Mini-Screen-Howoto

Screen ist ein Fenstermanager zur Verwendung mit textbasierten Eingabefenstern (Textkonsole) und kommt typischerweise auf UNIX-ähnlichen Betriebssystemen zum Einsatz. Hierbei ist es möglich, innerhalb eines einzigen Zugangs (zum Beispiel über ein Terminal oder eine Terminalemulation) verschiedene virtuelle Konsolensitzungen zu erzeugen und zu verwalten. Darüber hinaus können Sitzungen in den Hintergrund geschoben und später fortgeführt werden.

Steuerung von Screen

screen starten

```
$ screen <optional zu startendes Programm>
```

Die verschiedenen Funktionen in screen werden über ein Tastenkürzel „STRG-a + <x>“ (STRG und a werden zusammen gedrückt und losgelassen, danach folgt der Befehl) aufgerufen.

Hier die wichtigsten Kürzel:

```
STRG-a + c    # neues Window innerhalb der Session erstellen
STRG-a + leer # zum nächsten Window wechseln (oder STRG-a + n)
STRG-a + p    # zum vorherigen Fenster wechseln
STRG+a 0-9    # zum Fenster mit der entsprechenden Nummer wechseln
```

Beendet wird eine Screen-Session wenn das letzte Fenster geschlossen wird (Ausloggen mit STRG+d).

Soll die Session im Hintergrund weiterlaufen, kann man sie mit diesem Kürzel einfach abhängen (detach):

```
STRG-a + d
```

mit diesem Befehl kann eine Session aus dem Hintergrund hervorgeholt werden:

```
screen -list # laufende Screen-Sessions anzeigen

screen -r <pid> # auf eine abgehängte Session verbinden
oder
screen -x  <pid> # auf eine NICHT abgehängte Session verbinden. Damit kann man z.B. eine Shell m
```

Es reichen übrigens die Anfangszahlen einer PID (bis sie nicht mehr mit anderen Sessions verwechselt werden kann:

```
There are screens on:
 18209.pts-18.server (01/11/2013 11:11:23 AM) (Detached)
 19097.pts-24.server (01/11/2013 11:11:07 AM) (Detached)
2 Sockets in /var/run/screen/S-user.

$ screen -r 19 ## holt den Screen mit der PID 19097 zurück
```

Scrollen in Screen

Shift+Bild-hoch/runter funktioniert im Screen leider nicht. Dafür gibts den „Copy Mode“. Dieser wird über „STRG+a ESC“ aktiviert. Jetzt kann man mit Bild-hoch/runter arbeiten. Zum Verlassen des Copy Mode „ESC“

drücken.

In meiner Screenrc ist eine Option, die das Scrollen mit BILD HOCH/RUNTER wieder aktiviert. Allerdings wird dieser Terminal-Buffer zw. allen offenen Fenstern geteilt (Im Gegensatz zum Screen-internen Buffer).

gemeinsame Screensession / Shared Screen

Es ist möglich einen Screen gemeinsam zu bedienen. Das ist z.B. für Fernsupport nützlich oder um gemeinsam an einem Problem zu arbeiten.

Dazu startet user1 einen Screen und aktiviert den Multiuser-Mode:

```
STRG-a :multiuser on
STRG-a :acladd user2
```

Der user2 kann sich nun mit dem Screen verbinden (er muss screen nicht nochmals starten):

```
# screen -x [screen-id]
```

Sind die User die gleichen (also z.B. root auf einem Server) kann man sich das acladd sparen.

screenrc

Mit einer screenrc lassen sich einige Optionen setzen, die das ganze noch etwas komfortabler machen. Die Screenrc kann entweder als .screenrc im Homeverzeichnis oder als /etc/screenrc global abgelegt werden. Hier ist meine aktuelle .screenrc:

```
#
# magenbrot screenrc
#
# $Id: screenrc 125 2013-01-10 09:59:41Z magenbrot $
#

# disable startup message
startup_message off

# the status line and window caption
hardstatus alwayslastline "%{yb}  [ $LOGNAME@%H ]  %{gb} %0c:%s | %d.%m.%Y %{rb} %l %{wb} %w "

#defbce on
deflogin off

# scrollback history
defscrollback 10000

# visual bell off (screen flicker)
vbell off
bell_msg ""

# detach on disconnect
autodetach on

# make search case-insensitive
ignorecase on

# display time for warnings and messages
#msgminwait 10
msgminwait 0

# buffer emulator enables scrollback with PAGE-UP/DOWN (with some drawbacks, all windows share t
termcapinfo xterm|xterms|xs|rxvt ti@:te@
```

Revision #1
Created 2021-05-31 12:17:35 UTC by magenbrot

Updated 2021-05-31 12:18:11 UTC by magenbrot