Linux (allgemein): Grub: Fallback-Strategie

Problem:

Man möchte mal einen neuen Kernel testen bzw. einen selber kompilieren.
Mit einem kleinen Eintrag in der `/boot/grub/menu.lst` und dem entsprechenden `default` ist der Kernel schnell geladen.

Was macht man aber, wenn wenn der Kernel nicht läuft und das System-Crashed?

Lösung:

Wir nutzen das Fallback-Feature von Grub.

Dazu müssen folgende Einträge in `/boot/grub/menu.lst` hinzugefügt/geändert werden:

```
default saved
timeout 1
fallback 1 2

title A
root (hd0,0)
kernel /kernel
savedefault fallback

title B
root (hd1,0)
kernel /kernel
savedefault fallback

title C
root (hd2,0)
kernel /kernel
savedefault
```

Erklärung:

Mit `default saved` weiß Grub, daß er auf ein zwischengespeichertes Ergebnis (`/boot/grub/default`) zurückgreifen muß.

Der `timeout 1` ist nur dafür da, möglichst schnell zu booten. Wir gehen schließlich von einem Server im RZ aus.

Die Einträge `fallback 1 2` und `savedefault fallback` geben die Fallback-Strategie an:

1. boote default 0 merke 1

(c) 2025 huschi <huschi@huschi.net>
URL: https://www.huschi.net/index.php?action=artikel&cat=2&id=215&artlang=de

2. boote default 1 merke 2

3. boote default 3

Dies kann natürlich beliebig variiert werden.

grub-set-default 0

Grub weiß nicht, ob ein Reboot erfolgreich war oder nicht. Daher merkt er sich jeweils den nächsten Zustand (`fallback`) in der Datei `/boot/grub/default`.

Damit er nach einem erfolgreichen Booten aber auch wieder auf das erste Kernel-Image zugreift, muß der gespeicherte Default-Wert zurück gesetzt werden.

Dazu nutzen wir ein eigenes Start-Script (`/etc/init.d/grub-set-default`), welches das automatisch nach einem erfolgreichem Booten für uns erledigt:

```
#!/bin/bash
#
# grub-set-default
#
# Script to reset default to GRUBDEFAULT in scenario where
# /boot/grub/menu.lst has fallback set
#
#

GRUBDEFAULT=0


case "$1" in
  start)
        cp /boot/grub/default /boot/grub/old.default
        echo "setting grub default to " $GRUBDEFAULT
        grub-set-default $GRUBDEFAULT
        ;;
  stop)
        if [ -e /etc/deadman.d/imok ]; then
                echo "resetting deadman state..."
                rm /etc/deadman.d/imok
        fi
        ;;
  status)
        if [ -e /boot/grub/default ]; then
                cat /boot/grub/default
        fi
```


(c) 2025 huschi <huschi@huschi.net>

```
        ;;
  *)
        # do not advertise unreasonable commands
        echo $"Usage: $0 {start|stop|status}"
        exit 1
esac

exit $?
```

Startbar machen, als Startscript initialisieren und sicherheitshalber einmal laufen lassen:

```
chmod 0755 /etc/init.d/grub-set-default
update-rc.d grub-set-default defaults 90 80
/etc/init.d/grub-set-default start
```

Oder als Download:

```
wget http://www.huschi.net/download/grub-set-default.gz
gzip -dc grub-set-default.gz >/etc/init.d/grub-set-default
chmod +x /etc/init.d/grub-set-default

# Runlevel setzen:
#SuSE:
insserv grub-set-default
#Debian:
update-rc.d grub-set-default defaults
```

Was noch?

Das Einzige, was wir jetzt noch brauchen, wenn mal ein Kernel beim Booten abschmiert:

Eine Möglichkeit, den Reset-Knopf zu drücken!

Bzw. beachtet den ersten Kommentar unten.

Eindeutige ID: #1208
huschi
2008-03-04 11:07

(c) 2025 huschi <huschi@huschi.net>
URL: https://www.huschi.net/index.php?action=artikel&cat=2&id=215&artlang=de