



Fedora 29 gibt Ausblick auf RHEL 8

Modular

Michael Plura

Fedora 29 dürfte die Basis für das kommende Red Hat Enterprise Linux 8 sein. Dank Modularity können Systemverwalter unterschiedliche Paketversionen parallel nutzen.

Fast pünktlich zum 15. Geburtstag – „Fedora Core 1“ erschien im November 2003 – veröffentlichte das Fedora-Projekt Release 29 seiner Linux-Distribution. Im September 2003 hatte Red Hat die Consumer-Version seiner Distribution mit dem Fedora Linux Project verschmolzen. Seitdem gilt Fedora als Consumer-Variante von Red Hats Enterprise Linux (RHEL), in der Entwickler neue Funktionen für das kommende RHEL testen. Fedora umfasst drei Editionen, die alle gemeinsame Basispakete nutzen: Fedora Workstation für den Desktop-Einsatz, Fedora Atomic Host für Cloud- und Containerinstanzen sowie eine Servervariante.

Letztere ist für Systemverwalter interessant, die RHEL einsetzen, weil sie einen Ausblick auf die zukünftige Version gibt und so Anhaltspunkte für eine langfristige Planung liefert. Für den produktiven Einsatz ist der Fedora-Server weniger

geeignet – und auch nicht gedacht –, da nicht jede neue Software automatisch problemlos und stabil läuft. Kleine Probleme tauchten im Test etwa beim Upgrade von Fedora-28-Systemen auf. Einmal gingen die deutschen Umlaute verloren und auf einer Workstation liefen einige Programme aus zusätzlichen Quellen (RPMfusion) zunächst nicht. Die Fehler behob das Projekt wie üblich innerhalb weniger Tage. Bekanntermaßen sollte man Fedora-Updates etwa zwei Wochen nach Erscheinungen vornehmen.

Einblick in die Neuerungen der Version 29

Es gibt viele kleine Aktualisierungen, wie einen Linux-Kernel 4.18 mit Patches aus dem aktuellen 4.19-Zweig. Die binutils sind als Version 2.31 enthalten, die glibc in Version 2.28. Der gcc-Compiler ist aus

dem buildroot des Buid-Systems (Koji und mock) geflogen. Der Grund: Moderne Software wird laut Fedora-Wiki immer weniger in C/C++, sondern in Python, Ruby, Node.js, Go, Rust und anderen aktuellen Programmiersprachen geschrieben, sodass ein klassischer C/C++-Compiler beim Erzeugen von Paketen nicht immer erforderlich ist, was beim Build-Prozess Zeit erspart.

Python aktualisierten die Entwickler auf Version 3.7, Perl auf 5.28 und Golang auf 1.11. Node.js liegt in Version 10 bei, die MySQL-Community-Edition machte einen Sprung von 5.7 auf 8. Die alte i686-Architektur setzt nun SSE2 voraus. Von den 64-Bit-PowerPCs (ppc64) verabschiedete sich das Projekt. Dafür erhielt der ARM-Port (ARMv7 und aarch64) Funktionen zum Einsatz von ZRAM (Virtual Swap Compressed in RAM, auch zSwap oder früher compcache). Komprimiertes RAM als Swap soll den Verschleiß von SD-Karten verringern.

Starre Paketverwaltung

Die Paketverwaltung von Linux ist recht starr. Der Maintainer eines Paketes entscheidet, welche Version aktuell und damit verfügbar ist. Um Abhängigkeiten nicht zu brechen und eine gewisse Stabilität im Sinne von Langlebigkeit einer Installation zu gewährleisten, sind die Pakete in Distributionen wie RHEL, SLES oder Debian oft recht alt. Den immer neuesten Stand bieten die „Cutting-Edge“-Varianten wie Gentoo, Arch Linux oder Fedora Rawhide. Die verwendete Distribution aber von der benötigten Softwareversion einer Anwendung abhängig zu machen, ist praxisfremd. Die Software in der passenden Version aus den Quellen selbst zu übersetzen und Abhängigkeiten von Hand aufzudröseln, bleibt erfahrenen Systemverwaltern vorbehalten.

Ansätze, diese Problematik elegant zu lösen, sind beispielsweise Container (Docker, LXC, OpenVZ), Appliance und Canonicals SnapApps/Snappy. Sie enthalten alle benötigten Dateien und Abhängigkeiten für eine Anwendung in einem Verzeichnis und lassen sich parallel zur Paketverwaltung installieren. Red Hat antwortete 2015 darauf mit den etwas komplizierteren Flatpacs. Während Snap-App grundsätzlich auf die neuesten Versionen abzielt und alle Abhängigkeiten (im Zweifel einen kompletten GNOME-Desktop) mitbringt, befüllen die Maintainer Flatpacs nur teilweise und können so unterschiedliche Versionen ausrollen. Allerdings erfordert die Pflege beider

Systeme große Sorgfalt, da eine durch einen Fehler gefährdete Bibliothek zwar vielleicht im Basissystem aktualisiert wurde, der Maintainer der SnapApp oder des Flatpacks aber unter Umständen keine Zeit, Lust oder Ahnung hatte, auch seine Version zu korrigieren.

Ein neues Konzept: Modulare Repositories

Mit dem neuen Modulkonzept Modularity soll Fedora näher am System arbeiten und Softwarepakete unabhängig von deren Version verfügbar machen. Das in Fedora 28 für den Server-Spin eingeführte Verfahren gibt es nun für alle Fedora-Varianten (Server, Workstation und Atomic Host). Das hauseigene Fedora-Magazin erklärt es in der Ankündigung zu Fedora 29 recht praxisnah: Mit Modularity kann die Paketverwaltung einer Fedora-Basis unterschiedliche Paketversionen einer Software ausliefern (siehe ix.de/ix1812064).

Beispielsweise lässt sich Node.js sowohl in Version 8 als auch in Version 10 installieren, die Abbildung oben zeigt das für das aktuelle Fedora 29. Auf dieselbe Weise kann man bei Kubernetes entweder die neueste Version für den Stand-alone-Betrieb oder die zu OpenShift Origin passende, ältere Version einspielen. Modularity ist somit ein weiterer Versuch, das starre Konzept der Linux-Paketverwaltungen aufzubrechen.

In der Praxis sieht das so aus: `dnf module list` zeigt alle verfügbaren Module. Das gewünschte installiert man per `dnf module install nodejs:8/default`

Eventuell vorhandene Updates pflegt der Aufruf `dnf modules update` explizit ein, beim Upgrade wird etwas ungeschickt einfach eine andere Version über den Vorgänger „gebügelt“: `dnf module install nodejs:9/default`. Mehr Informationen zum noch in der Entwicklung befindlichen Modularity gibt es auf Fedora Docs (siehe ix.de/ix1812064).

Gewöhnungsbedürftige Installation

Red Hats Anaconda gebührt auch nach einigen Jahren Entwicklung nach wie vor der Preis für den mit Abstand unergonomischsten und umständlichsten Installer. Die Skalierung schlägt unter VirtualBox teilweise fehl, überflüssige Scroll-Effekte zwischen den einzelnen Seiten des Installers zerschneiden die Darstellung. Generell vergehen nach einer Benutzereingabe

**Dank Modularity
kann Fedoras
Paketverwaltung
parallel für un-
terschiedliche
Profile diverse
Paketversionen
bereithalten.**

```

root@dhcp167:~# dnf module list nodejs
Letzte Prüfung auf abgelaufene Metadaten: vor 0:10:16 am Di 06 Nov 2018 14:36:59 CET.
Fedora Modular 29 - x86_64
Name Stream Profiles Summary
nodejs 10 development, minimal, default Javascript runtime
nodejs 8 development, minimal, default [d] Javascript runtime

Fedora Modular 29 - x86_64 - Updates
Name Stream Profiles Summary
nodejs 10 development, minimal, default Javascript runtime
nodejs 8 development, minimal, default [d] Javascript runtime

Hint: [d]efault, [e]nabled, [x]disabled, [i]nstalled
root@dhcp167 ~#

```

(Mausklick) etliche Sekunden, bis Anaconda eine visuelle Rückmeldung gibt.

Weiter fehlen der umfangreichen Auswahl an Basisumgebungen (beispielsweise Fedora Betriebssystem, Minimalinstallation, Fedora Server Edition, Fedora Cloud Server et cetera) sowie den jeweils darauf aufbauenden Zusatzpaketen genaue Beschreibungen, was dort eigentlich installiert wird. Für einen Server starten Systemverwalter daher oft mit einer nackten Minimalinstallation, bestehend aus 315 RPMs, und installieren benötigte Software von Hand per `dnf`. Im laufenden System lässt sich mit `dnf groups info "Minimal Install"` (hier ist die englische Bezeichnung nötig) erkunden, welche RPM-Pakete und Paketgruppen zur jeweiligen Basisumgebung gehören. Der Aufruf `dnf grouplist installed [hidden]` listet alle aktuell [verdeckt] installierten Gruppen.

Spielweise für Entdecker

Seit Fedora 23 wird das System bei der Server Edition automatisch mit dem Web-GUI Cockpit gestartet, das Einsteigern die Administration erleichtern soll (siehe Aufmacher-Screenshot). Das grafische Frontend zur Systemverwaltung kommuniziert über `cockpit-bridge` mit den System-APIs, also `systemd`, dem `NetworkManager` und anderen Diensten, aber auch mit dem Kernel oder Docker. Nach der Installation kann sich der Systemverwalter über `https://<Server-IP>:9090/` und das bei der Installation festgelegte Passwort einloggen – einen modernen Browser oder ein Smartphone vorausgesetzt.

Fedora-Workstation mit nur einem Betriebssystem startet per Default ohne GRUB-Menü. Die oft allzu technischen Startmeldungen ersetzte man bereits vor einiger Zeit durch einen schicken Plymouth-Startbildschirm. Durch weitere Änderungen am Kernel, an GRUB2, SHIM und Plymouth startet Fedora 29 Workstation nun flickerfrei, schaltet also nur noch einmal vom Text- in den Grafikmodus um.

Eine ausführliche Beschreibung und ein Video, das die Vorteile zeigt, befinden sich auf der Projektseite. Für Intel-Grafikkarten gibt es dort auch eventuelle „Kernel-Tweaks“, auf NVIDIA- oder AMD/ATI-Karten funktioniert das noch nicht (siehe ix.de/ix1812064).

Eine Option, das GRUB-Menü beim nächsten Systemstart anzuzeigen, erhält man mit gedrückter Alt-Taste auf dem gdm-Anmeldeschirm nach Wechseln der Option „Restart“ zu „Boot Options“. Beim Start kann man alternativ die Shift-Taste drücken. Ein `grub-set-bootflag menu_show_once` schaltet die GRUB-Anzeige permanent ein. Der Fedora-29-Server startet bislang noch mit GRUB-Menü.

Vorgeschmack auf RHEL 8

Fedora 29 dürfte Systemverwaltern bereits einen Vorgeschmack auf RHEL 8 liefern, dessen Alphaversion Red Hat vor Kurzem an ausgewählte Kunden verteilte. Die Beta von RHEL 7 (12/2013) und die spätere Release basierten auf Fedora 19 (07/2013) und Fedora 20 (12/2013). Erscheint also die Beta von RHEL 8 vor Mai 2019 (dem geplanten Release-Termin von Fedora 30), dürfte RHEL 8 auf Fedora 29 mit den bis dato verfügbaren Erweiterungen basieren.

So weist die Roadmap für Version 30 darauf hin, dass Python 2.7 das EOL erreichen wird, die glibc die Version 2.29 erreichen soll und der Linux-Kernel 4.21 oder 4.22 aktuell sein wird (siehe ix.de/ix1812064). Langfristige IT-Projekte, die auf RHEL 8 aufbauen sollen, lassen sich mit Fedora 29 schon jetzt zumindest teilweise validieren. (avr@ix.de)

Michael Plura

arbeitet in Schweden als freier Autor mit den Schwerpunkten IT-Sicherheit, Virtualisierung und freie Betriebssysteme.

Alle Links: ix.de/ix1812064