

**Berufskolleg-Olsberg****Informationstechnische  
Assistenten****Betriebssystem – LinuX***Jobsteuerung  
mit Crontab***Crontab**

Der **cron**-Daemon ist eine Jobsteuerung von Unix bzw. Unix-artigen Betriebssystemen wie [Linux](#) oder Mac OS X, die wiederkehrende Aufgaben (*cronjobs*) zu einer bestimmten Zeit ausführen kann. Die auszuführenden Befehle werden in einer [benutzereigenen](#) Tabelle, der sogenannten *crontab*, gespeichert.

Diese Tabelle besteht aus sechs Spalten; die ersten fünf dienen der Zeitangabe (Minute, Stunde, Tag, Monat, Wochentag), alle weiteren Zeichen bis zum Zeilenumbruch werden als der auszuführende Befehl aufgefasst. Die einzelnen Spalten werden durch Leerzeichen oder Tabulatoren getrennt.

Beispiel einer *Crontab* unter Debian GNU/Linux:

SHELL=/bin/sh

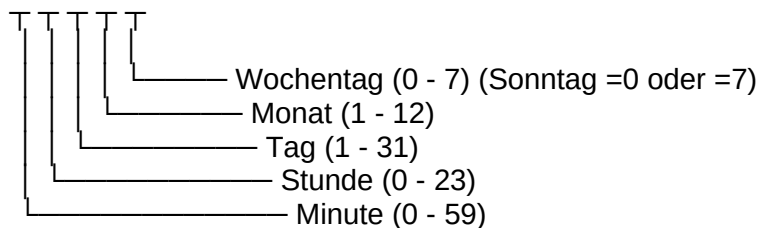
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

#M	S	T	M	W	Befehl
5	*	*	*	*	/usr/bin/message.sh
*5	*	*	*	*	/usr/bin/message.sh
59	23	*	*	0	cp /var/log/messages /log/backup/messages
0	0	*	*	*	cp /var/log/syslog /log/backup/syslog
20,30	1	*	*	1-5	/usr/bin/work.sh

Der erste Befehl (beginnend mit 5...) wird 5 Minuten nach jeder vollen Stunde, der zweite alle 5 Minuten (die Schrittweite wird durch \*/Schrittweite angegeben), der dritte einmal pro Woche Sonntag|sonntags um 23:59 Uhr, der vierte täglich um 00:00 Uhr und der fünfte montags bis freitags jeweils um 01:20 und 01:30 ausgeführt. Cron bietet noch mehr Funktionen, die bei den unten genannten Weblinks eingesehen werden können.

Die Struktur ist also wie folgt zu verstehen:

\* \* \* \* \* auszuführender Befehl



Mit dem Befehl `crontab -e` kann man die benutzereigene *Crontab* editieren. Um sich seine *Crontab* anzeigen zu lassen, kann man den Befehl `crontab -l` benutzen. Mit `crontab -u <user>` -e kann man auch *Crontabs* anderer Benutzer bearbeiten.

Für die Wochentage und die Monate können die drei ersten Buchstaben des englischen Namen benutzt werden, z. B. Sun für Sonntag. Groß-/Kleinschreibung spielt dabei keine Rolle, allerdings sind keine Angaben von Zeiträumen erlaubt, dazu müssen Ziffern verwendet werden.

Angaben von Zeiträumen und Schrittweiten innerhalb dieser Zeiträume sind erlaubt, sofern Ziffern benutzt werden. Setzt man beispielsweise für die Stunde 8-10, wird der Befehl in den Stunden 8, 9 und 10 Uhr ausgeführt. Statt der Schreibweise 8-10 kann auch 8, 9, 10 geschrieben werden.

Setzt man hingegen \*/2 wird der Befehl jede zweite Stunde ausgeführt. Beides kann kombiniert werden, so bedeutet 1-9/2 dasselbe wie 1, 3, 5, 7, 9.

Man sollte aber bedenken, dass \* \*/2 \* \* \* beispielsweise den Befehl 60 mal (also zu jeder Minute) jede zweite Stunde ausführt.

Zur Vereinfachung können die ersten fünf Felder auch durch eine einzelne Zeichenkette ersetzt werden. Hierfür sind acht Schlüsselwörter definiert:

String	Bedeutung	cron-Schreibweise
@reboot	einmalig, beim Start	
@daily	einmal pro Tag	0 0 * * *
@midnight	einmal pro Tag	0 0 * * *
@hourly	einmal pro Stunde	0 * * * *
@weekly	einmal pro Woche	0 0 * * 0
@monthly	einmal im Monat	0 0 1 * *
@annually	einmal im Jahr	0 0 1 1 *
@yearly	einmal im Jahr	0 0 1 1 *

Häufig führt der Cron-Daemon wichtige Programme für die Instandhaltung des Systems aus, wie zum Beispiel Dienste für das regelmäßige Elektronische Archivierung oder das Archivieren und Löschen von Logdateien.

Hinweis: Systemweit gibt es eine crontab Datei. Datei steht in /etc/crontab hierzu werden root Rechte benötigt.

Mit crontab -e kann jeder Nutzer eine eigene Crontab erstellen.

**Aufgabe:** Montag bis Freitags soll von 15:00 Uhr bis 17:00 Uhr jede Minute an eine Datei mit dem Namen muellhalde eine Zeile angehängt werden.

In der Zeile soll stehen:

Dies ist die Müllhalde von <Namen>. Es ist jetzt <Datum und Uhrzeit>

Verfolgt in einem zweiten Terminal-Fenster , wie eure Datei oder die Datei von jemand anderem immer länger wird (wenn auch nicht besonders schnell)

### Tipps:

- In jedem Linux-System ist ein Hintergrund-Programm namens crond (Cron-Dämon) installiert, welches regelmäßige Arbeiten erledigen kann. Mit dem Befehl crontab lässt sich dessen „Arbeitsauftrag“ bestimmen. (Hilfe über man crontab) Der genaue Aufbau der Crontab-Datei findet sich auf der fünften Manual-Seite (crontab(5)), die über man 5 crontab aufgerufen werden kann.
- man zeigt Hilfe-Seiten (Manual-Pages) mit Hilfe des Anzeigetools less an. In less kann man Seitenweise vorwärts oder zurückblättern und mit Hilfe der Eingabe „/suchwort“ nach einem Wort suchen (Weitersuchen mit „/“). Beenden kann man less mit der Taste Q.
- Mit Hilfe des Befehls echo „bla“ kann man einen Text ausgeben lassen, echo „bla“ >> dateiname hängt diesen an eine Datei an.
- Der Befehl date gibt das aktuelle Datum/Uhrzeit aus (auf Wunsch auch nach eigenen Vorstellungen formatiert...).
- Der Befehl tail <dateiname> zeigt das Ende einer Datei an, das Schalter (Flag) -f (für follow) hilft, Änderungen zu verfolgen (zum Beispiel in Protokoll-Dateien)
- Weiterführend (wenn noch Zeit ist): Der Befehl tar dient dazu, mehrere Dateien zusammenzufassen, mit Hilfe des Schalters -z werden die Dateien auch gleich mit gzip gepackt, sie entsprechen dann in etwa den bekannten ZIP-Dateien und haben die Endung .tar.gz. Als Test findest du im /tmp-Verzeichnis einen gepackten Ordner, kopiere die Datei mit cp (für copy) in dein Homedirectory und entpacke sie dort. Versuche auch, selber ein gepacktes .tar.gz-Archiv (auch „Tarball“ genannt) zu erstellen