Diagramme. Unregelmäßig verlaufende Linien oder stark geschwungene Kurven deuten dabei bereits auf Fehler im Uhrwerk hin.

Zusätzlich geben Zeitwaagen neben den grafisch aufbereiteten akustischen

Messwerten des Uhrwerks meist auch die Anzahl der Halbschwingungen der Unruh pro Stunde und die Amplitude in absoluten Zahlen an. Auch die errechnete Gangabweichung erscheint auf dem Display, meist auf Zehntelsekunden gerundet. Die gewonnenen Werte lassen sich bei professionellen Geräten ausdrucken und so beispielsweise zu Dokumentationszwecken archivieren.

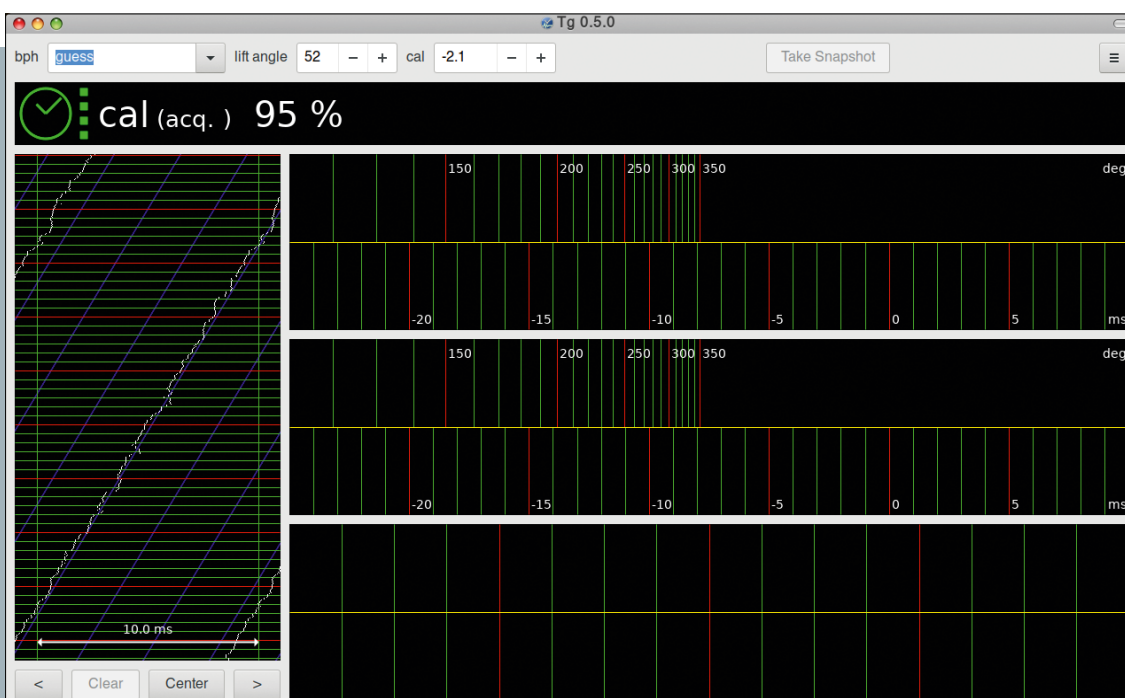
## Timegrapher

Timegrapher erhalten Sie als vorkompiliertes DEB-Paket in Varianten für 32- und 64-Bit-Systeme [☞](#). Auf der Homepage steht zudem der Quellcode zum Herunterladen bereit, wobei die Entwickler für diverse Distributionen Anleitungen zum Kompilieren und Installieren auf der Projektseite [☞](#) vorhalten.

Die Software legt bei Installation der Binärpakete im Menü *Zubehör* einen Starter an und öffnet nach einem Klick darauf ein einfaches Programmfenster. Es fasst verschiedene grafische Anzeigen zusammen, auf denen später die Messdaten nahezu in Echtzeit erscheinen. Am oberen Fensterrand finden Sie außerdem eine Reihe von Datenfeldern, die Informationen zum jeweils getesteten Uhrwerk auflisten.



Timegrapher 0.5.0 (Quellen, DEB)  
[LU/timegrapher/](https://lu.timegrapher/)



**1** Zunächst müssen Sie Timegrapher kalibrieren, was rund eine Viertelstunde dauert.

Links oben im Fenster blendet Timegrapher eine symbolisierte Uhr ein, die er zusammen mit vier kleinen übereinander angeordneten Punkten zunächst in roter Farbe darstellt. Sobald die Software ein ausreichend starkes akustisches Signal aufzeichnet und auswertet, wechselt die Farbe des Uhrensymbols ebenso wie jene der Punkteleiste auf Grün.

Bei der Signalaufnahme erkennen Sie anhand der Zahl der farbig leuchtenden Punkte, ob die Eingangsempfindlichkeit der Soundkarte und des Mikrofons passt: Ein oder zwei Punkte deuten auf ein zu schwaches Signal hin, bei dem eventuell Nebengeräusche die Messung der Gangwerte beeinflussen. Achten Sie deshalb penibel auf eine ausreichend starke Eingangsempfindlichkeit.

## Problemkind Soundkarte

Zunächst müssen Sie über den Pulseaudio-Lautstärkeregler (zumindest bei modernen Distributionen wie Ubuntu) die im Computer verbaute Soundkarte so anpassen, dass das angeschlossene Mikrofon die sehr leisen Geräusche des Uhrwerks zuverlässig erfasst. Dazu modifizieren Sie die Konfiguration der Eingabegeräte für eine möglichst hohe Empfindlichkeit des Mikrofons. Achten Sie dabei darauf, vorhandene externe Lautsprechersysteme aus- oder stummzuschalten, um mögliche Rückkopplungen zwischen Mikrofon und Lautsprechern zu vermeiden.

Nach dem Anpassen der Eingangsempfindlichkeit kalibrieren Sie nun Timegrapher. Die Software verwendet zur Zeitmessung den in der Soundkarte verbauten Timer. Da der oft jedoch nicht sehr präzise arbeitet, gleicht das Programm dessen Daten mit einer weiteren Referenz ab. Dazu bedarf es einer Quarzuhr, die Sie einfach in der Nähe des Mikrofons platzieren. Anschließend öffnen Sie oben rechts im Programmfenster das Menü durch einen Klick auf den entsprechenden Schalter und wählen die Option *Calibrate* an.

Timegrapher gleicht nun die beiden Referenzzeiten miteinander ab und stellt sich entsprechend für weitere Messun-

gen ein. Dabei erscheint rechts neben dem Uhrensymbol die Anzeige *cal (wait)*, gefolgt von einer langsam hochzählenden Prozentangabe. Die Software benötigt zur Kalibrierung nun etwa 15 Minuten. Nach deren Abschluss zeigt die Software die Kalibrierungskonstante als absoluten Sekundenwert pro Tag an und übernimmt den Wert automatisch in das Datenfeld *cal* [1](#).

In einem weiteren Schritt rufen Sie nochmals das Menü auf und deaktivieren dort die Option *Calibrate*. Die Anpassungen, die Timegrapher während der Kalibrierung intern vornimmt, speichert das Programm automatisch, sodass Sie beim erneuten Aufruf der Anwendung nicht nochmal eine Kalibrierung ausführen müssen.

## Hebungswinkel

In einem weiteren vorbereitenden Schritt müssen Sie den Hebungswinkel der Rolle des zu testenden Uhrwerks einstellen. Er unterscheidet sich von Uhrwerk zu Uhrwerk und lässt sich nicht durch die Software ermitteln. Im Internet finden sich jedoch eine Reihe von Listen [☞](#), die Hebungswinkel für gängige Uhrwerke zahlreicher Hersteller zusammentragen.

In Timegrapher stellen Sie den für Ihre Uhr relevanten Wert im Feld *lift angle* ein, indem Sie über das Plus- oder Minus-Symbol den vorgegebenen Winkel modifizieren. Danach steht die Software für eine erste Analyse zur Verfügung.

## Messung

Um nach Abschluss aller Vorbereitungen die erste Messung vorzunehmen, genügt es, die zu bewertende Uhr in direkter Nähe des Mikrofons anzubringen. Die Software erkennt automatisch das Uhrwerksgeräusch und beginnt selbsttätig mit der Messung.

Dabei ermittelt Timegrapher zunächst die Schlagzahl des Uhrwerks und gibt sie oben im Programmfenster aus. Alternativ tragen Sie die Schlagzahl von Hand im Eingabefeld *bph* ein. Achten Sie auf die oben links im Fenster angezeigte Uhr und die daneben angeordnete vertikale

Punkteleiste für das Eingangssignal: Zeigt das Programm alle vier Punkte grün an und erscheint auch das Uhrensymbol in grüner Farbe, vermag Timegrapher das Ganggeräusch korrekt zu erfassen.

Der Algorithmus, den Timegrapher zum Empfang des Signals nutzt, filtert dabei automatisch – und im Test auch sehr zuverlässig – störende Nebengeräusche aus. Bei nur einem oder zwei eingefärbten Punkten und einem in roter Farbe angezeigten Uhrensymbol sollten Sie dagegen die Uhr vor dem Mikrofon neu justieren.

Links unter der Anzeigeleiste mit den Gangdaten zeigt Timegrapher nun fortlaufend, wie auf dem ablaufenden Papier einer herkömmlichen Zeitwaage, den Uherschlag in zwei Linien an. Diese bestehen aus einzelnen Punkten und stehen in direkter Relation zur in blauer Farbe angezeigten Referenzlinie im selben Diagramm. Je weiter sich die gepunkteten Linien von der Referenzlinie entfernen und je unregelmäßiger sie verlaufen, desto schlechter ist das Uhrwerk eingestellt.

Rechts im Fenster sehen Sie die Hemmungsgeräusche in Echtzeitdarstellung, mitsamt den Amplitudenwerten, die ebenfalls in Echtzeit variieren und durch eine blaue Linie dargestellt werden. Oben in der Zahlenleiste sehen Sie nun neben der Schlagzahl im Bereich *deg* auch die Amplitude, den Abfallfehler sowie die Ganggenauigkeit des Uhrwerks, wobei Timegrapher die Abweichung in Sekunden pro Tag angibt [2](#).

Die Messung muss einige Minuten laufen, damit sich die Werte stabilisieren. Danach lässt sich der Inhalt des aktuellen Bildschirms über den Schalter *Take Snapshot* im Programmfenster in einem neuen Reiter festhalten. Mit den Optionen *Save current display* oder *Save all snapshots* speichern Sie die Snapshots dann auf der Festplatte ab, um sie für einen späteren Vergleich erneut heranziehen zu können.

## Interpretation

Die von Timegrapher ermittelte Amplitude sollte zwischen 150 und 280 liegen.

Amplitudenwerte von jenseits der 300 deuten darauf hin, dass entweder die Software nicht korrekt eingestellt wurde oder ein sogenanntes Pellen auftritt, was das Uhrwerk im schlimmsten Fall beschädigt. In einem solchen Fall sollten Sie die Messung mehrfach wiederholen und die Uhr revisionieren lassen, falls die Amplitude dauerhaft zu hoch ausfällt.

Üblicherweise besitzen „gesunde“ mechanische Uhrwerke eine Amplitude von zwischen 240 und 290. Um aussagekräftige Messwerte zu erhalten, sollten Sie dabei darauf achten, die Uhr in verschiedenen Lagen zu testen. Fällt die Amplitude dabei generell unter Werte von etwa 150 bis 170, empfiehlt es sich, das Uhrwerk überholen zu lassen.

Solche niedrigen Werte deuten bei älteren Uhrwerken auf verhartetes Öl oder eine mangelnde Schmierung hin, was beides den Verschleiß erhöht. Niedrige Amplitudenwerte nur in bestimmten Lagen entstehen häufig durch defekte Lager. Die Gangwerte weichen in aller Regel einige Sekunden von der Norm ab. Sobald sich hier jedoch Werte von mehr

als circa 20 Sekunden pro Tag ergeben, sollten Sie die Uhr einregulieren lassen.

## Fazit

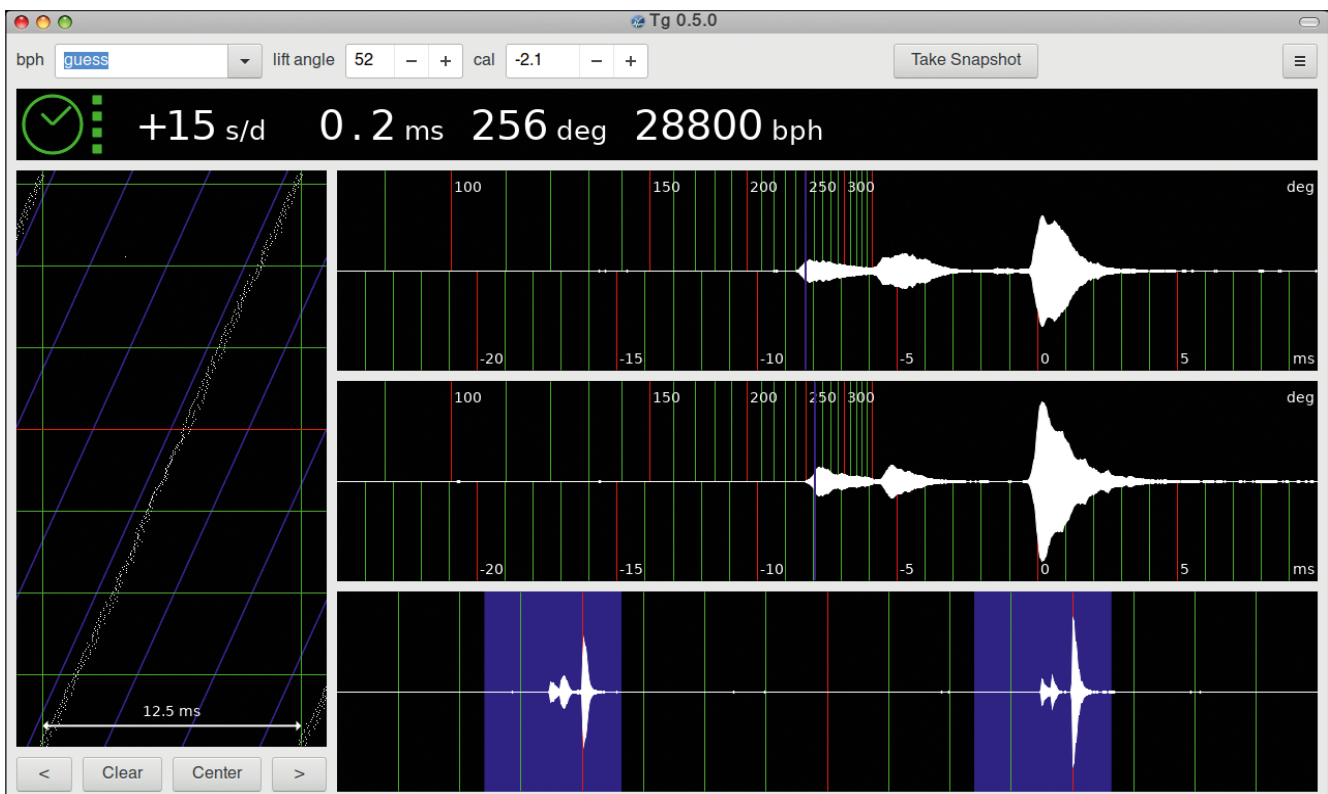
Timegrapher bietet für Uhrenliebhaber, Sammler und Investoren eine sehr gute Möglichkeit, den Regulierungs- oder gar Revisionsbedarf eines Uhrwerks festzustellen. Dabei fällt die Software gleich mehrfach positiv auf: Sie braucht keinerlei zusätzliche Komponenten wie Körperschallmikrofone oder Verstärker, um das Ganggeräusch zuverlässig aufzuzeichnen, und sie ermittelt alle in diesem Zusammenhang üblichen Werte.

Auch in Umgebungen mit Nebengeräuschen arbeitet Timegrapher zuverlässig. Durch die Möglichkeit, Ergebnisse zu speichern, lassen sich zudem Vergleichswerte über die Zeit ermitteln. Für den professionellen Einsatz empfiehlt sich der Einsatz eines zusätzlichen Uhrenträgers, an dem sich dann auch ein Mikrofon zum Anschluss an die Soundkarte anbringen lässt. Die Software steht somit einfachen Zeitwaagen in nichts nach. (cla) ■



Weitere Infos und interessante Links

[www.linux-user.de/qr/43046](http://www.linux-user.de/qr/43046)



2 Timegrapher stellt alle Ergebnisse übersichtlich dar.

McFly wertet die Bash mit künstlicher Intelligenz auf

# Intelligent geordnet

Für das effektive Arbeiten mit der Befehlszeile ist die History eines der wichtigsten Features. McFly erweitert deren Funktionen so, dass Sie schneller die gewünschten Befehlszeilen finden.

Karsten Günther

## README

Dass das maschinelle Lernen immer größeren Einfluss in der Software gewinnt, zeichnete sich seit Langem ab. Trotzdem überrascht es, dass das Konzept mittlerweile sogar bei Shells eine wichtige Rolle spielt.

Die unter Linux mit Abstand beliebteste Shell ist die GNU Bourne-Again Shell oder kurz Bash. Zu deren vielen hervorragenden Funktionen zählt der integrierte Befehlsverlauf: In der sogenannten History speichert die Bash bereits ausgeführte Befehlszeilen, sodass man sie später ohne viel Tipperei erneut aufrufen kann.

Geben Sie in der Befehlszeile das Kommando `history` ein, erhalten Sie eine Ausgabe, die jener aus [Listing 1](#) ähnelt. Jede Zeile beginnt mit einer aufsteigend nummerierten Zeilennummer. Die zuletzt eingegebene Befehlszeile steht ganz hinten in der History. Mit den Pfeiltasten blättern Sie im Terminal einen Schritt in der History zurück oder vor. Mit einem Druck auf die Eingabetaste übernehmen Sie die angezeigte Zeile und führen sie erneut aus.

Die Bash kennt zwei Möglichkeiten, direkt in der History nach bestimmten Zeichenketten zu suchen: `[Strg]+[R]` sucht von der aktuellen Cursor-Position aus rückwärts, `[Strg]+[S]` vorwärts. Die

zweite Variante ist nur dann sinnvoll, wenn Sie sich in der History bereits weiter zurückbewegt haben und sich nicht mehr an deren Ende befinden. Zudem funktioniert die Eingabe von `[Strg]+[S]` nur in einem entsprechend konfigurierten Terminal, normalerweise wird diese Tastenkombination blockiert.

Die durch `[Strg]+[R]` aktivierte Rückwärtssuche in der History zeigt die Bash durch einen speziellen Prompt an:

```
(reverse-i-search)` `:
```

Der Begriff `i-search` steht dabei für eine inkrementelle Suche, die Sie durch jedes neu eingegebene Zeichen weiter verfeinern. Die Funktion findet die zuletzt eingegebenen Zeilen als Erste, mit jedem neu eingegebenen `[Strg]+[R]` springt die Suche um einen Treffer weiter in die Vergangenheit.

In der Praxis erweist sich diese Form der Suche in der History als ebenso einfach wie effektiv: Wenige geschickt ge-

wählte Eingaben genügen, um die gewünschte Zeile zu finden. Sie müssen allerdings wissen, wonach Sie suchen, und vor allen Dingen ein gutes Muster kennen, das schnell zum gewünschten Kommando führt. Dabei darf das Muster an jeder beliebigen Position in der Befehlszeile stehen; Sie müssen die Zeile also nicht von vorn beginnend eingeben.

Um in [Listing 1](#) etwa das Kommando `mv /tmp/linify.png .` hervorzuholen, würde die Eingabe `[Strg]+[R],[I],[F],[Y]` genügen. Befehlszeilen wie `trizen -Syu` oder `yay -Syu` lassen sich schnell über `[Strg]+[R],[Umschalt]+[S]` oder `[Strg]+[R],[Umschalt]+[S]` finden.

Wie [Listing 1](#) zeigt, enthält die History jedoch meist viele Zeilen, die einander sehr ähneln und sich zwar gut mit Mustern beschreiben, aber nicht gut auseinanderhalten lassen. Entsprechende Abhilfe erfordert einen neuen Ansatz.

## Moderne Zeiten

Der History fehlt jegliche Möglichkeit, kontextbasiert zu agieren. Ein Beispiel: Bei vielen modernen, auf Systemd basierenden Systemen finden sich unter `/var/log/journal/` große Mengen von Protokolldateien. Durch das Löschen veralteter Log-Files lässt sich oft viel Speicherplatz freigeben.

Um nachzusehen, ob und wie viele alte Log-Dateien dort lagern, wechselt man sinnigerweise in das Verzeichnis. Das Kommando `du -sh` ermittelt dann den von den vorhandenen Dateien ver-

wendeten Speicherplatz. Mit `find -mtime 7` sucht man dann beispielsweise alle Dateien heraus, die mehr als eine Woche nicht mehr verändert wurden, und löscht sie mit `rm [...]`.

Sie könnten nun für diese Aufgabe in wenigen Augenblicken ein Skript oder besser eine Shell-Funktion schreiben, aber das Problem bleibt dasselbe: Man muss für jede Situation aktiv eine Lösung entwickeln. Wäre es da nicht viel besser, wenn die Shell beispielsweise anhand des Verzeichnisses schon erriete, was Sie vorhaben, und Ihnen die entsprechenden Aktionen bevorzugt anbietet?

Genau hier setzt McFly [an](#). Das Programm liest die bisherige History ein, analysiert sie dabei in einem synthetischen neuronalen Netz und versucht durch maschinelles Lernen Zusammenhänge in den Befehlszeilen zu erkennen. Diese Zusammenhänge dienen dann als Basis für die Vorschläge, die Ihnen die Suchfunktionen der Shell anbieten.

Im Gegensatz zum recht komplizierten Aufbau des neuronalen Netzes lässt sich der letzte Schritt beim Einsatz von McFly ganz einfach verstehen: Die normalerweise an `[Strg]+[R]` gebundene Funktion `previous-history` wird durch die neue Funktion `mcfly search` ersetzt.

Als Basis für die Auswertung der History und die McFly-Vorschläge bei der Suchfunktion dienen die folgenden Parameter, in der angegebenen Reihenfolge:

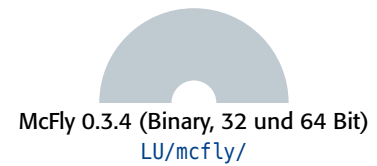
- das aktuelle Verzeichnis,
- die (dort) zuvor eingegebenen Befehlszeilen,

```

test@test-VirtualBox: ~
McFly | ESC - Exit | ⌘ - Run | TAB - Edit | F2 - Delete
$
ls -al
lsblk
cd /etc/
sudo reboot
sudo apt update
mc
tp a
htop
cd Desk
joe todo.txt

```

**1** McFly erweitert die Suche nach Befehlen in der Kommando-History der Shell.



## TIPP

Kombinieren Sie `history` mit `grep`, um alle Zeilen der History zu finden, in denen ein vorgegebenes Muster auftritt ([Listing 1](#), zweiter Aufruf). Mit `agrep -B` anstelle von `grep` suchen Sie bei Bedarf „unscharf“ in den Ergebnissen.

## Listing 1

```

$ history
[...]
446 nano .config/GIMP/2.10/
scripts/image-subdivide.scm
447 nano .config/GIMP/2.10/
scripts/slice-and-join.scm
[...]
901 yay -Syu
902 trizen -Syu
903 yay -Syu
[...]
1044 ++
1045 ++ /tmp/
1046 mv /tmp/*png . -v
1047 ++
1048 mv /tmp/*png . -v
1049 ++
1050 mv /tmp/linify.png .
$ history | grep slice
447 nano .config/GIMP/2.10/
scripts/slice-and-join.scm
1051 history | grep slice

```

- die Häufigkeit und Reihenfolge des Aufrufs,
- ob eine Befehlszeile schon zuvor durch McFly gefunden wurde, und
- ob das Kommando ohne Fehler durchlief.

Nach dem Einrichten von McFly (siehe Kasten [McFly installieren](#)) ändert sich das Verhalten der Shell: Die Tastenkombination [Strg]+[R] führt direkt zum Aufruf des Tools, die Titelleiste ändert sich entsprechend, und neue Tastenbindungen werden aktiv [1](#).

**TIPP**

Sollte McFly bei Ihnen mit einer kryptischen Meldung ähnlich der aus [Listing 4](#) abstürzen, dann kämpfen Sie mit einem Bug, der auch viele andere Anwender betrifft [2](#). Dabei bringen Kommandos mit Sonderzeichen das Programm aus dem Tritt. Im Test passierte dies durch den aus versehentlich eingegebenen „Befehl“ `^@^@^ [...] ^@^@sudo`. Solange die Entwickler das Problem nicht beheben, müssen Sie sich damit behelfen, einfach die betreffende Zeile mit einem Editor aus der `~/.bash_history` zu löschen.

**Künstliche Intelligenz**

Beim Initialisieren liest McFly die bestehende History ein und wertet sie aus, was einige Zeit benötigt. Mit [Esc] verlassen Sie diesen Modus wieder und kehren direkt zur Eingabeaufforderung zurück. Über die Pfeiltasten nach oben und unten navigieren Sie in den von McFly angebotenen Empfehlungen.

Geben Sie einen Suchbegriff ein, holt McFly passende Treffer aus der History. Mit einem Druck auf [Eingabe] überneh-

men Sie die gerade hervorgehobene Zeile aus dem Verlauf und führen das Kommando umgehend aus.

Mit der Tabulatortaste hingegen übertragen Sie die hervorgehobene History-Zeile in den Eingabeprompt, ohne das Kommando gleich auszuführen. So lässt es sich noch bearbeiten, mit [Eingabe] führen sie es anschließend aus. Mit [F2] lassen sich einzelne Zeilen nach Rückfrage aus der Vorschlagsliste löschen.

Der Witz an McFly liegt in der zugrundeliegenden künstlichen Intelligenz, also dem zugrundeliegenden neuronalen Netzwerk. Was früher eines riesigen Aufwands bedurfte und ganze Rechenzentren benötigte, lässt sich heute (mit Abstrichen) sogar auf einem Raspberry Pi ausführen. Solche kleinen neuronalen Netze kommen heute für alle möglichen Aufgaben infrage, von der Bildbearbeitung bis hin zur Zeichen- und Mustererkennung.

**Probleme**

Als problematisch erweist sich dabei oft die fehlende Logik dieser Systeme: Sie lernen Zusammenhänge, ohne dass sich in jedem Fall klar nachvollziehen lässt, wie sie das tun.

Bei McFly kann man das Lernverhalten jedoch noch erkennen, zumindest rudimentär. Rufen Sie einen Befehl aus der Vorschlagsliste auf, platziert McFly das Kommando beim nächsten Aufruf der Vorschlagsliste weiter vorne. Weniger nachvollziehbar ist jedoch, welche Befehle initial in welcher Reihenfolge in der Vorschlagsliste auftauchen. Die hierbei verwendeten Regeln erscheinen bereits so komplex, dass es einigen Aufwands bedarf, ihre Wirkung vorherzusagen.

Ein ganz ähnliches System begegnet uns übrigens täglich bei der Nutzung von Internet-Suchmaschinen. Auch hier lässt sich oft nicht erfassen, wie, wo und warum bestimmte Ergebnisse in der Trefferliste erscheinen und andere oft nicht. Diese Problematik tritt grundsätzlich beim Einsatz neuronaler Netze auf und ist nicht ganz ungefährlich. Daher mahnen IT-Forscher immer wieder zum vorsichtigen Umgang mit der Technik [3](#).

**Listing 2**

```
01 $ sudo apt install linuxbrew-wrapper
02 $ brew
03 $ echo 'PATH="/home/linuxbrew/.linuxbrew/bin:$PATH"' >> ~/.profile
04 $ echo 'MANPATH="/home/linuxbrew/.linuxbrew/share/man:$MANPATH"' >>
    ~/.profile
05 $ echo 'INFOPATH="/home/linuxbrew/.linuxbrew/share/info:$INFOPATH"'
    >> ~/.profile
06 $ PATH="/home/linuxbrew/.linuxbrew/bin:$PATH"
07 $ brew tap cantino/mcfly https://github.com/cantino/mcfly
08 $ brew install mcfly
09 $ source "$(brew --prefix)/opt/mcfly/mcfly.bash"
```

**Listing 3**

```
if [ -f $(brew --prefix)/opt/mcfly/mcfly.bash ]; then
. $(brew --prefix)/opt/mcfly/mcfly.bash
fi
```

**Listing 4**

```
Unable to add cmd_tpl to commands: SqliteFailure([...]), src/libcore/
result.rs:1009:5
```

## Fazit

Neben der Autovervollständigung gilt die History als zweite in der interaktiven Nutzung besonders wichtige Eigenschaft der Shell. Hier legt McFly noch eine Schippe Nutzwert obendrauf. Erfreulicherweise ist das Tool als unabhängiges Programm implementiert, statt als Modifikation der ohnehin schon komplexen Bash oder als Plugin dafür. Das minimiert Nebeneffekte und erlaubt ein schnelles, gefahrloses Ausprobieren des Werkzeugs. Was die KI-Leistungsfähigkeit von McFly angeht, greift hier ein typisches Problem selbst-

lernender Systeme: Es ist kaum möglich, verlässliche Aussagen über die Software zu machen, da ihr Einsatz das System selbst zur Laufzeit verändert. In unserem Test lieferte McFly sehr gute Ergebnisse; meist fanden sich die gesuchten Befehlszeilen schon in der ersten Vorschlagsliste.

Geht es nur um eine Erweiterung der Shell-History, finden sich im Open-Source-Universum allerdings auch einfachere Alternativen, wie etwa die „Bash and zsh shell history suggest box“ [Hstr](#) und [Cdhist](#), die ganz ohne neuronale Netzwerke auskommen und deutlich weniger Festplattenplatz benötigen. (cla/jlu) ■



Weitere Infos und interessante Links

[www.linux-user.de/qr/42242](http://www.linux-user.de/qr/42242)

## McFly installieren

McFly fehlt in den Paketquellen der gängigen Distributionen. Lediglich Arch Linux führt das Programm im AUR. Hier spielen Sie das Programm mit einem AUR-Helper, etwa mittels `yay -S mcfly`. Das zieht über Abhängigkeiten eine ganze Reihe weiterer Pakete mit einem Gesamtumfang von einigen Hundert MByte nach.

Bei Ubuntu und anderen Distributionen benötigen Sie zur Installation das Programm Brew, das eigentlich als Paketmanager für MacOS X entwickelt und später für Linux portiert wurde. Je nach Distribution heißt das fragliche Paket *brew*, *linuxbrew*, *homebrew*, *homebrew-bundle* oder – wie bei Ubuntu – *linuxbrew-wrapper*. Mit den Befehlen aus den ersten sechs Zeilen von [Listing 2](#) spielen Sie Brew unter Ubuntu ein und konfigurieren das Programm. Nun richten Sie mit Brew sogenannte Taps ein (Zeile 7), was einige Hundert MByte zusätzlicher Software nachzieht. Dann installieren Sie mittels Brew McFly (Zeile 8), das Sie nun noch initialisieren müssen (Zeile 9).

Zum Abschluss erweitern Sie `$PATH` noch mit dem Ordner der ausführbaren Datei. Dazu übertragen Sie den Inhalt aus [Listing 3](#) in die Datei `~/.bashrc`. Spätestens nach einem Neustart lässt sich dann das Kommando `mcfly` aus dem Terminal heraus aufrufen.

Auf unserem Ubuntu-Testsystem lief dabei alles glatt, jedoch kann bei dem Prozedere einiges schiefgehen: So passiert es eventuell, dass das Verzeichnis `/home/linuxbrew/` sowie der dortige Unterordner `.linuxbrew/` nicht automatisch entstehen. Beim Ausführen von `brew install mcfly` erhalten Sie dann eine entsprechende Fehlermeldung. In diesem Fall müssten Sie das Verzeichnis selbst anlegen beziehungsweise einen symbolischen Link dorthin legen.

Kurz nach Redaktionsschluss Ende Mai hat der McFly-Entwickler auf GitHub erstmals auch fertige 32- und 64-Bit-Binaries seines Programms zum Download bereitgestellt. Sie finden beide auch auf der Heft-DVD dieser Ausgabe.

# LINUX

ONLINE

MAGAZIN

## NEWSLETTER FÜR IT-PROFIS

Newsletter

News

**Stadt Dortmund prüft Einsatz freier Software und offener Standards**  
Die Stadt Dortmund hat das Projekt freie Software und offene Standards als Bestandteil ihres Masterplans für die digitale Stadtverwaltung aufgenommen. In den...

- Tagesaktuelle IT-News
- Security-Infos des DFN-CERT
- Online-Stellenmarkt

Jetzt kostenfrei abonnieren! [www.linux-magazin.de/subscribe](http://www.linux-magazin.de/subscribe)



Das Arch User Repository von Arch Linux im Blick

# Nutzerpakete

Als Rolling-Release-Distribution unterliegt Arch Linux einem stetigen Wandel. Besonders wenn Sie Pakete aus dem AUR laden, sollten Sie die Änderungen im Blick behalten. Christoph Langner

## README

Über das Arch User Repository oder kurz AUR lassen sich unter Arch Linux nicht offiziell unterstützte Programme leicht installieren. Dabei gilt es jedoch, ein paar Regeln einzuhalten und die richtigen Helfer zu wählen.

Die Welt der Linux-Distributionen ist nur schwer zu überschauen: Die Datenbank von Distrowatch [↗](#) kennt Hunderte von Distributionen mit unterschiedlichsten Ansätzen und Spezialisierungen. Die Vielfalt täuscht allerdings ein wenig: Nicht jede Distribution erfindet das Rad neu und beginnt bei null. Die meisten Projekte basieren auf einer der großen Distributionen wie Debian, Slackware oder Arch Linux. So entstehen beispielsweise Ubuntu, OpenSuse oder Manjaro. Über die GNU/Linux Distribution Timeline [↗](#) lässt sich die Entwicklungsgeschichte gut nachvollziehen.

Einer der großen Unterschiede zwischen den einzelnen Distributionssträngen liegt in der Paketverwaltung: Debian und dessen Derivate nutzen das DEB-Format sowie Dpkg und Apt als Werkzeuge. Von Red Hat stammt das RPM-Format, wo Dnf die Verwaltung übernimmt. Unter Arch Linux und dessen Ablegern kommen Binärpakete im Format `.pkg.tar.xz` zum Einsatz; das Frontend zur Paketverwaltung nennt sich Pacman.

## Open-Source-Universum

In der Regel kümmern sich die Entwickler einer Distribution um die Paketierung der Anwendungen und Bibliotheken in den Paketquellen. Die oft kleinen Teams beschränken sich dabei allerdings oft auf das Nötigste – was nicht heißt, dass die Auswahl dürftig ausfällt: Ubuntu 18.10 stellt in seinen Paketquellen über 60 000 Pakete bereit, Arch Linux beschränkt sich auf 10 500 [1](#). Die große Differenz ergibt sich aus der Art der Paketierung. Debian und Ubuntu teilen Software-Projekte gerne in deren einzelne Komponenten auf und verpacken sie auch so, wodurch sich die Zahl der Pakete massiv erhöht.

Die Open-Source-Welt dreht sich allerdings so schnell, dass die oft kleinen Entwicklerteams kaum mit dem Verpacken von Programmen und neuen Versionen hinterherkommen. Selbst die Paketquellen von Rolling-Release-Distributionen wie Arch Linux sind nicht immer auf dem neuesten Stand und enthalten auch nicht immer jede für Linux verfügbare

Software. Schließlich geht es nicht nur darum, den Quellcode zu bauen und in Pakete zu stecken, sondern auch darum, das Ergebnis im Kontext der Distribution auf Fehler zu testen. Letztendlich soll die angebotene Software auch funktionieren und die Sicherheit und Stabilität des Systems nicht gefährden.

## Ergänzt von der Community

Um unabhängigen Entwicklern und neuen Projekten eine Tür zu den Systemen der Anwender zu öffnen und so eine möglichst einfache Installation von nicht in den Paketquellen bereitgestellten Anwendungen zu ermöglichen, bieten die großen Distributionen in der Regel eine weniger intensiv betreute „Hintertür“ an. Bei Ubuntu etwa stehen Entwicklern und Anwendern die Personal Package Archives oder kurz PPAs zur Verfügung [↗](#). OpenSuse betreibt mit dem OpenSuse Build Service OBS [↗](#) sogar einen Dienst, der Pakete nicht nur für die eigene Distribution erstellt, sondern auch für Fedora, Debian, Ubuntu und andere.

Arch Linux bietet seiner Community mit dem Arch User Repository AUR [↗](#) ebenfalls einen Weg an, die offizielle Paketverwaltung zu umgehen, wählt aber im Vergleich zu PPAs oder dem OBS einen etwas anderen Ansatz. Statt binärer Pakete, die nur über die Paketverwaltung auf die Festplatte kopiert werden müssen, enthält das AUR lediglich Rezepte in Form von PKGBUILD-Dateien [↗](#). Wie bei einem Kochrezept instruiert diese Datei dann, woher die Software stammt, wie sie sich bauen lässt, welche Abhängigkeiten es dabei zu beachten gilt, und welches „Gewürz“ (ein Patch oder ein kleiner Workaround) das Gericht für Arch erst so richtig schmackhaft macht.

**Listing 1** zeigt als Beispiel die PKGBUILD-Datei von Yay [↗](#), des populärsten Programms im AUR. Die Variablen pkgname bis pkgdesc beschreiben Anwendung und Versionsnummer; arch gibt an, unter welchen Architekturen das Programm läuft. Die Variable url enthält einen Link zur Webseite des Projekts. Über depends und makedepends definiert das PKGBUILD-File die Abhängigkeiten –

Letztere dürfte man nach der Installation wieder löschen, da sie nur zum Bauen der Anwendung gebraucht werden.

Unter source findet sich der Link zum Quellcode oder einem bereits fertigen Binärpaket, entweder in einem anderen Paketformat (DEB, RPM) oder als stati-

```

toff@ontario:~$ neofetch
toff@ontario
-----
OS: Arch Linux x86_64
Kernel: 5.0.7-arch1-1-ARCH
Uptime: 8 hours, 8 mins
Packages: 1747 (pacman), 4 (flatpak)
Shell: bash 5.0.2
Resolution: 2560x1440, 1680x1050
DE: GNOME 3.32.0
Theme: vimix [GTK2/3]
Icons: Adwaita [GTK2/3]
Terminal: gnome-terminal
CPU: Intel i7-7700T (8) @ 2.900GHz
GPU: Intel HD Graphics 630
Memory: 7468MiB / 15894MiB

[19:04:59 toff ~]$ pacman -Slq | wc -l
10580
[19:05:03 toff ~]$

```

**1** Die Paketverwaltung von Arch Linux kennt „nur“ 10 580 Pakete, kombiniert aber viele Einträge.

### Listing 1

```

# Maintainer: Jguer <joaogg3@gmail.com>
pkgname=yay
pkgver=9.2.0
pkgrel=1
pkgdesc="Yet another yogurt. Pacman wrapper and AUR helper written in go."
arch=('i686' 'x86_64' 'armv7h' 'armv6h' 'aarch64')
url="https://github.com/Jguer/yay"
license=('GPL')
depends=(
    'pacman>=5.1'
    'sudo'
    'git'
)
makedepends=(
    'go'
)
source=("${pkgname}-${pkgver}.tar.gz::https://github.com/Jguer/yay/archive/v${pkgver}.tar.gz")
sha1sums=('760cd8667c68fbd0dbdd838490cbaaddc48bcc3')

build() {
    cd "$srcdir/${pkgname}-${pkgver}"
    make VERSION=${pkgver} DESTDIR="$pkgdir" PREFIX=/usr
}

package() {
    cd "$srcdir/${pkgname}-${pkgver}"
    make VERSION=${pkgver} DESTDIR="$pkgdir" PREFIX=/usr install
}

```

scher Build, etwa bei proprietären Anwendungen wie Googles Chrome-Browser. Die Prüfsumme unter sha1sums stellt sicher, dass beim Download alles mit rechten Dingen zugegangen ist (siehe Kasten [Trau, schau, wem](#)). In den Funktionen `build()` und `package()` stehen dann Instruktionen, wie das System die Software bauen und installieren soll. Im Beispiel beschränken sich diese auf Make-Kommandos, wie man Sie auch beim

klassischen Build-Dreischnitt aus `./configure && make && make install` verwenden würde.

## Do it yourself

Um nun Pakete aus dem AUR zu bauen, müssen Sie auf einem frisch installierten Arch-System das Git-Werkzeug sowie die Basis-Pakete zum Kompilieren einspielen ([Listing 2](#), erste Zeile). Anschließend suchen Sie sich aus der AUR-Homepage unter [aur.archlinux.org](http://aur.archlinux.org) das gewünschte Paket mit dem entsprechenden Paketnamen heraus. Wie in [Listing 2](#) gezeigt, laden Sie dann die PKGBUILD-Datei aus dem Git-Repository (Zeile 2) und lassen das Paket bauen (ab Zeile 4). Voraussetzung dafür ist jedoch, dass Sie zuvor von Hand dafür sorgen, die Abhängigkeiten zu erfüllen (im Beispiel den Go-Compiler in Form des Pakets `go-pie`).

Abbildung 2 zeigt den Vorgang im Terminal von Anfang bis Ende, in [Abbildung 3](#) sehen Sie das Ergebnis: In der Datei `yay-9.2.0.tar.gz` steckt der Quellcode des gerade gebauten Programms. Bei der Datei `yay-9.2.0-x86_64.pkg.tar.xz` handelt es sich um das finale Produkt, das Sie mit administrativen Rechten über das Kommando in Zeile 5 von [Listing 2](#) installieren. Einmal eingespielt, dürfen Sie den gesamten Ordner `yay/` löschen, oder Sie bewahren die `pkg.tar.xz`-Datei für spätere Neuinstallationen auf. Sie könnten die Datei auch auf andere Rechner übertragen oder dort das Programm einspielen, ohne es neu kompilieren zu müssen.

Auf den ersten Blick sieht das nach viel Arbeit aus. Alleine die Worte „aus dem Quellcode kompilieren“ und das händische Auflösen von Abhängigkeiten dürfte einige Anwender vor Arch und dem User Repository abschrecken. Doch die meisten Arch-Nutzer vermeiden die Handarbeit, indem sie auf ein Hilfswerkzeug in Form eines sogenannten AUR-Helfer zurückgreifen. Das Arch-Wiki listet eine ganze Reihe dieser Programme [auf](#). Sie haben allerdings eine bewegte Geschichte hinter sich, und nicht jeder AUR-Helfer ist zu empfehlen und vor allen Dingen noch aktuell.

```

tofff@ontario:~/AUR-Pakete/yay
[13:19:49 tofff AUR-Pakete]$ git clone https://aur.archlinux.org/yay.git
Klone nach 'yay' ...
remote: Enumerating objects: 244, done.
remote: Counting objects: 100% (244/244), done.
remote: Compressing objects: 100% (173/173), done.
remote: Total 244 (delta 69), reused 244 (delta 69)
Empfange Objekte: 100% (244/244), 50.42 KiB | 10.08 MiB/s, Fertig.
Lose Unterschiede auf: 100% (69/69), Fertig.
[13:19:54 tofff AUR-Pakete]$ cd yay
[13:19:57 tofff yay]$ makepkg
==> Erstelle Paket: yay 9.2.0-1 (Mo 15 Apr 2019 13:20:03 CEST)
==> Prüfe Laufzeit-Abhängigkeiten...
==> Prüfe Buildtime-Abhängigkeiten...
==> Empfange Quellen...
-> Lade yay-9.2.0.tar.gz herunter...
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
100  117      0  117    0     0    308      0  --:--:-- --:--:-- --:--:--    308
100  345k  100  345k    0     0   313k      0  0:00:01 0:00:01 --:--:-- 1090k
==> Überprüfe source Dateien mit sha1sums...
yay-9.2.0.tar.gz ... Durchgelaufen
==> Entpacke Quellen...
-> Entpacke yay-9.2.0.tar.gz mit bsdtar
==> Beginne build()...
go build -v -ldflags '-s -w -X main.version=9.2.0' -o yay -mod=vendor
github.com/jguer/go-alpm
github.com/Morganamilo/go-srcinfo
github.com/Morganamilo/go-pacmanconf/ini
github.com/mikkeloscar/aur
github.com/Morganamilo/go-pacmanconf
github.com/Jguer/yay/v9
==> Betrete fakeroot Umgebung...
==> Beginne package()...
install -Dm755 yay /home/tofff/AUR-Pakete/yay/pkg/yay/usr/bin/yay
install -Dm644 doc/yay.8 /home/tofff/AUR-Pakete/yay/pkg/yay/usr/share/man/man8/yay.8
install -Dm644 completions/bash /home/tofff/AUR-Pakete/yay/pkg/yay/usr/share/bash-completion/completions/yay
install -Dm644 completions/zsh /home/tofff/AUR-Pakete/yay/pkg/yay/usr/share/zsh/site-functions/_yay
install -Dm644 completions/fish /home/tofff/AUR-Pakete/yay/pkg/yay/usr/share/fish/vendor_completions.d/yay.fish
==> Säubere Installation...
-> Entferne libtool Dateien...
-> Bereinige ungewollte Dateien...
-> Entferne statische Bibliotheken...
-> Entferne unnötige Symbole aus Binär-Dateien und Bibliotheken...
-> Komprimiere Man-Pages und Info-Seiten...
==> Checking for packaging issues...
==> WARNUNG: Paket enthält einen Verweis auf $srcdir
usr/bin/yay
==> Erstelle Paket "yay"...
-> Erstelle .PKGINFO Datei...
-> Erstelle .BUILDINFO Datei...
-> Erstelle .MTREE-Datei...
-> Komprimiere Paket...
==> Verlasse fakeroot Umgebung.
==> Beendete Erstellung: yay 9.2.0-1 (Mo 15 Apr 2019 13:20:15 CEST)
[13:20:15 tofff yay]$

```

**2** Mithilfe von Kochrezepten in Form von PKGBUILD-Dateien macht es Arch Linux dem Anwender leicht, Programme aus dem Quellcode zu bauen.

### Listing 2

```

01 $ pacman -S git base-devel
02 $ git clone https://aur.archlinux.org/paketname.git
03 $ cd paketname
04 $ makepkg
05 # pacman -U paketname-*.pkg.tar.xz
06 ### Alternativ die zwei letzten Aktionen in einem Schritt
07 $ makepkg -sri

```

## Es war einmal

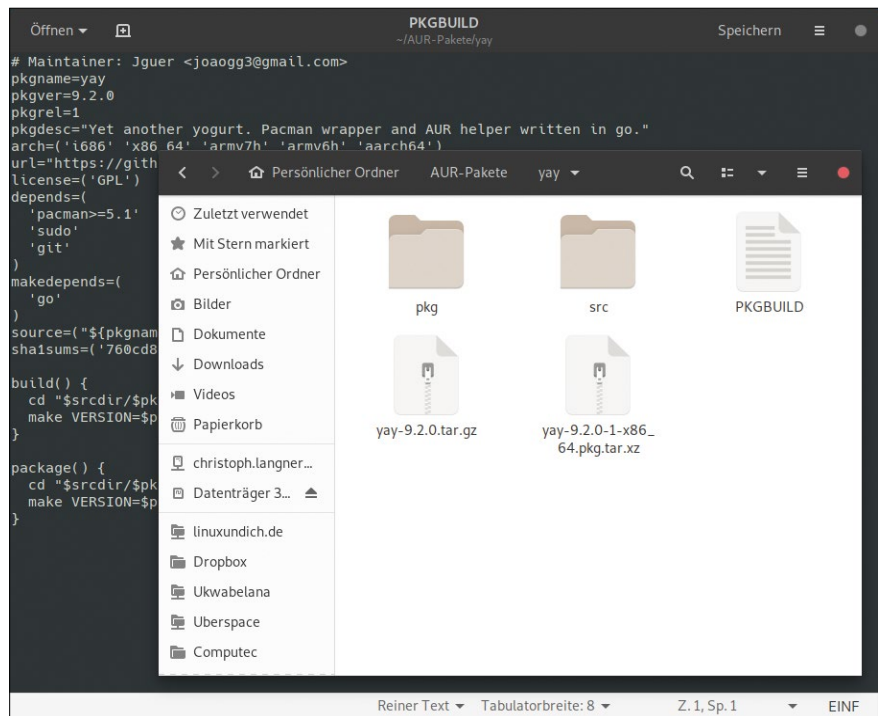
Wer vor ein paar Jahren zur Arch-Linux-Gemeinde stieß, der kennt mit Sicherheit den AUR-Helper Yaourt. Empfehlungen für das Programm fanden sich im Arch-Wiki, in vielen Dokumentationen und auch in Veröffentlichungen im Internet. Die Syntax von Yaourt gleicht jener des offiziellen Paketverwaltungswerkzeugs Pacman: Mit `yaourt -Ss Begriff` durchsucht das Programm zum Beispiel sowohl die offiziellen Paketquellen als auch das AUR; mit `yaourt -S Paket` erfolgt die Installation. Auf diese Art und Weise hat sich Yaourt als einfacher Ersatz für die Paketverwaltung etabliert.

Als AUR-Helper kümmert sich Yaourt nicht nur um den Build-Vorgang, sondern vor allen Dingen auch um das Auflösen der Abhängigkeiten. Das erleichtert insbesondere das Einrichten größerer Programme, die weitere Pakete aus dem AUR nach sich ziehen – speziell dann, wenn Abhängigkeiten weitere Abhängigkeiten benötigen. So lassen sich dann auch große Brocken mit nur einem kurzen Kommando aus dem AUR ins System einspielen.

Inzwischen ist das Programm allerdings deutlich in die Jahre gekommen. Der Entwickler selbst empfiehlt, das Projekt auf dem Dachboden in Rente zu schicken [🔗](#). Es gibt seit Jahren schon keine wirklichen Updates mehr, die Seite im Arch-Wiki zu den AUR-Helfern listet Yaourt nicht mehr auf. Zwischendurch wurde das Programm sogar selbst aus dem AUR genommen, nach kurzer Zeit jedoch wieder eingefügt. Schon alleine deshalb sollten Arch-Nutzer zu moderneren Alternativen greifen.

## Nachfolger Pacaur

Pacaur [🔗](#) galt bereits vor dem stotternen Ende von Yaourt als die bessere Alternative. Sein Vorteil liegt darin, alle die Installation betreffenden Abfragen in einem Rutsch zu stellen und erst dann, ohne weitere Zwischenstopps, die Pakete zu bauen und einzuspielen [4](#). Das erleichtert besonders bei vielen anstehenden Updates aus dem AUR oder bei



**3** Die PKGBUILD-Datei des AUR-Helpers Yay sowie das finale Paket im Format einer `pkg.tar.gz`-Datei. Diese können Sie auch auf andere Systeme übertragen.

der Installation größerer Anwendungen mit vielen Abhängigkeiten die Arbeit.

Wie so manches ambitionierte Open-Source-Projekt trudelte allerdings auch Pacaur in eine Krise. Schon 2017 erklärte der Hauptentwickler *Spyhawk*, dass der Code einen massiven Umbau benötige

## Trau, schau, wem

AUR-Helper haben unter Arch-Enthusiasten nicht den besten Ruf. Das Arch-Wiki warnt vor deren Einsatz und empfiehlt, „sich mit dem manuellen Build-Prozess vertraut zu machen, um aufkommende Probleme selbst zu beheben“. Die wichtigsten Gründe vergisst das Wiki allerdings in der Warnung: Die Hürden, eine PKGBUILD-Datei in das AUR-Repository zu bekommen, liegen sehr niedrig. Eine Qualitätskontrolle findet nur oberflächlich statt, einzig eine Gruppe an Trusted Users [🔗](#) überwacht die Vorgänge im AUR.

Mitte letzten Jahres gelang es Angreifern, eine (wenn auch harmlose) Malware ins AUR zu schmuggeln [🔗](#). Sie übernahmen dazu damals eine Reihe verwaister AUR-Einträge – was für jeden Nutzer möglich ist –

und ergänzten die Installationsroutine um eine Funktion, die eine Reihe von Daten (Datum, Zeit, CPU-ID, Pacman-Details und die Ausgaben etwa von `uname -a`) an einen Pastebin-Dienst übermittelte. Zu den manipulierten Paketen gehörte auch der bei vielen Nutzern beliebte Acrobat Reader.

Beim Installieren oder Aktualisieren von AUR-Paketen sollten Sie als Anwender daher immer einen Blick auf die PKGBUILD-Datei werfen und prüfen, ob der Quellcode aus dem offiziellen Repository des Projekts stammt, und ob Patches und Aktionen wirklich zur Software gehören. Für Rückfragen steht beispielsweise die Kommentarfunktion auf der AUR-Webseite oder die Mailingliste *aur-general* [🔗](#) zur Verfügung.

```

tofff@ontario:~$ pacaur -Syua
:: Synchronisiere Paketdatenbanken...
  core ist aktuell
  extra ist aktuell
  community          4,8 MiB  6,71M/s  00:01 [#####] 100%
  multilib           175,0 KiB  7,43M/s  00:00 [#####] 100%
:: Starte AUR-Upgrade...
:: ttf-ms-win10: Ignoriere Paket-Upgrade (10.0.17134.1-1 => 10.0.17763.348-1)
:: Löse Abhängigkeiten auf...
:: Suche nach Paketkonflikten...

AUR Packages (4) perl-cpanel-json-xs-4.11-3 perl-mojolicious-8.13-1 perl-test-cpan-meta-0.24-1 skypeforlinux-preview-bin-8.43.76.52-1
Repo Packages (37) asar-1.0.0-1 perl-b-keywords-1.20-1 perl-class-data-inheritable-0.08-8 perl-class-inspector-1.34-1 perl-class-tiny-1.006-1 perl-clone-0.41-1 perl-config-tiny-2.23-3 perl-devel-stacktrace-2.03-2 perl-exception-class-1.44-2 perl-exporter-tiny-1.002001-2 perl-file-find-rule-0.34-4 perl-file-find-rule-perl-1.15-1 perl-file-homedir-1.004-1 perl-file-sharedir-1.116-1 perl-inc-latest-0.500-4 perl-io-string-1.08-11 perl-lingua-en-inflect-1.903-1 perl-list-moreutils-0.428-2 perl-list-moreutils-xs-0.428-2 perl-module-build-0.4224-7 perl-module-pluggable-5.2-4 perl-number-compare-0.03-7 perl-params-util-1.07-9 perl-path-tiny-0.108-1 perl-perl-critic-1.132-1 perl-perl-minimumversion-1.38-1 perl-pod-spell-1.20-1 perl-ppi-1.236-1 perl-ppix-quotelike-0.006-1 perl-ppix-regexp-0.064-1 perl-ppix-utilities-1.001000-1 perl-readonly-2.05-3 perl-string-format-1.18-1 perl-task-weaken-1.06-1 perl-test-leaktrace-0.16-4 perl-text-glob-0.11-4 perl-tidy-20181120-1

Download-Größe der Repo-Pakete:  2.48 MiB
Installationsgröße der Repo-Pakete:  7.16 MiB

:: Mit Installation fortfahren? [Y/n] y
:: Hole Paket(e)...
clone complete: /home/tofff/.cache/pacaur/perl-test-cpan-meta
clone complete: /home/tofff/.cache/pacaur/perl-cpanel-json-xs
clone complete: /home/tofff/.cache/pacaur/perl-mojolicious
clone complete: /home/tofff/.cache/pacaur/skypeforlinux-preview-bin
:: PKGBUILD für perl-cpanel-json-xs betrachten? [Y/n] n
:: PKGBUILD für skypeforlinux-preview-bin betrachten? [Y/n] n
:: PKGBUILD für perl-mojolicious betrachten? [Y/n] n
:: PKGBUILD für perl-test-cpan-meta betrachten? [Y/n]

```

**4** Der AUR-Helper Pacaur galt als verwaist, wurde aber wiederbelebt. Seine Stärke liegt im Ablauf: Er stellt alle Fragen vor dem (oft zeitaufwendigen) Build-Prozess.

```

tofff@ontario:~$ yay -S android-studio
:: There are 2 providers available for android-studio:
:: Repository AUR
   1) android-studio 2) android-studio-dummy

Enter a number (default=1):
:: Checking for conflicts...
:: Checking for inner conflicts...
[Aur: 1] android-studio-3.3.2.0-1

:: Downloaded PKGBUILD (1/1): android-studio
   1 android-studio (Build Files Exist)
==> Diffs to show?
==> [N]one [A]ll [Ab]ort [I]nstalled [No]tInstalled or (1 2 3, 1-3, ^4)
==>
:: Parsing SRCINFO (1/1): android-studio
==> Erstelle Paket: android-studio 3.3.2.0-1 (Di 16 Apr 2019 15:51:53 CEST)
==> Empfange Quellen...
-> Lade android-studio-ide-182.5314842-linux.zip herunter...
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
   Dload  Upload   Total   Spent    Left   Speed
  9 1014M    9 101M    0     0  6756k      0  0:02:33  0:00:15  0:02:18 6921k

```

**5** Ebenso wie Pacaur stellt der AUR-Helper Yay sämtliche Fragen vor dem Kompilieren. Besonders Programme mit vielen Abhängigkeiten, die wiederum aus dem AUR zu installieren sind, lassen sich so im Handumdrehen bauen.

und dass die Arbeit an dem Programm für ihn einen inzwischen ungeliebten und vor allem auch unbezahlten Job darstelle. Als Konsequenz zog er die Notbremse und stellte weitere Entwicklungen am Programm ein. Wie schon Yaourt verschwand Pacaur auf Antrag des Entwicklers auch aus dem AUR.

Inzwischen steckt allerdings wieder frisches Leben im Projekt. Mit dem neuen Hauptentwickler *E5ten* am Steuer trudelt seit Anfang 2019 wieder regelmäßig Commits im (geforkten) Github-Repository der Anwendung ein.

Auch das Arch-Wiki führt Pacaur nicht mehr als hoffnungslos, weil aufgegebenen, Fall. Im Gegenteil: Bei allen wichtigen Kriterien signalisiert ein grün hinterlegtes Yes, dass Pacaur wieder auf der Höhe der Zeit angekommen ist.


## Der Neue: Yay

So schließt sich der Kreis zum bereits in der Einleitung des Artikels angesprochenen Programm Yay. Der AUR-Helper gilt in der Community aktuell als sehr zuverlässig und gut gewartet. Zudem erfüllt das Programm alle Anforderungen an einen modernen AUR-Helfer **5**.

So stellt Yay zum Beispiel vor dem Build-Prozess alle relevanten Fragen, bietet eine Möglichkeit die PKGBUILD-Datei zu bearbeiten, und löscht bei Bedarf nach Abschluss automatisch alle Pakete, die nur zum Bauen der Anwendung geeignet haben. Als Pacman-Wrapper bindet Yay zudem Pacman selbst mit ein, sodass der Aufruf von yay -Syu (oder noch einfacher yay) das komplette Arch-System auf den aktuellen Stand bringt.


Einzig der Einsatz der Programmiersprache Go mag sich auf Systemen mit knapp bemessenem Festplattenplatz als Nachteil erweisen. Die Sprache belegt knapp 500 MByte auf dem Massenspeicher und wird von nur wenigen anderen Programmen benötigt. Geht bei Ihrem System der Platz zur Neige, hilft es, nach dem Build das Paket go wieder zu deinstallieren – bei einem Update von Yay müssen Sie es dann allerdings wieder neu aus dem Netz laden, um die Aktualisierung übersetzen zu können.


## Grafischer AUR-Helfer

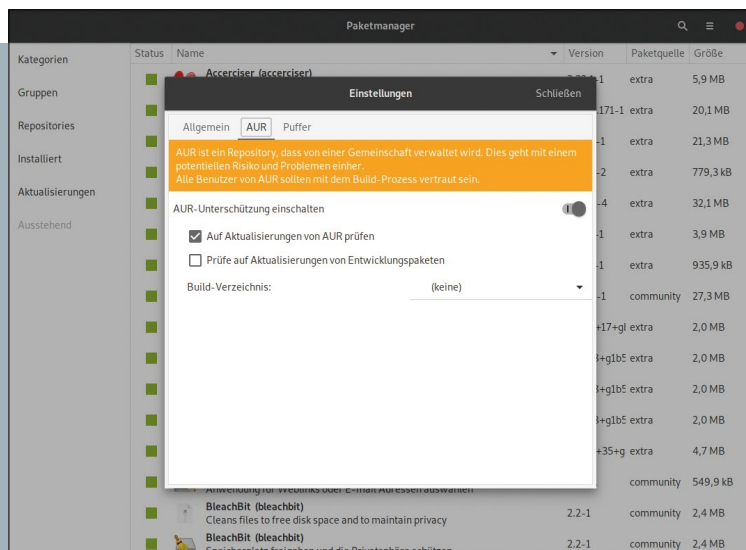
Um dem Anwender eine benutzerfreundliche Lösung für die Paketverwaltung an die Hand zu geben, entwickelt das Arch-Derivat Manjaro mit Pacman  einen grafischen Pacman-Aufsatz, der auch das AUR integriert. Manjaro spielt Pacman von Haus aus während der Installation mit auf die Festplatte, unter Arch Linux installieren Sie das Programm über das gleichnamige PKGBUILD aus dem AUR. In beiden Fällen müssen Sie die AUR-Unterstützung in der Anwendung noch in den Einstellungen aktivieren. Eine auffällige Warnung weist dabei auf die potenziellen Gefahren des AUR hin [6](#).

Pacman zeigt danach in der Seitenleiste mit *Repositories* und *AUR* zum Suchbegriff passende Treffer aus den jeweiligen Quellen an. Zu installierende AUR-Pakete wählen Sie über Kontrollkästchen in der Spalte *Status* an und starten mit einem Klick auf *Übernehmen* am unteren Rand der Anwendung deren Einrichtung [7](#). Pacman zeigt daraufhin die zu installierenden Pakete samt Quelle an. Etwas versteckt in der Kopfleiste findet sich mit *Build-Dateien bearbeiten* die Möglichkeit, die PKGBUILD-Dateien einzusehen und bei Bedarf anzupassen.

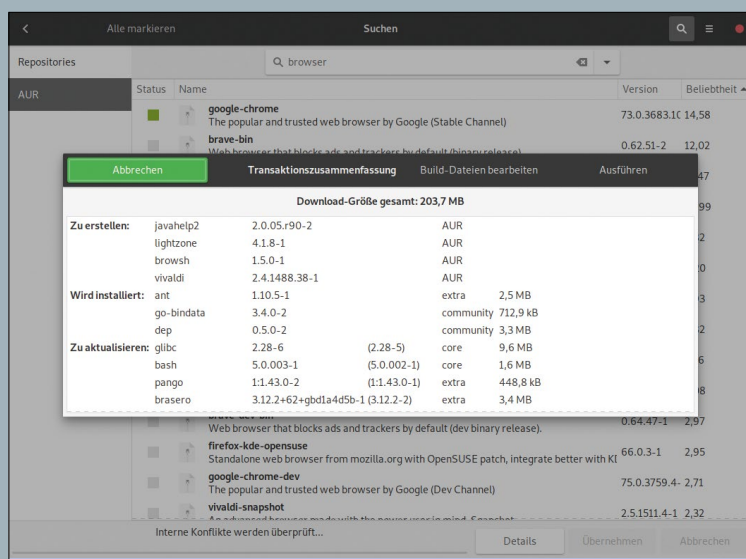
## Fazit

Das Geschehen rund um die AUR-Helfer demonstriert den Charakter von Arch Linux als Rolling-Release-Distribution, die sich stetig verändert. Anwender, die sich nur wenig informieren, nutzen noch heute Yaourt, obwohl das Programm seit Jahren als veraltet und verwaist gilt. Da es sich beim AUR zudem nicht um einen offiziellen Bestandteil der Distribution handelt, gibt es von offizieller Seite – etwa über die Arch Linux News  – keine Informationen zu Änderungen oder gar Gefahren. Als Anwender müssen Sie also selbst die Augen offenhalten.

Benötigen Sie nur wenige Anwendungen aus dem AUR, bauen Sie die Pakete am besten von Hand. Sind erst einmal alle Abhängigkeiten erfüllt, beschränkt sich der Build-Prozess auf zwei Kommandos. Verwenden Sie einen AUR-Helfer, sollten Sie grundsätzlich die PKGBUILD-Datei kontrollieren: Nur so vermeiden Sie böse Überraschungen und behalten die Kontrolle über das System. (cla/jlu) 



[6](#) Der im Rahmen des Arch-Derivats Manjaro entwickelte Paketmanager Pacman dient als Frontend für Pacman. In den Einstellungen lässt sich das AUR als zusätzliche Paketquelle aktivieren.



[7](#) Pacman informiert Sie bei der Installation darüber, woher ein Paket stammt. Achten Sie darauf, über einen Klick auf den Schalter *Build-Dateien bearbeiten* in der Fensterleiste die PKGBUILD-Dateien zu überprüfen.



Weitere Infos und  
interessante Links

[www.linux-user.de/qr/42865](http://www.linux-user.de/qr/42865)

# Neues auf den Heft-DVDs

## Durchdacht und ausgereift: Fedora Workstation

Als Desktop nutzt **Fedora 30 Workstation** standardmäßig **Gnome 3.32**. Zur Software-Grundausstattung gehören unter anderem **GCC 9**, die **Bash 5.0** und **PHP 7.3**. **Fedora 30** verwendet zur Interprozesskommunikation jetzt den **Dbus-Broker**, der unter anderem schneller arbeiten soll. Der **Linux-Kernel** wer-

kelt im Hintergrund in **Version 5.0**. Mehr zu den Neuerungen von **Fedora 30** lesen Sie im großen **Distributionsvergleich** ab **Seite 20** in dieser Ausgabe. Sie starten das System von **Seite A** der ersten **DVD**, das zugehörige **ISO-Image** finden Sie unter **isos/**.

## Ubuntu „Disco Dingo“ ist da

Beim aktuellen **Ubuntu 19.04** handelt es sich um ein **Zwischen-Release**, das vor allem **Paketaktualisierungen**, **Fehlerbehebungen** und einige **kosmetische Korrekturen** gegenüber dem Vorgänger mitbringt. So kommt jetzt etwa **Gnome 3.32** zum Einsatz, den **Linux-Kernel** hoben die Entwickler

von **4.18** auf **5.0**. Sie starten die **Distribution** von **Seite A** der ersten **DVD**. Das zugehörige **ISO-Image** finden Sie unter **isos/**, einen eigenen Artikel lesen Sie ab **Seite 10**. Wie sich der „**Disco Dingo**“ gegen **Fedora** und **OpenSuse** schlägt, zeigt der **Vergleich** ab **Seite 20**.

## Klein, handlich, übersichtlich: EasyOS

Minimaler **Ressourcenbedarf**, gepaart mit einfachem **Handling** – das macht das von **Puppy Linux** abgeleitete **EasyOS 1.0** vor allem aus. Dabei enthält es jedoch mehr **Software** als **Puppy** und bietet außerdem **Zu-**

satzfunktionen, wie das **Betreiben** von **Applikationen** in **Containern**. Mehr dazu lesen Sie in einem Artikel ab **Seite 30**. Sie booten die **Distribution** von **Seite A** der ersten **DVD**.

## Containerbasiert: Clear Linux OS

Bei **Clear Linux OS r29610** des Halbleiterherstellers **Intel** handelt es sich um ein **Hybrid-Betriebssystem**, das ursprünglich hauptsächlich für den Einsatz mit **Containern**, in der **Cloud** oder dem **Internet** der Dinge ausgelegt ist. Dank einiger zusätz-

licher **Komponenten** macht es jedoch zunehmend auch auf dem **Desktop** eine gute Figur. Sie booten das **Intel-System** von **Seite B** des ersten **Datenträgers**. Mehr zu dieser ungewöhnlichen **Distribution** lesen Sie im Artikel ab **Seite 36**.

## Neue Wege gehen: Solus 4.0 Budgie

Zahlreiche **Linux-Distributionen** basieren auf **Ubuntu** oder **Debian** und nutzen in diesem Fall deren spezifische **Werkzeuge**. Wenn Sie **ausgetretene Pfade** verlassen und trotzdem mit einem **ausgereiften Betriebssystem** arbeiten möchten, empfiehlt sich

ein Blick auf das unabhängig entwickelte **Solus 4.0 Budgie**. Die **Stärken** und **Schwächen** der **Distribution** beschreibt ein Artikel ab **Seite 42**. Sie finden das **ISO-Image** auf **Seite B** der zweiten **DVD** im Verzeichnis **isos/**.



## Neues aus Nürnberg: OpenSuse Leap 15.1

Zu den zahlreichen Verbesserungen von OpenSuse Leap 15.1 zählen ein Update des Grafikstacks, für das die Entwickler Teile von Kernel 4.19 auf Kernel 4.12 zurückportiert haben, sowie der verbesserte Support für AMDs Vega-Chips. Daneben überarbeiteten die Nürnberger den Installer YaST. Auf Seite A der zweiten Heft-DVD finden Sie die installierbare Vollversion. Die Rückseite enthält die Live-Ableger KDE, Gnome und Rescue. Aus produktionstechnischen Gründen bootet nur die KDE-Variante, die anderen finden Sie als ISO-Image im Verzeichnis `isos/`. Mehr zu den Neuerungen von Leap 15.1 lesen Sie im Artikel ab Seite 20. (tle) ■



Bei der DVD-Edition klebt an dieser Stelle der zweite Heft-Datenträger. Bitte wenden Sie sich bei Reklamationen wegen fehlender oder defekter Medien unter Angabe Ihrer Postanschrift per E-Mail an [computec@dpv.de](mailto:computec@dpv.de).

## Neue Programme

Mithilfe der Software **Cert 0.13.0** überprüfen Sie bei Bedarf, ob für eine URL ein gültiges SSL-Zertifikat existiert oder nicht. Das Tool lässt sich intuitiv bedienen und gibt alle Informationen strukturiert und übersichtlich aus. Über ein Template passen Sie bei Bedarf die Ausgabe an Ihre Wünsche an. → S. 8

Möchten Sie Ihren Nameserver auf Herz und Nieren testen, sollten Sie sich **Dnstrace 0.1.1** ansehen, mit dem Sie DNS-Anfragen simulieren und verfolgen. Auf diese Weise überprüfen Sie beispielsweise die Arbeitsgeschwindigkeit eines DNS-Servers. Dabei verfolgt das Tool zusätzlich CNAME-Einträge. → S. 8

Mit **Highlight 3.51** bereiten Sie den Quellcode von Programmen, Dokumenten und Webseiten optisch auf. Das Tool unterstützt alle gängigen Programmiersprachen und konvertiert den Quellcode bei Bedarf nach HTML, LaTeX oder RTF. → S. 8

Bei **Lsd 0.15.1** handelt es sich um eine Erweiterung des Befehls `ls`, die die Ausgabe unterschiedlich einfärbt und damit übersichtlicher gestaltet. Die Software ist stark vom Super-Colorls-Projekt inspiriert, aber in Rust statt wie das Original in Ruby geschrieben.

Mithilfe von **Lutris 0.5.12** installieren und verwalten Sie Spiele unter einer einheitlichen Benutzeroberfläche. Das Tool unterstützt neben nativen Linux-Spielen auch Windows-Software via Wine sowie Konsolen- und DOS-Klassiker über Emulatoren. → S. 48

Dass maschinelles Lernen einen immer größeren Einfluss in der Software gewinnt, zeichnete sich seit Langem ab. Trotzdem überrascht, dass das Konzept sogar bei Shells eine wichtige Rolle zu spielen beginnt. So erweitert **McFly 0.3.4** die History-Funktionen der Shell, sodass Sie schneller die gewünschten Einträge finden. → S. 82

Sogenannte Zeitwaagen zum Prüfen mechanischer Uhrwerke identifizieren Gang- und Lagefehler anhand akustischer Merkmale. Sie kosten mehrere Hundert Euro. Ein Linux-Notebook und das Programm **Timegrapher 0.5.0** ersparen Ihnen diese Ausgabe. → S. 58

Der **Tor-Browser 8.5** bietet eine einfache Möglichkeit, anonym zu surfen. Dazu verbindet er sich automatisch mit dem Tor-Netzwerk. Die aktuelle Version basiert auf Firefox 60.7 ESR. Mit dieser Version veröffentlichte das Projekt das erste stabile Android-Release, das Sie über Googles Play Store beziehen.