

Paket EASYCRON

Version 3.10.4

Frank Meyer Das fli4l-Team
E-Mail: frank@fli4l.de E-Mail: team@fli4l.de

25. Oktober 2015

Inhaltsverzeichnis

| | |
|--|----------|
| 1. Dokumentation des Paketes EASYCRON | 3 |
| 1.1. EASYCRON - Befehle zeitgesteuert ausführen | 3 |
| 1.1.1. Konfiguration | 3 |
| 1.1.2. Beispiele | 3 |
| 1.1.3. Voraussetzungen | 4 |
| 1.1.4. Installation | 4 |
| A. Anhang zum Paket EASYCRON | 5 |
| A.1. EASYCRON - Crontab in der Boot-Phase ergänzen | 5 |
| Abbildungsverzeichnis | 7 |
| Tabellenverzeichnis | 8 |
| Index | 9 |

1. Dokumentation des Paketes EASYCRON

1.1. EASYCRON - Befehle zeitgesteuert ausführen

Dieses Paket wurde von Stefan Manske E-Mail: fli4l@stephan.manske-net.de zusammengestellt und vom fli4l-Team an 2.1 angepaßt.

1.1.1. Konfiguration

Mit `OPT_EASYCRON` kann man über das entsprechende config-file gesteuert zu bestimmten Zeiten Befehle ausführen lassen.

Dabei werden folgende Einträge benutzt:

OPT_EASYCRON mit `OPT_EASYCRON='yes'` wird das Paket aktiviert

Standard-Einstellung: `OPT_EASYCRON='no'`

EASYCRON_MAIL Da immer wieder Probleme auftraten, dass der crond unerwünschte Mails verschickt, kann man dies generell mit diesem Flag verhindern.

Standard-Einstellung: `EASYCRON_MAIL='no'`

EASYCRON_N Die Anzahl der verschiedenen Befehle, die von cron gestartet werden sollen.

EASYCRON_x_CUSTOM Wer sich mit den Einstellungen in der crontab auskennt, kann hier für jeden Eintrag eigene Einstellungen wie `MAILTO`, `PATH`, ... einstellen. Mehrere Einträge müssen durch `\\` getrennt werden. Hier sollte man sich aber wirklich mit cron auskennen.

Standard-Einstellung: `EASYCRON_CUSTOM=""`

EASYCRON_x_COMMAND In `EASYCRON_x_COMMAND` wird der gewünschte Befehl eingetragen, wie z.B.

```
EASYCRON_1_COMMAND='echo 1 '>' /dev/console'
```

EASYCRON_x_TIME In `EASYCRON_x_TIME` wird die Ausführungszeit gemäß der üblichen cron-Syntax eingetragen.

1.1.2. Beispiele

- Der Computer wünscht uns "Ein gutes neues Jahr"

```
EASYCRON_1_COMMAND = 'echo Ein gutes neues Jahr! > /dev/console'
EASYCRON_1_TIME      = '0 0 31 12 *'
```

- xxx wird von Montag bis Freitag jeweils von 7-20 Uhr zu jeder vollen Stunde ausgeführt.

1. Dokumentation des Paketes EASYCRON

```
EASYCRON_1_COMMAND = 'xxx'
EASYCRON_1_TIME     = '0 7-20 0 * 1-5'
```

- Der Router beendet jede Nacht um 03:40 die Internet-Verbindung die per DSL aufgebaut ist baut sie nach 5sec Wartezeit wieder auf. Die folgenden Devicenamen sind möglich: pppoe, ippp[1-9], ppp[1-9].

```
EASYCRON_1_COMMAND = 'fli4lctrl hangup pppoe; sleep 5; fli4lctrl dial pppoe'
EASYCRON_1_TIME     = '40 3 * * *'
```

Weitere Informationen zur cron-Syntax finden Sie unter

- <http://www.pro-linux.de/artikel/2/146/der-batchdaemon-cron.html>
- http://de.linwiki.org/wiki/Linuxfibel_-_System-Administration_-_Zeit_und_Steuerung#Die_Datei_crontab
- <http://web.archive.org/web/20021229004331/http://www.linux-magazin.de/Artikel/ausgabe/1998/08/Cron/cron.html>
- http://web.archive.org/web/20070810063838/http://www.newbie-net.de/anleitung_cron.html

1.1.3. Voraussetzungen

- fli4l in einer Version > 2.1.0
- für ältere Versionen bitte die entsprechenden opt_easycron-Versionen aus der OPT-Datenbank verwenden

1.1.4. Installation

OPT_EASYCRON wird einfach wie jedes andere OPT im aktuelle fli4l-Verzeichnis entpackt.

A. Anhang zum Paket EASYCRON

A.1. EASYCRON - Crontab in der Boot-Phase ergänzen

Hinweis: Die folgenden Ausführungen richten sich nur an Entwickler von Opt-Paketen für den *fi4l* Router.

Ab Version 2.1.7 stellt das rc-Script von Easycron die Funktion `add_crontab_entry()` zur Verfügung. Durch Aufruf dieser Funktion können andere rc-Scripts ab Startposition rc101 bis Startposition 949 Einträge an die Crontab anhängen. Am Ende der Boot-Phase mit dem Starten des cron Daemon sind die zusätzlichen Einträge aktiv.

```
add_crontab_entry time command [custom]
```

Mit `time` wird die Ausführungszeit in cron Notation übergeben, siehe die Manpage zu `crontab(5)` (<http://linux.die.net/man/5/crontab>). `command` enthält den auszuführenden Befehl. Der dritte Parameter `custom` ist optional. Mit ihm können Umgebungsvariablen passend zum Befehl gesetzt werden. Soll mehr als eine Variable gesetzt werden, sind die Zuweisungen durch `\\` zu trennen. Bitte **nicht** die Umgebungsvariable `PATH` verändern, da sonst nachfolgende `crontab` Einträge nicht mehr korrekt abgearbeitet werden können.

```
#
# example I: normal use, 2 parameters
#
    crontime="0 5 1 * *"
    croncmd="rotate_i_log.sh"

    add_crontab_entry "$crontime" "$croncmd"

#
# end of example I
#

#
# example II: extended use, 3 parameters and
#               multiple environment values
#
    croncustom="source=/var/log/current \\ dest=/mnt/data/log"
    croncmd='cp $source $dest-`date +%Y%m%d`; > $source'
    crontime="59 23 * * *"

    add_crontab_entry "$crontime" "$croncmd" "$croncustom"

#
# end of example II
#
```

A. Anhang zum Paket EASYCRON

Die Richtigkeit der Einträge muß das aufrufende Script sicherstellen.

Abbildungsverzeichnis

Tabellenverzeichnis

Index

EASYCRON_MAIL, [3](#)
EASYCRON_N, [3](#)
EASYCRON_x_COMMAND, [3](#)
EASYCRON_x_CUSTOM, [3](#)
EASYCRON_x_TIME, [3](#)

OPT_EASYCRON, [3](#)