



cron

Autor: JC PollmanChristian Richter (*jpollman@bigfoot.com**crichter@users.sourceforge.net*)

Layout: Matthias Hagedorn (*matthias.hagedorn@selflinux.org*)

Lizenz: GPL

Linux bietet zwei verwandte Programme für das Automatisieren von Aufgaben: **cron** und **at**. Beide starten beim Booten und laufen als Daemons, so werden sie niemals beendet.

Cron ist für sich wiederholende Aufgaben zuständig, **at** für einmalig ablaufende.

Inhaltsverzeichnis

1 cron

1 cron

Linux bietet etliche Programme für das Automatisieren von Aufgaben. Ein Beispiel ist **cron**. Beim Booten wird ein **cron** Daemon gestartet und im Normalbetrieb nicht beendet. **Cron** ist für sich wiederholende Aufgaben zuständig, die automatisch zu bestimmten Zeiten stattfinden sollen.

Cron liest die **crontab**-Datei, um seine Informationen zu erhalten. Das System und jeder User haben ihre eigenen **crontab**-Dateien. Die des Systems befindet sich in **/etc/crontab**. Verändern Sie diese bloß nicht! Selbst der Benutzer root sollte seine eigene Datei erzeugen.

Erzeugen der root **crontab**-Datei: Als erstes definieren Sie die **EDITOR**-Variable. Wahrscheinlich ist es das beste, die Einstellungen in **/etc/profile** zu machen, damit sie für alle Benutzer beim Anmelden gelten. Öffnen Sie die **/etc/profile** und fügen Sie diese beiden Zeilen hinzu:

/etc/profile
<pre>EDITOR=vi export EDITOR</pre>

Wenn Sie einen anderen Editor als **vi** bevorzugen, ändern Sie die Zeilen entsprechend. Sie müssen sich aus- und wieder einloggen, um die Änderung wirksam zu machen. Dann tippen Sie

```
user@linux ~/ # crontab /etc/crontab
```

ein. Das Kommando wird eine **crontab**-Datei für Sie erzeugen, die eine Kopie der System **crontab**-Datei ist. Editieren Sie nun ihre Datei mit:

```
user@linux ~/ # crontab -e
```

Beachten Sie, daß **crontab** sowohl der Name der Datei, als auch der Name des ausführbaren Programmes ist, ähnlich wie **passwd**. Die erzeugte Datei wird in etwa wie diese Konfiguration aussehen (dies ist eine RedHat-Datei):

Konfiguration RedHat-Datei
<pre>SHELL=/bin/bash PATH=/sbin:/bin:/usr/sbin:/usr/bin MAILTO=root HOME=/ #run-parts 01 * * * * root run-parts /etc/cron.hourly 01 4 * * * root run-parts /etc/cron.daily 22 4 * * 0 root run-parts /etc/cron.weekly 42 4 1 * * root run-parts /etc/cron.monthly</pre>

Löschen Sie alles nach der **HOME=/** Zeile, da das nicht die Aktionen sind, die Sie ausführen wollen. Ändern Sie außerdem den Pfad so, daß er in das Verzeichnis zeigt, in dem Sie Ihre Skripte speichern. Jede Zeile im **crontab** führt ein eigenes Programm aus. Die Zeilen haben ein spezielles Format:

Fünf Zeitfelder, gefolgt von dem auszuführenden Programm. Beachten Sie: Im System-**crontab** ist ein weiteres Feld, das **cron** anweist, das Programm als spezieller User auszuführen (z. B. root). In einem User-**crontab** wird dieses

Feld ignoriert. Die fünf Zeitfelder sind:

Minuten Stunden Tag-des-Monats Monat Wochentag

Auszug aus der Manual-Page

Die Zeit- und Datumsfelder sind:

Minute 0-59
Stunde 0-23
Tag-des-Monats 1-31
Monat 1-12 (oder Namen, siehe unten)
Wochentag 0-7 (0 oder 7 ist Sonntag oder Namen)

Ein Feld kann ein Stern (*) sein, was immer für "Erster-Letzter" steht.

Zahlenbereiche sind erlaubt. Bereiche sind zwei Zahlen, getrennt durch einen Bindestrich. Die angegebenen Grenzen sind inklusive.
Beispielsweise: 8-11 in "Stunde" bewirkt die Ausführung um 8, 9, 10, 11 Uhr.

Listen sind erlaubt. Eine Liste ist eine Menge von Nummern (oder Bereichen), getrennt durch Kommata.
Beispiele: "1,2,5,9" , "0-4,8-2". [Keine Hochkommata, Anmerkung des Übersetzers]

Schrittweiten können in Verbindung mit Bereichen genutzt werden. Hinter einem Bereich mit "/<Schrittweite>" angegeben, bestimmt die Schrittweite, ob Werte innerhalb des Bereiches übersprungen werden.
Beispiel: "0-23/2" kann unter Stunden benutzt werden, um ein spezielles Kommando alle zwei Stunden auszuführen. Die Alternative wäre: "0,2,4,6,8,10,12,14,16,18,20,22". Schrittweiten sind auch nach Sternen (*) erlaubt, "alle zwei Stunden" läßt sich auch durch "*/2" beschreiben.

Namen können für "Monat" und "Wochentag" benutzt werden. Benutzen Sie die ersten drei Buchstaben des entsprechenden Tages oder Monats (Groß-/Kleinschreibung wird nicht beachtet). Bereiche oder Listen sind mit Namen nicht erlaubt.

So. Hier nun ein Beispiel:

Wenn wir ein Backup-Skript haben, das um 5 Minuten nach 1 Uhr morgens jeden Tag laufen soll, würde die zugehörige **crontab**-Datei folgendermaßen aussehen:

Backup-Skript

```
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin:/usr/local/bin
MAILTO=root
HOME=/

5 1 * * * /usr/local/bin/run-backup
```

Dies teilt die Beendigung der Aufgabe dem Nutzer "root" mit. Wenn Sie mit dem Ablauf des Skriptes run-backup zufrieden und der Benachrichtigungen überdrüssig sind, ändern Sie die **MAILTO**-Zeile folgendermaßen:

run-backup
MAILTO= " "

Weitere Informationen finden sich in folgenden Manual-Seiten:

- * man crontab
- * man 5 crontab
- * man cron