

```

lloyd@debian:~/temp$
F10 key ==> File Edit Search Buffers Windows S

This is a scratch buffer. It is NOT saved when you exit.

To access the menus, press F10 or ESC-m and the usual
keys to navigate.

Latest version information is available on the web at
<http://www.jedsoft.org/jed/>. Other sources of JED
information include the usenet newsgroups comp.education.
alt.lang.s-lang. To subscribe to the jed-users mailing list
<http://www.jedsoft.org/jed/maillinglists.html>.

Copyright (C) 1994, 2000-2006 John E. Davis
Email comments or suggestions to <jed@jedsoft.org>
() All 2:1

0 shell
top - 02:14:15 up 36 min, 6 users, load average: 0.11
Tasks: 158 total, 1 running, 157 sleeping, 0 stopp
Cpu0 : 2.3%us, 0.7%sy, 0.0%ni, 97.0%id, 0.0%wa,
Cpu1 : 0.9%us, 2.1%sy, 0.0%ni, 96.7%id, 0.0%wa,
Mem: 3977876k total, 782900k used, 3194976k free,
Swap: 2947916k total, 0k used, 2947916k free,

1 editor
1 [|||] 3.8% Tasks: 145 total, 1 running
2 [||] 2.4% Load average: 0.11 0.12 0.
Mem[||||] 387/3884MB Uptime: 00:36:56
Swp[ ] 0/2878MB

PID USER PRI NI VIRT RES SHR S CPU% MEM% T
3350 lloyd 20 0 318m 80m 30m S 4 2.1 1:
3207 lloyd 20 0 63332 17m 11m S 3 0.4 0:
9316 lloyd 20 0 2284 1144 928 R 2.5 0.0 0:
9308 lloyd 20 0 2260 1156 876 S 2.5 0.0 0:
3458 lloyd 20 0 318M 81972 31392 S 1.2 2.1 0:
3058 root 20 0 293M 21428 6232 S 0.6 0.5 1:
3253 lloyd 20 0 93348 31548 11400 S 0.6 0.8 0:
3246 lloyd 20 0 65776 20144 12364 S 0.0 0.5 0:
1 root 20 0 1984 732 636 S 0 0.0
2 root 15 -5 0 0 0 S 0 0.0

2 top
3 htop
F1Help F2Setup F3Search F4Invert F5Tree F6SortBy F7Nice

```

۱. Screen چیست؟

سکرین، یک مدیر پنجره‌ی تمام‌صفحه‌ی مبتنی بر متن است و قادر است چندین ترمینال مجازی را درون یک ترمینال فیزیکی شبیه‌سازی کند.

مفید برای مواقعی که نیاز یا دسترسی به محیط گرافیکی و X نداریم، سرعت کار با سیستم برایمان در اولویت است و یا صرفاً دوست داریم نزد دیگران حرفه‌ای جلوه کنیم! به علاوه، سکرین، چندکارگی (multitasking) را به محیط ترمینال می‌آورد.

۲. راهنمای مقدماتی

این بخش قابلیت‌های مقدماتی سکرین را پوشش می‌دهد. با مطالعه‌ی آن به سرعت راه خواهید افتاد.

۱.۲ آغاز به کار

یک ترمینال باز کنید و بنویسید `screen`. بسته به نحوه‌ی کامپایل شدنش، ممکن است `splash screen` آن نمایش داده شود. سپس به محیط خط فرمان باز خواهید گشت انگار که هیچ اتفاقی نیفتاده باشد. اما

اتفاقی افتاده. اکنون در پنجره‌ی شماره صفر نشست قرار دارید. سکرین به هر پنجره یک نام و شماره اختصاص می‌دهد. شماره‌ها با ساختن پنجره‌ی جدید، یکی یکی افزایش می‌یابند. نام پنجره‌ها اما به طور پیشفرض، همنام برنامه‌ی در حال اجرا تنظیم خواهند شد.

فرمان‌های سکرین همچون ایمکس، با پیشوند همراه می‌شوند. پیشوند پیشفرض سکرین، C-a (کلید کنترل+A) است. برای نمونه با تایپ C-a w (کنترل و کلید A را بفشارید؛ سکرین منتظر دریافت دستور می‌ماند. حال کلید W را بزنید)، لیستی از پنجره‌های جاری و شماره‌های آنها نمایان خواهد شد.

سکرین، نواری با رنگ متضاد واقع در پایین صفحه را به نمایش اطلاعات ضروری اختصاص می‌دهد (همان status bar). توجه کنید، اگر سکرین را تحت X اجرا کرده باشید، به جای پایین صفحه، از نوار عنوان بهره می‌برد.

داخل پرانتز اینکه برای فرستادن یک C-a به برنامه‌ی در حال اجرا باید از C-a a استفاده کنید. مثلا داخل یک نشست سکرین، ssh زده‌اید و داخل این نشست ssh نیز باز سکرینی دیگر باز کرده‌اید. برای فرستادن C-a به سکرین داخلی، باید از دستور بالا استفاده کنید. حتی ممکن است در حال استفاده از یکی دیگر از برنامه‌های GNU - مثلا ایمکس - باشید و آن برنامه نیز از فرامین پیشوندی پشتیبانی کند.

۲.۲ ساختن پنجره‌ی جدید

برای ساخت پنجره‌ی جدید چند راه وجود دارد.

- C-a c کنید (c مخفف create). پنجره‌ی جدیدی حاوی شل لاگین شما ساخته و به آن منتقل خواهید شد.
- راه دیگر آن است که بنویسید screen و به طور بازگشتی سکرینی دیگر باز کنید. اکنون C-a w کنید. باید چیزی شبیه این نشان دهد:

```
0 bash 1- bash 2* bash
```

همانطور که حدس زدید، * نشان‌دهنده‌ی پنجره‌ی فعال است. علامت - نیز آخرین پنجره‌ی فعال را مشخص می‌کند.

- با تایپ دستوری با سینتکس screen [program_name] پنجره‌ای جدید باز شده و برنامه‌ی مورد نظر اجرا می‌شود. مثلا بنویسید screen vi text.txt. وی در یک پنجره‌ی جدید باز خواهد شد. اگر C-a w کنید خواهید دید این پنجره، vi نام گرفته:

```
0 bash 1 bash 2- bash 3* vi
```

۳.۲ جابه‌جا شدن میان پنجره‌ها

داشتن چند پنجره به تنهایی کافی نیست. باید بتوان میان آنها سوئیچ کرد. راه‌های مختلفی وجود دارد که هر کدام به منظوری خاص طراحی شده.

- سراسرترین روش `C-a [number]` کردن است. بنویسید `C-a 1` تا به پنجره‌ی شماره ۱ منتقل شوید. با `C-a w` نتیجه باید اینطور شود:

```
0 bash 1* bash 2 bash2 3- vi
```

- `C-a "` کنید تا لیستی از پنجره‌ها با قابلیت انتخاب فعال شود.
 - `C-a n` و `C-a p` به ترتیب پنجره‌ی بعدی (`n` مخفف `next`) و قبلی (`p` مخفف `previous`) لیست عددی را می‌آورد.
 - با تایپ `C-a C-a` آخرین پنجره‌ی فعال، یعنی همان پنجره‌ای که با علامت - مشخص شده باز می‌شود.
- با آزمودن این دستورات تفاوت دو بند آخر را مرور کنید.

۴.۲ بستن پنجره‌ها

دو راه برای بستن یک پنجره تعبیه شده.

- اولی استفاده از `C-a k` است (`k` مخفف `kill`). با صدور این فرمان پنجره‌ی فعال بسته شده و هر برنامه‌ای در حال اجرا باشد نیز خاتمه می‌یابد.
- به پنجره‌ی ۱ سوئیچ و `C-a w` کنید. باید چیزی شبیه این ببینید:

```
0 bash2 1* bash2 2 bash2 3- vi
```

بنویسید `C-a k` تا چنین چیزی نمایش داده شود:

```
0 bash2 2 bash2 3* vi
```

پنجره‌ی ۱ بسته شده و پنجره‌ی فعال قبلی، دوباره فعال می‌گردد. اگر پنجره‌ی جدید بسازید به آن شماره‌ی ۱ تعلق خواهد گرفت نه ۴.

- راه دیگر پایان دادن تمام برنامه‌های در حال اجرای یک پنجره است. مثلا به پنجره‌ی ۲ رفته و تایپ کنید exit تا شل بسته شود. نتیجه می‌شود:

```
0 bash2 3* vi
```

۵.۲ خارج شدن از Screen

برای خارج شدن از سکرین، یا تمام پنجره‌های ساخته شده را ببندید و یا فرمان `C-a \` را صادر کنید تا پس از کسب اجازه مبنی بر بستن تمام پنجره‌ها، خارج شود.

۳. راهنمای پیشرفته

وقتشه که ببینیم سکرین دیگه چه چیزای جالبی در چنته داره.

۱.۳ جدا شدن از نشست‌ها و اتصال دوباره

یکی بهترین قابلیت‌های سکرین، امکان جابه‌جا شدن میان ترمینال‌ها و ماشین‌ها بدون دست‌خورده شدن برنامه‌های در حال اجرای میزبان است. حتی اگر ارتباط شما با یک نشست قطع گردد، اسکرین به طور خودکار از آن جدا می‌شود. هر زمان دیگری قادرید دوباره و به سادگی متصل شوید.

برای جدا کردن دستی یک نشست، تایپ کنید `C-a d` تا به شل اصلی باز گردید (d مخفف detach). پیغامی مانند این نمایش داده خواهد شد:

```
[detached]
mdxi@fornax:~$
```

چه دستی جدا شوید چه ارتباط قطع گردد، روش متصل شدن یکسان است. اگر نمی‌دانید قبلا نشستی را جدا کرده‌اید یا نه، بنویسید `screen -ls`. اگر نشستی وجود داشته باشد خروجی به فرم زیر خواهد بود.

```
mdxi@fornax:~$ screen -ls
There is a screen on:
  31960.pts-0.fornax    (Detached)
1 Socket in /var/run/screen/S-mdxi.
```

```
mdxi@fornax:~$
```

اکنون با اجرای `screen -r` دوباره به نشست متصل می‌شوید (r مخفف reattach). همه چیز باید همانطور باشد که قبلا بود. هیچ برنامه‌ای به خاطر از دست دادن ارتباط بسته نشده.

اگر توجه کرده باشید برای اتصال، باید pid نشست را مقابل `screen -r` بنویسید. چنانچه می‌خواهید نشست‌ها به جای فرم `<host>.<tty>.<pid>`، نامی دلخواه بگیرند، نشست سکرین را با دستور زیر آغاز کنید.

```
screen -S <screen_name>
```

سکرین حتی اجازه‌ی اتصال به نشستی که خودتان آن را آغاز نکرده‌اید را نیز می‌دهد. برای این منظور به جای سوییچ `-r`، از `-x` استفاده کنید:

```
screen -x <screen_name or pid>
```

اگر سکرین اجازه‌ی اتصال نمی‌دهد (مثلا ارتباط قطع شده ولی توسط سکرین شناسایی نشده و تصور می‌کند هنوز متصل است. یا به خانه برگشته و فراموش کرده‌اید پیش از ترک محل کار، جدا شوید.) دستور `screen -DD -r` را صادر کنید تا دوباره لاگین کنید. می‌توانید برای دریافت یک شل لاگین نیز از این دستور استفاده کنید. اگر نشستی موجود نباشد، یکی ساخته می‌شود.

ترفندی کاربردی و جالب برای این قابلیت مربوط به زمانی است که به یک سرور ssh زده باشید. در سرور سکرینی باز کنید و فرامین مورد نظر را صادر و هر گاه نیاز بود detach کنید. اکنون می‌توانید با خیال آسوده ارتباط خود را با سرور قطع کنید. دستورات صادر شده در پس‌زمینه باقی می‌مانند و بار دیگر که لاگین کردید reattach کرده و کار را ادامه دهید.

۲.۳ اشتراک یک نشست

از امکانات عالی سکرین share کردن یک سشن است. دو نفر قادرند به طور همزمان یک برنامه را کنترل کنند یا یکی می‌تواند تنها بیننده‌ی کار دیگری باشد.

ابتدا در سیستمی که قرار است نشست خود را به اشتراک بگذارد:

```
# chmod u+s /usr/bin/screen  
# chmod 755 /var/run/screen
```

سپس سکرین را اجرا کرده و

- دستور `C-a :multiuser` را صادر کنید تا قابلیت چندکاربری فعال شود. (توضیح اینگونه دستورات در بند ۱.۴ آمده)
 - بنویسید `C-a :acladd guestuser`. نام کاربری کاربری که قرار است اشتراک بگیرد را جایگزین `guestuser` کنید.
- هم‌اکنون با سیستم مهمان `ssh` بزنید به ماشین میزبان و در ترمینال تایپ کنید:

```
screen -x hostuser/
```

مشخص است که باید نام کاربری سیستم میزبان را جایگزین `hostuser` نمود. هم‌اکنون `guestuser` متصل شده است و قادر به صدور فرمان است. اگر می‌خواهید نشست او را به صورت فقط خواندنی در آورید، از دستور زیر استفاده کنید.

```
C-a :aclchg guestuser -w "#"
```

۳.۳ رونوشت برداری و چسباندن

سکرین امکان برداشتن رونوشت و چسباندن آن - داخل خود پنجره یا در پنجره‌ای دیگر - را، حتی زمانی که آن را تحت `X` اجرا نکرده باشید دارا است.

برای شروع کپی تایپ کنید `C-a [` یا `C-a <Esc>`. چیزی شبیه این در نوار وضعیت ظاهر خواهد شد:

```
Copy mode - Column 17 Line 1(+1024) (80,24)
```

اکنون با استفاده از کلیدهای جهتی و `Page Up` و `Page Down`، مکان‌نما را به موقعیت مطلوب برده و کلید `space` را بفشارید:

```
First mark set - Column 17 Line 1
```

حال به انتهای سلکشن رفته و باز `space` کنید. خروجی نوار وضعیت:

```
Copied 122 characters into buffer
```

آزادید میان پنجره‌ها سوییچ کرده و هر جا دوست دارید بروید. سپس `C-a]` کنید تا متن کپی شده درج گردد.

۴.۳ تغییر نام دادن پنجره‌ها

گاهی یک پردازنده عنوان طویل و درازی به خود داده و روی اسمش در سکرین تأثیر می‌گذارد.

گاهی چندین شل گرفته‌اید و دوست دارید به هر کدام نامی منحصر به فرد بدهید.

گاهی فقط می‌خواهید جهت سرگرمی، اسم پنجره‌ها را عوض کنید. `C-a A` نموده و نام جدید را وارد کنید!

۵.۳ بُراندن و قطعه قطعه کردن یک پنجره

- `C-a S` ناحیه‌ی فعلی را به دو بخش جدید افقی تبدیل می‌کند. ناحیه‌ی جدید تا زمانی که جابه‌جا نشده‌اید، خالی می‌ماند.
- `C-a |` یک ناحیه‌ی عمودی اضافه می‌کند.
- با `C-a <Tab>` میان نواحی حرکت کنید.
- `C-a X` ناحیه‌ی فعال را می‌بندد.
- `C-a Q` تمام نواحی به جز آنکه درونش هستیم را حذف می‌کند.

۶.۳ ثبت تصویر لحظه‌ای و گزارشات

برای عکس گرفتن از صفحه به جای استفاده از نرم‌افزار پیش‌فرض سیستم عامل، تایپ کنید `C-a h`. فایل با نام `hardcopy.n` حاوی متن کامل موجود در کنسول، در دایرکتوری خانگی ساخته می‌شود.

قابلیت خوب دیگر سکرین امکان ثبت گزارش است. از `C-a H` استفاده کنید. فایل `screenlog.o` ساخته شده و هر کاری در سکرین انجام دهید ثبت می‌گردد. برای توقف عملیات نیز باز تایپ کنید `C-a H`. همچنین اگر نشست سکرین را با سویچ `-L` اجرا کنید، ثبت گزارش از ابتدا فعال می‌شود.

۷.۳ پشتیبان آنلاین

برای دیدن دستورات اضافه‌تر یا فقط به یاد آوردن فرامین فراموش شده، رفرنس سکرین همیشه در خدمت است. بنویسید `C-a ?`.

اگر فکر می‌کنید چیزی آنطور که باید کار نمی‌کند یا دوست دارید به اعماق بروید، man page معظم سکرین را مطالعه کنید.

۴. راهنمای پیشرفته‌تر

۱.۴ دستورات کامل

تناظری یک‌به‌یک میان تمامی فرامین کوتاهی که به آنها اشاره شد و آنهایی که نامی ازشان نبردیم، با عبارات کامل وجود دارد. بعضی از دستورات اصلا فرمان میان‌بر ندارند. بنویسید : C-a و دستور موردنظر را در این prompt وارد کنید. برای نمونه فرمان کامل خود : C-a هست colon . یا فرمان کاربردی bind برای اختصاص کلیدهای آزاد به وظایفی مشخص یا حتی تغییر وظیفه‌ی کلیدهای مشغول، به کار می‌رود. ده‌ها دستور این مدلی وجود دارد که لیست کامل آنها در صفحه‌ی راهنمای سکرین موجود است.

۲.۴ چند ترفند برای دستکاری فایل تنظیمات screen

این فایل screenrc نام دارد و در \$HOME/ واقع شده.

- اگر نمی‌خواهید پنجره‌ها از شماره‌ی صفر شروع شوند، خطوط زیر را به این فایل اضافه کنید.

```
bind c screen 1
bind ^c screen 1
bind 0 select 10
screen 1
```

از این به بعد اولین پنجره شماره‌ی ۱ را خواهد گرفت.

- به طور پیشفرض، سکرین از یک شبیه‌ساز ترمینال ۸-رنگه استفاده می‌کند. برای داشتن رنگ‌های بیشتر:

```
term screen-256color
```

ممکن است در xterm کار ندهد. پس به جای خط قبلی:


```
attrcolor b ".I" # allow bold colors - necessary for some reason
termcapinfo xterm 'Co#256:AB=\E[48;5;%dm:AF=\E[38;5;%dm' # tell screen how to set
colors. AB = background, AF=foreground
defbce on # use current bg color for erased chars
```

اگر از rxvt-unicode استفاده می‌کنید نیز خطوط زیر را اضافه کنید:

```
terminfo rxvt-unicode 'Co#256:AB=\E[48;5;%dm:AF=\E[38;5;%dm'
```

- خطوط زیر را اضافه کنید تا نام شل، لیست پنجره‌ها و تاریخ و زمان در نوار وضعیت نمایش یابد.

```
hardstatus off
hardstatus alwayslastline
hardstatus string '%{= kG}[ %{G}%H %{g}][%= %{= kw}%?-Lw%?%{r}(%{W}%n*
%f%t%?(%u)%?%{r})%{w}%?%+Lw%?%?%= %{g}][%{B} %m-%d %{W} %c %{g}]'
```

- پیغام خوش‌آمدگویی را خاموش کنید:

```
startup_message off
```

- با افزودن خطوط زیر، اطلاعات کامل‌تری در نوار عنوان نمایش می‌یابد. من جمله عبارتی همچون `screen (0 bash)` افزون بر تاریخ و زمان. همچنین لیست پنجره‌ها را رنگی می‌کند.

```
backtick 1 5 5 true
termcapinfo rxvt* 'hs:ts=\E]2;:fs=\007:ds=\E]2;\007'
hardstatus string "screen (%n: %t)"
caption string "%{= kw}%Y-%m-%d;%c %{= kw}%-Lw%{= kG}%{+b}[%n %t]%-b}%
{= kw}%+Lw%1`"
caption always
```

پشتیبانی شده در `urxvt`، `xterm` و `aterm`.

- سکروول توکار سکترین با فرمان `C-a [` قابل دسترسی است (همان حالت کی). با این وجود کار با آن شاید زیاد راحت نباشد. برای استفاده از سکروول بار مثلا `xterm` یا `konsole` و ... :

```
termcapinfo xterm* ti@:te@
```

متوجه هستید که این کار تنها تحت X قابل انجام است.

- اگر برنامه‌ای را خارج از سکرین اجرا کرده‌اید و اکنون قصد دارید آن را به سکرین متصل کنید، از بسته‌ی reptyr استفاده کنید. سینتکس دستور:

```
$ reptyr <pid>
```

- با دو خط زیر، C-a E و C-a W را متناظر کرده‌ایم با دو دستور emacs -nw با عنوان emacs و w3m ~/.w3m/bookmark.html با عنوان w3m.

```
bind E screen -t emacs emacs -nw
```

```
bind W screen -t w3m w3m ~/.w3m/bookmark.html
```

- اگر دوست دارید پس از باز کردن سکرین، پنجره‌هایی به طور خودکار باز شوند، از سینتکسی مشابه خطوط زیر استفاده کنید.

```
screen -t mutt mutt
```

```
screen -t emacs emacs -nw
```

```
screen -t w3m w3m ~/.w3m/bookmark.html
```

- سکرین تصویر ابتدای مطلب، اینگونه کانفیگ شده:

```
screen -t "shell" 0
```

```
split -v
```

```
split
```

```
focus
```

```
screen -t "top" 2 top
```

```
focus
```

```
screen -t "editor" 1 jed
```

```
split
```

```
focus
```

```
screen -t "htop" 3 htop
```

منابع:

<http://nathan.chantrell.net/linux/an-introduction-to-screen>

https://wiki.archlinux.org/index.php/GNU_Screen

<http://magazine.redhat.com/2007/09/27/a-guide-to-gnu-screen>

<http://arundelo.livejournal.com/390.html?nojs=1>

<http://www.saltycrane.com/blog/2008/01/how-to-scroll-in-gnu-screen>

<http://www.geekride.com/screen-linux-basic-tutorial>

<http://www.geekride.com/screen-linux-advanced>

ترجمه و گردآوری از آرمان اسماعیلی.

آدرس وبلاگ:

<http://armanes.ir>

آدرس ایمیل:

arman1992esmaeili@hotmail.com