

Paket BBTUNE - Tuning-Tool für die BusyBox

Version 3.10.0-rxxxxx-testing

Helmut Backhaus

E-Mail: helmut.backhaus@gmx.de

Das fli4l-Team

E-Mail: team@fli4l.de

14. Dezember 2014

Inhaltsverzeichnis

1. Dokumentation des Paketes BBTUNE	3
1.1. BBTUNE - Konsoleneinstellungen, Alias	3
1.1.1. Installation des BBTUNE	3
1.1.2. BBTUNE - Die optionalen Parameter	4
1.2. Opt Alias	6
1.2.1. Einrichten von Alias	6
1.2.2. Beispiel	7
A. Anhang zum Paket BBTUNE	8
A.1. BBTUNE	8
A.1.1. Hilfe aus FLI4L	8
Index	9

1. Dokumentation des Paketes BBTUNE

1.1. BBTUNE - Konsoleneinstellungen, Alias

Mit BBTUNE ist es möglich, den Begrüßungstext variabel anzupassen. Das Prompt ist veränderbar um den Pfad sichtbar zu machen. Weiterhin können beim anmelden einige vielleicht wichtige Informationen angezeigt werden. Aliase können mit dem Opt einfach in der Config-Datei eingetragen werden. Alle diese Funktionen sind natürlich auch in einer SSH-Sitzung verfügbar.

1.1.1. Installation des BBTUNE

OPT_BBTUNE Standard-Einstellung: `OPT_BBTUNE='no'`

Um die Verschiedenen Funktionen dieses Opts nutzen zu können, ist die Änderung der Variable `OPT_BBTUNE` auf `'yes'` nötig. Erst dann sind alle folgenden Funktionen aktiv.

BBTUNE_SHOW Standard-Einstellung: `BBTUNE_SHOW='no'`

Um eine der folgenden Variablen, `BBTUNE_SHOW_XXX`, anzuzeigen oder zu aktivieren muss diese Variable auf `'yes'` gestellt werden. Dies gilt jedoch nicht für das Aliasing, dass ist separat schaltbar.

BBTUNE_SHOW_HELLO Standard-Einstellung: `BBTUNE_SHOW_HELLO='Welcome to fli4l !'`

In diese Variable `BBTUNE_SHOW_HELLO` wird der individuelle Begrüßungstext eingetragen. Ist diese Variable leer, also `BBTUNE_SHOW_HELLO=""`, wird die Standard Begrüßung ausgegeben.

Wichtig: *Achtung! Diese Variable ist in der Länge begrenzt, es sind nur 65 Zeichen möglich. Dies ist für die Lesbarkeit auf der Konsole erforderlich. Ein längerer Eintrag wird durch die Prüfung mit `mkfli4l` unterbunden.*

BBTUNE_SHOW_VER_MSG Standard-Einstellung: `BBTUNE_SHOW_VER_MSG='no'`

Hier wird eine Versions Meldung beim Anmelden ein- oder ausgeschaltet.

BBTUNE_SHOW_PATH Standard-Einstellung: `BBTUNE_SHOW_PATH='no'`

Mit dieser Variable ist die Anzeige des Pfads im Prompt ein- oder auszuschalten. 'no' bedeutet die FLI4L übliche Anzeige (Hostname und Version). Bei 'yes' wird der Linux übliche Standard (Hostname und Pfad) angezeigt, hier ist zu überlegen, ob dann eine Versionsanzeige beim Anmelden erforderlich ist.

BBTUNE_SHOW_KERNEL Standard-Einstellung: `BBTUNE_SHOW_KERNEL='no'`

Mit dieser Variable wird festgelegt, ob der geladene Kernel angezeigt werden soll.

BBTUNE_SHOW_UPTIME Standard-Einstellung: `BBTUNE_SHOW_UPTIME='no'`

Um die Aktuelle Uptime anzeigen zu lassen, muss diese Variable auf 'yes' gesetzt werden.

BBTUNE_SHOW_UPTIME_VAR Standard-Einstellung: `BBTUNE_SHOW_UPTIME_VAR='time'`

Und hier stellt man das Format zu `BBTUNE_SHOW_UPTIME` ein.

- 'time' bedeutet, dass die Uptime als Tage, Stunden und Minuten angezeigt wird.
- 'date' stellt die Startzeit als Datum und Uhrzeit dar.

BBTUNE_SHOW_DF Standard-Einstellung: `BBTUNE_SHOW_DF='no'`

Soll der Zusatz df (wie voll sind die Partitionen z.B. `df -h`) nach dem Startbild angezeigt werden. Kann zur Fehlereingrenzung sehr sinnvoll sein.

BBTUNE_SHOW_DF_DISP Standard-Einstellung: `BBTUNE_SHOW_DF_DISP='t'`

Hier wird die Anzeigevariante von df ausgewählt. Möglich ist:

- 't' bedeutet, dass df normal im Textformat angezeigt wird.
- 'g' stellt df als Bar Grafik dar.
- 'b' stellt beide Formen dar.

BBTUNE_SHOW_DF_WARN Standard-Einstellung: `BBTUNE_SHOW_DF_WARN='80'`

Wenn dieser Wert überschritten wird, stellt df diese Partitionen in einer anderen Farbe da. Auch dies ist zur Fehlereingrenzung sehr hilfreich.

1.1.2. BBTUNE - Die optionalen Parameter

Hier werden die Parameter beschrieben, die nicht zwingend zum Betrieb des Opts erforderlich sind. Sie haben vielmehr nur Einfluss auf das aussehen des Ausgabefeldes und die Darstellung von DF, es sind also nur Optische Einstellungen. Sie müssen nicht gesetzt werden, wenn sie nicht vorhanden sind, werden sie vom Opt selbst auf Standardwerte gesetzt. Sie können auch einzeln genutzt werden, um nur eine Einstellung vorzunehmen.

Wichtig: *Nicht alle Parameter funktionieren in allen Situationen, dies kann viele Ursachen haben, zum Beispiel Terminalprogramm, ssh Verbindung, welches Programm wird zum Verbinden genutzt, welches Betriebssystem und viele mehr. Es ist mir also nicht möglich jede Eventualität zu testen. Hier ist also etwas Eigeninitiative gefragt. Probiert es aus!*

Ich habe hier nur die Parameter kurz beschrieben und bin nicht weiter auf die Funktion eingegangen, da die Parameter zum größten Teil selbsterklärend sind.

BBTUNE_SHOW_COLOR_TEXT Standard-Einstellung: BBTUNE_SHOW_COLOR_TEXT='cyan'

Hier wird die Textfarbe von der Anzeige eingestellt.

Mögliche Werte: black, red, green, blue, brown, magenta, cyan, white, normal.

Der Wert "normal" stellt den default Wert der jeweiligen Konsole dar.

BBTUNE_SHOW_COLOR_BG Standard-Einstellung: BBTUNE_SHOW_COLOR_BG=' '

Hier wird die Hintergrundfarbe von der Anzeige eingestellt.

Mögliche Werte: Siehe BBTUNE_SHOW_COLOR_TEXT

BBTUNE_SHOW_COLOR_MODE Standard-Einstellung: BBTUNE_SHOW_COLOR_MODE=' '

Hiermit werden die verschiedenen Modi der farblichen Darstellung auf dem FLI4L für die Anzeige eingestellt. Hiermit entstehen sehr schöne Effekte.

```
BBTUNE_SHOW_COLOR_TEXT='brown'  
BBTUNE_SHOW_COLOR_MODE='bright'
```

Ergibt eine Gelbe Textfarbe.

Ein Blick in die Hilfe von "**colecho**" auf dem FLI4L sollte hier weiterhelfen. Ich habe diese Hilfen in den Anhang gelegt.

BBTUNE_SHOW_FRAME Standard-Einstellung: BBTUNE_SHOW_FRAME='yes'

Hier wird Eingestellt, ob um die Ausgabe ein Rahmen erstellt werden soll. Standardmäßig wird hier das Zeichen "#"verwand.

BBTUNE_SHOW_FRAME_SIGN Standard-Einstellung: BBTUNE_SHOW_FRAME_SIGN='#'

Hier kann das Zeichen, welches für den Rahmen benutzt werden soll eingestellt werden.

Wichtig: *Nicht alle Zeichen der Tastatur sind möglich, da es bei einigen Probleme mit der Darstellung gibt. Die möglichen Zeichen sind: 'a-z, A-Z, 0-9,#, +, !,\$, <, >, ?, & = und .'.*

BBTUNE_SHOW_DF_TEXT Standard-Einstellung: BBTUNE_SHOW_DF_TEXT='normal'

Hier wird die Textfarbe für die df Ausgabe eingestellt. Prinzipiell ist die Funktion hier genau so wie bei BBTUNE_SHOW_COLOR_TEXT.

BBTUNE_SHOW_DF_WARN_TEXT Standard-Einstellung: BBTUNE_SHOW_DF_WARN_TEXT='normal'

Hier wird die Textfarbe für die df Ausgabe bei überschrittenem Warnwert BBTUNE_SHOW_DF_WARN eingestellt. Prinzipiell ist die Funktion hier genau so wie bei BBTUNE_SHOW_COLOR_TEXT.

BBTUNE_SHOW_DF_BG Standard-Einstellung: BBTUNE_SHOW_DF_BG=' '

Hier wird die Hintergrundfarbe für die df Ausgabe eingestellt. Prinzipiell ist die Funktion hier genau so wie bei BBTUNE_SHOW_COLOR_BG.

BBTUNE_SHOW_DF_WARN_BG Standard-Einstellung: BBTUNE_SHOW_DF_WARN_BG=' '

Hier wird die Hintergrundfarbe für die df Ausgabe bei überschrittenem Warnwert BBTUNE_SHOW_DF_WARN eingestellt. Prinzipiell ist die Funktion hier genau so wie bei BBTUNE_SHOW_COLOR_BG.

BBTUNE_SHOW_DF_MODE Standard-Einstellung: `BBTUNE_SHOW_DF_MODE=' '`

Hier wird der colecho Mode für die df Ausgabe eingestellt. Prinzipiell ist die Funktion hier genau so wie bei `BBTUNE_SHOW_COLOR_MODE`.

BBTUNE_SHOW_DF_WARN_MODE Standard-Einstellung: `BBTUNE_SHOW_DF_WARN_MODE='normal'`

Hier wird der colecho Mode für die df Ausgabe bei überschrittenem Warnwert `BBTUNE_SHOW_DF_WARN` eingestellt. Prinzipiell ist die Funktion hier genau so wie bei `BBTUNE_SHOW_COLOR_MODE`.

1.2. Opt Alias

Opt ALIAS dient zum einfachen einrichten von Alias Befehlen und ist Teil von OPT BBTUNE. Es muss also nichts in den Profildateien geändert werden. Diese Arbeit übernimmt das Opt für euch.

Was ist ein Alias?

Mit einem Alias kann man eigene Befehle für die Konsole erstellen oder bestehende Befehle für seine Zwecke anpassen. Dies kann sehr sinnvoll sein, wenn man öfter auf der Konsole arbeitet. Um auf der Konsole abzufragen welche Aliase eingerichtet sind, gibt man einfach `alias` ein und bekommt eine Liste aller eingerichteten Aliase.

Wichtig: Um dieses Opt nutzen zu können, muss die `OPT_BBTUNE` unbedingt auf `'yes'` gesetzt werden, sonst geht das Aliasing nicht!

Hier ein Beispiel:

```
ALIAS_n_NAME='ll'
ALIAS_n_COMMAND='ls -la'
```

Wichtig: Die Befehle (`ORDER`) müssen syntaktisch einwandfrei sein, sonst kann das böse Folgen haben.

1.2.1. Einrichten von Alias

BBTUNE_ALIAS Standard-Einstellung: `BBTUNE_ALIAS='no'`

Das setzen der Variable auf `yes` erlaubt es, Aliase über die Konfigurationsschicht sehr komfortabel einzurichten.

BBTUNE_ALIAS_N Standard-Einstellung: `BBTUNE_ALIAS_N=' '`

Hier wird die Anzahl der möglichen Aliase eingestellt. Es werden auch nur die eingebunden, die hier eingestellt sind. Auch wenn viel mehr im folgenden vorhanden sind. Aber das ist beim FLI4L ja immer so.

BBTUNE_ALIAS_n_NAME Standard-Einstellung: `BBTUNE_ALIAS_n_NAME=' '`

Hier wird der neue Befehl definiert, also die Kurzform für einen vielleicht sehr langen Befehl. Wir wollen uns das Leben ja einfacher machen.

```
BBTUNE_ALIAS_n_NAME='ll'
```

BBTUNE_ALIAS_n_COMMAND Standard-Einstellung: BBTUNE_ALIAS_n_COMMAND=' '

Nun wird mit der BBTUNE_ALIAS_n_COMMAND der auszuführende Befehl festgelegt. Also das, was wirklich ausgeführt werden soll.

```
BBTUNE_ALIAS_n_COMMAND='ls -la'
```

1.2.2. Beispiel

Hier noch mal ein Beispiel wie man einen bestehenden Befehl ändert. Aber aufpassen, bei falscher Anwendung kann es sein, dass der Befehl dann gar nicht mehr funktioniert.

```
BBTUNE_ALIAS_n_NAME='ping'  
BBTUNE_ALIAS_n_COMMAND='ping -c4 '
```

A. Anhang zum Paket BBTUNE

A.1. BBTUNE

Anhang zum Paket BBTUNE

A.1.1. Hilfe aus FLI4L

Hier die Ausgabe von "colecho -?":

```
test-router:/ # colecho -?
Usage: colecho <TEXT> [COLOR] [BKGCOLOR] [MODE]
Example:/usr/local/bin/colecho rd bl blink
TEXT/BKGCOLORs:
  b      : black
  rd     : red
  gn     : green
  bl     : blue
  br     : brown (MODE br -> Yellow)
  mg     : magenta
  cy     : cyan
  w      : white
  any other: normal

MODEs:
  br    : bright/bold
  dk    : darken
  bl    : blinking
  inv   : inverse
  brbl  : bright-blinking
  brinv : bright-inverse
  dibl  : dim-blinking
  diinv : dim-inverse
  any   : normal
-?
```


Index

BBTUNE_ALIAS, [6](#)
BBTUNE_ALIAS_N, [6](#)
BBTUNE_ALIAS_n_COMMAND, [6](#)
BBTUNE_ALIAS_n_NAME, [6](#)
BBTUNE_SHOW, [3](#)
BBTUNE_SHOW_COLOR_BG, [5](#)
BBTUNE_SHOW_COLOR_MODE, [5](#)
BBTUNE_SHOW_COLOR_TEXT, [4](#)
BBTUNE_SHOW_DF, [4](#)
BBTUNE_SHOW_DF_BG, [5](#)
BBTUNE_SHOW_DF_DISP, [4](#)
BBTUNE_SHOW_DF_MODE, [5](#)
BBTUNE_SHOW_DF_TEXT, [5](#)
BBTUNE_SHOW_DF_WARN, [4](#)
BBTUNE_SHOW_DF_WARN_BG, [5](#)
BBTUNE_SHOW_DF_WARN_MODE, [6](#)
BBTUNE_SHOW_DF_WARN_TEXT, [5](#)
BBTUNE_SHOW_FRAME, [5](#)
BBTUNE_SHOW_FRAME_SIGN, [5](#)
BBTUNE_SHOW_HELLO, [3](#)
BBTUNE_SHOW_KERNEL, [4](#)
BBTUNE_SHOW_PATH, [4](#)
BBTUNE_SHOW_UPTIME, [4](#)
BBTUNE_SHOW_UPTIME_VAR, [4](#)
BBTUNE_SHOW_VER_MSG, [3](#)

OPT_BBTUNE, [3](#)