

# Több parancssori ablak használata Tmux-szal

PCLinuxOS Magazine – 2015. november

Írta: Peter Kelly (critter)

Nemrég egy, a GNU Screen-nel foglalkozó cikkben említettem a tmux-ot, mint alternatívát és egy keveset írtam is róla. A terminált használók számára a Screen volt és most is a de-facto telepítendő frissítés, aminek az első kiadása 1987-ben jelent meg. Az új játékos a tmux, ami 2009 óta van jelen, ez a komputerék életében már jelentős kornak számít és ez idő alatt a screen népszerű és stabil alternatívájává nőtte ki magát.

A két program hasonló, de a tmux rendelkezik néhány extra jellemzővel és általában véve könnyebb beállítani. Ha a GNU screen tudja azt, ami neked kell, rendszerben, de, ha eszedbe jut valami, hogy „bárcsak megtudnám csinálni a screenben”, akkor ez az amiért a tmux-ot kifejlesztették. A tmux valószínűleg tudja.

A screen-t GNU General Public licenc alatt adták ki, miközben a tmux BSD licences. A Free Software Foundation mindkettőt jóváhagyta.

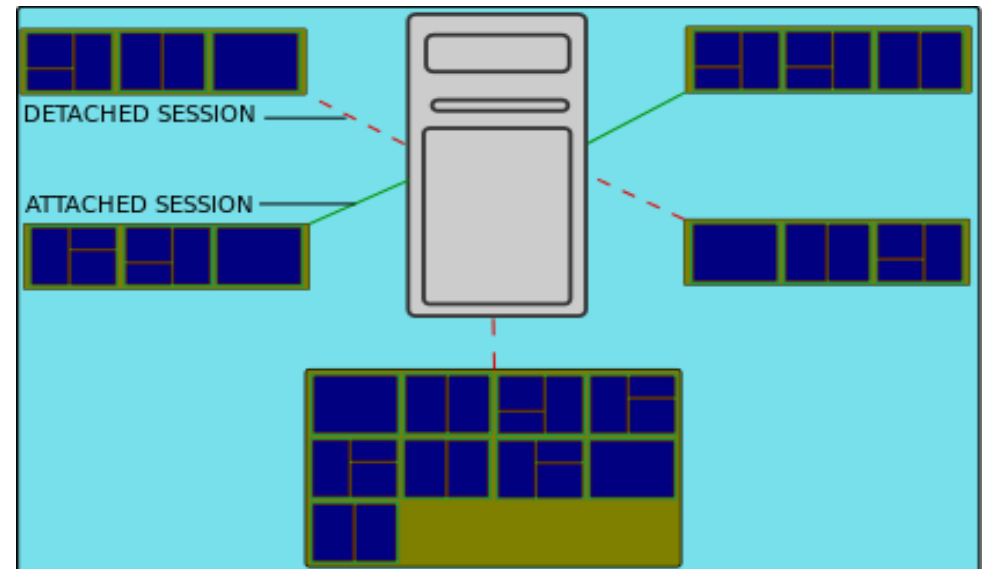
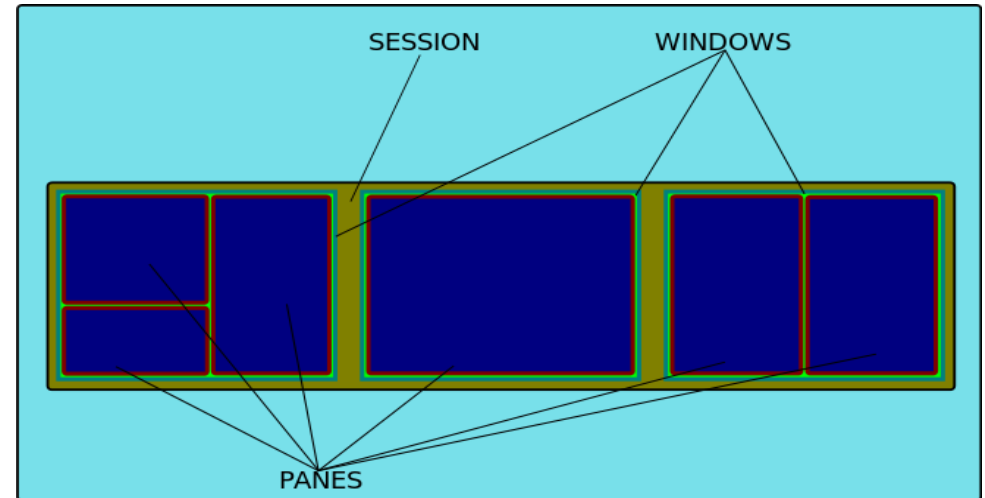
A tmux úgy írható le, hogy egy „**terminál alapú multiplexer alkalmazás**”, amittől nem fog a többség szíve hevesebben verni, ám a program kifejezetten hasznos lehet, ha bármennyi időt is töltesz terminálhasználattal. Az alkalmazás rendelkezik a screen legtöbb jellemzőjével, továbbfejlesztve azokat, és kiegészítve néhány trükkel és sajáttal. A legtöbb Linux-disztribúcióban megtalálható és az alap BSD-rendszerek is része.

Szintén az előző cikkben analógiát húztam a grafikus asztali környezetek, virtuális asztalok és a KDE aktivitásai között. A tmux kicsit közelebb hozza az előbbi állítás igazságát azáltal, hogy egy sokkal hatékonyabb munkafolyamat érdekében munkameneteket, ablakokat és táblákat használ.

A tmux alap műveleti egysége a „munkamenet”, amiből annyi lehet, amennyit csak szükségesnek tartasz, de valójában csak nagyon kevés kell. A tmux kliens-kiszolgáló kapcsolatot alkalmaz és a munkamenetek kapcsolhatók, leválaszthatók a szerverről, igény szerint. A munkamenet leválasztva addig futhat, amíg a kiszolgáló gép megy (vagy amíg szándékosan ki nem lövik). A munkamenet újracsatolása zökkenőmentes és lehetővé teszi, hogy onnan folytasd, ahol abbahagytad.

## Az elmélet

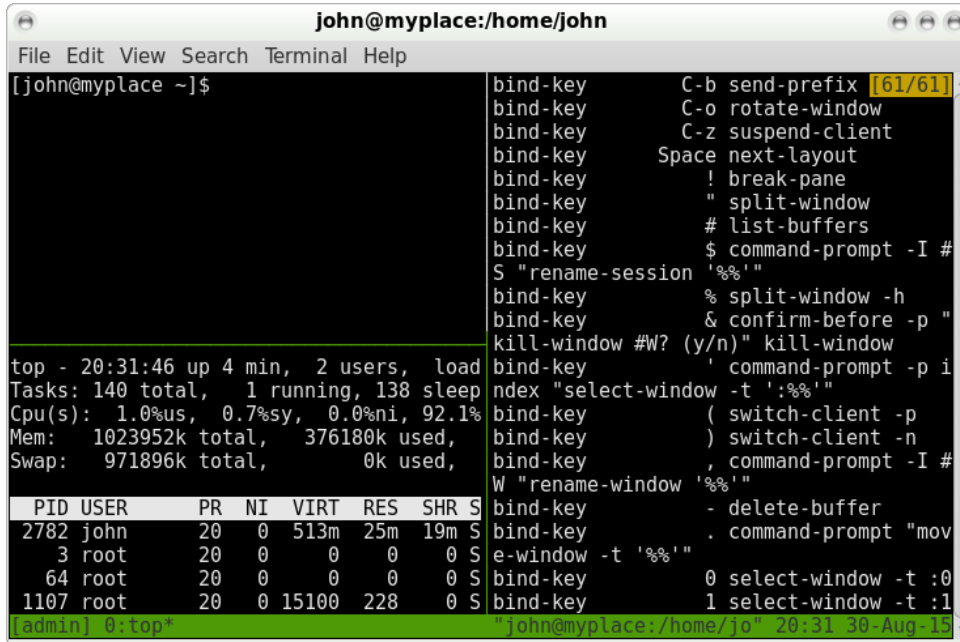
Minden munkamenet egy, vagy több „ablakot” tartalmaz, amik közül csak egy látható egy adott időben. Minden ablak egyetlen „táblával” indul, ami további, önálló folyamatokat tartalmazó táblákra osztható.



# Több parancssori ablak használata Tmux-szal

egy „log-files” ablakkal való létrehozásához a következő parancsot adtam ki:

```
tmux new -s admin -n log-files
```



```
john@myplace:/home/john
File Edit View Search Terminal Help
[john@myplace ~]$
bind-key C-b send-prefix [61/61]
bind-key C-o rotate-window
bind-key C-z suspend-client
bind-key Space next-layout
bind-key ! break-pane
bind-key " split-window
bind-key # list-buffers
bind-key $ command-prompt -I #
S "rename-session '%%'"
bind-key % split-window -h
bind-key & confirm-before -p "
kill-window #W? (y/n)" kill-window
bind-key ' command-prompt -p i
index "select-window -t '%%'"
bind-key ( switch-client -p
bind-key ) switch-client -n
bind-key , command-prompt -I #
W "rename-window '%%'"
bind-key - delete-buffer
bind-key . command-prompt "mov
e-window -t '%%'"
bind-key 0 select-window -t :0
bind-key 1 select-window -t :1
top - 20:31:46 up 4 min, 2 users, load
Tasks: 140 total, 1 running, 138 sleep
Cpu(s): 1.0%us, 0.7%sy, 0.0%ni, 92.1%
Mem: 1023952k total, 376180k used,
Swap: 971896k total, 0k used,
PID USER PR NI VIRT RES SHR S
2782 john 20 0 513m 25m 19m S
3 root 20 0 0 0 0 S
64 root 20 0 0 0 0 S
1107 root 20 0 15100 228 0 S
[admin] 0:top* "john@myplace:/home/jo" 20:31 30-Aug-15
```

Az egymáshoz kapcsolódásuk szerkezeti ábrája fent látható. A terminálban csak egy ablakot látsz és benne az aktuálisan tartalmazott és láthatóvá tett táblákat.

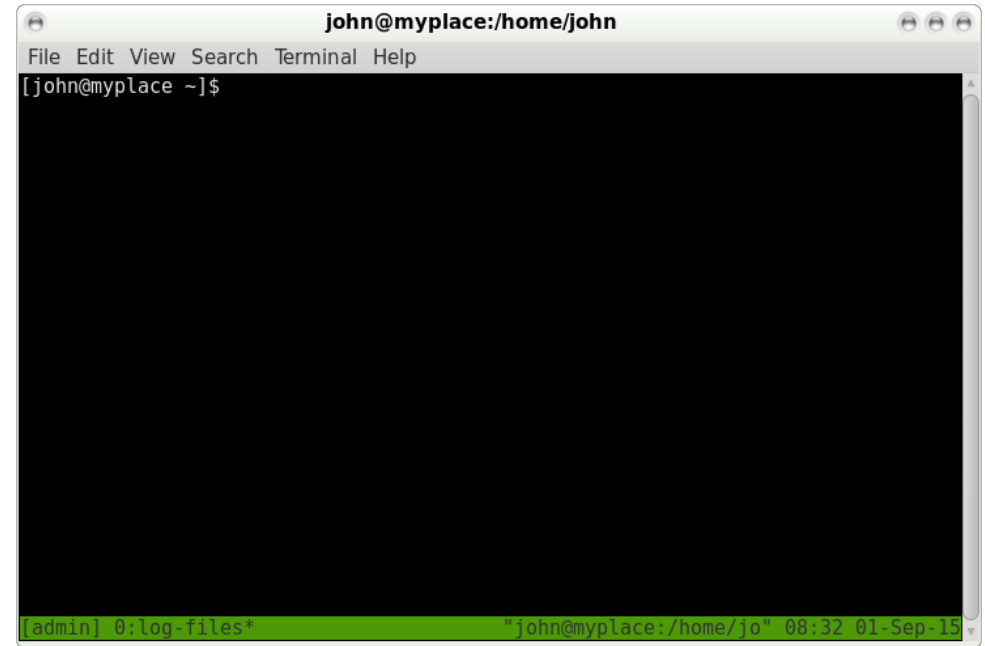
A képernyőképen egy munkamenetet futtatok, amit „admin”-nak neveztem el és az indító ablakát három táblára osztottam fel: egyiken a bash helyőrző, az alatta lévően a top parancs futása és továbbin jobbra pedig a jelenlegi konfigurációs beállítások láthatóak. Mindentől eltekintve az első futtatáskor ez így néz ki.

A legtöbb felhasználó további beállításokat végez, és döbbenetesen sokféle dolgot módosítható. A képernyő alján a státuszszáv nem azt mutatja, amit szeretnék, egy-két alapvető billentyűkombinációt is át akarok állítani.

Mind a GNU screen, mind a tmux egyfajta billentyűkombináció, vagy „előtag” használatát várja el valamilyen karakterrel a parancsbevitelhez. A screen előtagja a **Ctrl-a**, de a tmux-ban ez alából a **Ctrl-b** (innenről **C-b** formában hivatkozom rá), ami zavaró lehet és ezért később átállítom a screen-ben használt **Ctrl-a**-ra.

## Kezdeti lépések

A kezdéshez új munkamenetet kell nyitnunk. Ha nem nevesíted a munkamenetet, akkor is kap nevet „1”-től számozva, ami nem túl tartalmas. Az ablakok is nevesíthetők, ami szintén hasznos lehet, különösen, amikor újracsatolod a munkamenetet a szerverhez, amit már régen használtál. „Admin” munkamenet



```
john@myplace:/home/john
File Edit View Search Terminal Help
[john@myplace ~]$
[admin] 0:log-files* "john@myplace:/home/jo" 08:32 01-Sep-15
```

Ez megnyitja a munkamenetet is, a kurzorral az ablak egyetlen táblájában, ami 0 indexet és log-files nevet kapott. A státuszszorban a balra lent látható dolgok jelennek meg.

Most, hogy van futó munkamenetünk, gyorsan körbevezetlek, hogy bemutassam az elérhető jellemzők egy részét, így eldöntheted, olyasmi-e ez, ami hasznos neked. Azután megnézhetjük, hogyan szabhatjuk testre a dolgokat, hogy a te ízlésednek megfelelően kialakított eszközt kapjál. Munkamenet leválasztásához a „Ctrl-b d”-t és később, a legutóbbi munkamenet újracsatolásához a „tmux attach” parancsot használd. Ha több munkamenet van, akkor a munkamenet nevét használd, valahogy így: **tmux attach -t admin** Elég egyértelemű!

Az elérhető munkamenetek listázásához a „tmux ls”-t használd.

Amikor először nyitod meg a terminált, akkor az egyetlen tábla ablakában a preferált héj fut, a bash többnyire, várva a felhasználó utasításait. Ebben az első, nem testre szabott környezetben a **C-b** előtagot fogom használni.. Ha később úgy döntesz, hogy megváltoztatod, ahogy én is tervezem, akkor cseréld le új billentyűkombinációra. Az összes lent látható kombináció alapbeállítás, elég sok,



hogy megjegyezd, de amikor a beállító résznél leszünk, meglátod a jobban megjegyezhető kombinációk beállításának módját.

## Az ablakok tábláinak vezérlése

Ahhoz, hogy ez a szokásos termináltól több legyen, feloszthatjuk a jelenlegi képernyőt táblákra. Az osztás lehet akár vízszintes, akár függőleges is. Az vízszintes osztási parancs a „**C-b %**”.

A felosztás után a kurzor az új táblába kerül. Itt kiadva a „top” parancsot, a top program magára hagyható és vissza lehet váltani a bal oldali táblába, hogy valami mást csinálj. A bal és jobb oldalak közötti váltás a „**C-b bal nyíl**” és „**C-b jobb nyíl**” parancsokkal történhet.

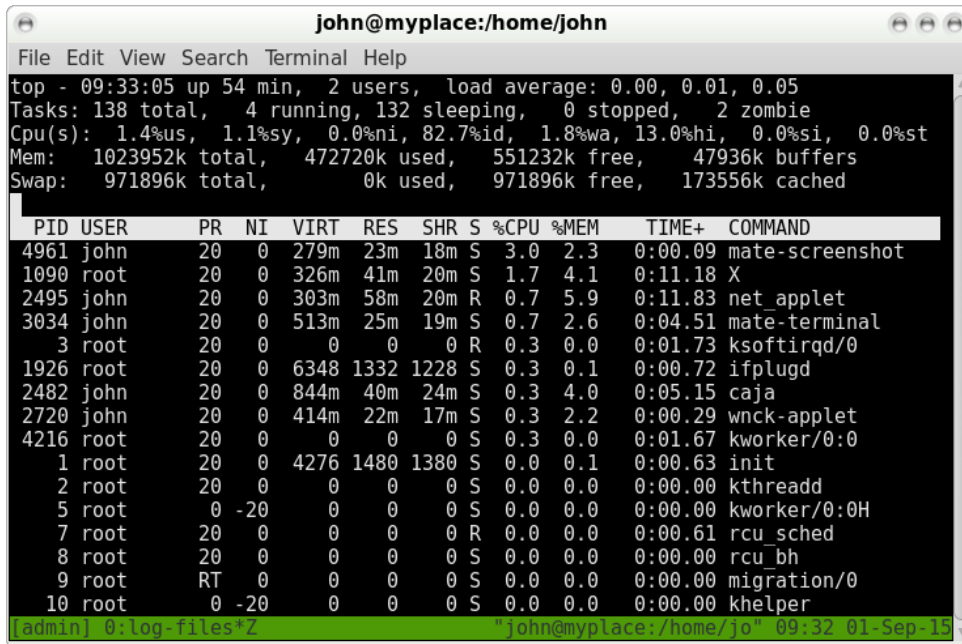
PID	USER	PR	NI	VIRT	RES	SHR
1090	root	20	0	326m	41m	20m
2444	john	20	0	712m	19m	16m
4050	john	20	0	279m	23m	18m
3	root	20	0	0	0	0
2269	root	20	0	680m	2104	1848
2446	john	20	0	516m	23m	19m
2482	john	20	0	843m	40m	24m
2495	john	20	0	303m	58m	20m
2532	john	20	0	265m	17m	13m
2553	john	20	0	131m	4272	3712
3034	john	20	0	513m	25m	19m
3073	john	20	0	34544	2804	2268
4065	root	39	19	25684	2972	2708
4086	root	39	19	0	0	0
1	root	20	0	4276	1480	1380
2	root	20	0	0	0	0

A tábla függőleges osztásának parancsa a „**C-b “**” (idézőjel).

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPT
sda	8:0	0	16G	0	disk	
├sda1	8:1	0	8.3G	0	part	/
├sda2	8:2	0	1K	0	part	
├sda5	8:5	0	949.1M	0	part	[SWAP]
├sda6	8:6	0	6.8G	0	part	/home
sr0	11:0	1	1024M	0	rom	



## Több parancssori ablak használata Tmux-szal



```
john@myplace:/home/john
File Edit View Search Terminal Help
top - 09:33:05 up 54 min, 2 users, load average: 0.00, 0.01, 0.05
Tasks: 138 total, 4 running, 132 sleeping, 0 stopped, 2 zombie
Cpu(s): 1.4%us, 1.1%sy, 0.0%ni, 82.7%id, 1.8%wa, 13.0%hi, 0.0%si, 0.0%st
Mem: 1023952k total, 472720k used, 551232k free, 47936k buffers
Swap: 971896k total, 0k used, 971896k free, 173556k cached

  PID USER      PR  NI  VIRT  RES  SHR  S  %CPU  %MEM   TIME+  COMMAND
 4961 john       20   0  279m  23m  18m  S   3.0   2.3   0:00.09 mate-screenshot
1090 root       20   0  326m  41m  20m  S   1.7   4.1   0:11.18 X
2495 john       20   0  303m  58m  20m  R   0.7   5.9   0:11.83 net applet
3034 john       20   0  513m  25m  19m  S   0.7   2.6   0:04.51 mate-terminal
   3 root       20   0   0     0     0  R   0.3   0.0   0:01.73 ksoftirqd/0
1926 root       20   0  6348 1332 1228  S   0.3   0.1   0:00.72 ifplugd
2482 john       20   0  844m  40m  24m  S   0.3   4.0   0:05.15 caja
2720 john       20   0  414m  22m  17m  S   0.3   2.2   0:00.29 wnck-applet
4216 root       20   0   0     0     0  S   0.3   0.0   0:01.67 kworker/0:0
   1 root       20   0  4276 1480 1380  S   0.0   0.1   0:00.63 init
   2 root       20   0   0     0     0  S   0.0   0.0   0:00.00 kthreadd
   5 root       0  -20   0     0     0  S   0.0   0.0   0:00.00 kworker/0:0H
   7 root       20   0   0     0     0  R   0.0   0.0   0:00.61 rcu_sched
   8 root       20   0   0     0     0  S   0.0   0.0   0:00.00 rcu_bh
   9 root       RT   0   0     0     0  S   0.0   0.0   0:00.00 migration/0
  10 root       0  -20   0     0     0  S   0.0   0.0   0:00.00 khelper

[admin] 0:log-files*Z "john@myplace:/home/jo" 09:32 01-Sep-15
```

A függőlegesen felosztások közötti lépkedés a „C-b fel nyíl” és a „C-b le nyíl”-lal. Mostanra az egyes táblák mérete elég kicsi lett, valójában a praktikus használathoz túl kicsi. Ahhoz, hogy lásd, mi folyik a top-ban, zoomolhatunk arra a táblára a „C-b z” paranccsal.

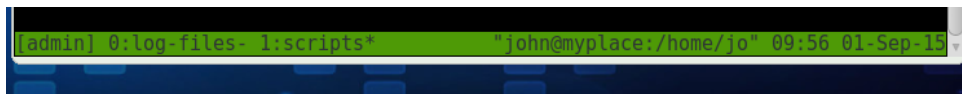
A „C-b jobb nyíl”-lal válts a megtekintendő táblára és „C-b z”-vel zoomolj. A státuszszoron a „0:log-files\*Z” olvasható, csillaggal jelezvén, hogy a log-files az aktív, és nagyítva egyetlen táblára.



```
(rename-window) scripts
```

A zoom visszavonása a „C-b z” ismétlésével. A zoomolási lehetőség ellenére is hamar elérjük a képernyőosztás használhatóságának határát. Készíthetünk új ablakot az aktuális munkamenetben a „C-b c”-vel és a „C-b ,”-vel (vessző) nevesíthetjük. Amikor az átnevezés prompt-ja megjelenik, visszatörlés az alpból kapott néven és az új név beírása.

Nyomj „Enter”-t és a státuszszor a munkamenetben két aktuális ablakot jelez, az 1-es indexű **scripts** nevű \*-gal, hogy aktív.



```
[admin] 0:log-files- 1:scripts* "john@myplace:/home/jo" 09:56 01-Sep-15
```

Lépkedhetsz az előző és a következő ablakok között a „C-b n” és „C-b p”-vel. Választhatsz ablakot az indexe segítségével, pl. „C-b 0” a log-files, az indexelt ablak promptja „ C-b ' ”-re (aposztróf) megjeleníthető, vagy lekérheted le az interaktív ablaklistát a választáshoz „C-b w”-vel.



```
john@myplace:/home/john
File Edit View Search Terminal Help
(0) 0: log-files- "john@myplace:/home/john"
(1) 1: scripts* "john@myplace:/home/john"
```

Bármikor lekérheted a parancsok billentyű-kombinációinak listáját „ C-b ? ”-lel, ami az Enter, vagy Return lenyomásával eltüntethető.

Tábla eltávolításához használj „C-b x”-et, ez megerősítést fog kérni alpból. Egy ablak és a táblái „C-b &”-tel távolítható el. Ez is feltehetően megerősítést kér, mivel egyirányú művelet – ha elment, akkor elment!

Tábla felosztása mindig fele-fele arányban történik, de nem kell emellett megmaradnod. A táblák átméretezhetők a következő parancsokkal:

**C-b C-up (Control b Control fel nyíl)**

**C-b C-down (le nyíl)**

**C-b C-left (balra nyíl)**

**C-b C-right (jobbra nyíl)**

Az átméretezés egyszerre egy karakternyi, vagy sornyi lesz. (5) betűnyi, vagy sornyi változáshoz „C-b Alt-up” stb. kell. Az (5)-ös érték átállítható.

A parancs „ C-b ! ” kiragadja az aktuális táblát az ablakból, új ablakot készítvén az adott egyetlen táblával. A megmaradt ablaktáblák megfelelően lesznek átméretezve.

A tmux saját prompttal bír, ami „C-b :” kombinációval hívható meg. Ezzel a tmux teljes irányítása átvehető. A **:list-commands** parancs, vagy rövidítve **:lscm** behív egy görgethető listát az összes elérhető paranccsal, ámde a teljes értékű használatához, elkerülhetetlen a kézikönyv (man) oldalainak elolvasása. A tmux man valójában a legjobb kézikönyv, ami a „picon” van, nagyon hasznos segítség.



## Másolás és beillesztés

A táblák közötti másolás és -beillesztés szövege előbb egy pufferbe kerül. Ehhez előbb ki kell választani a puffert, vagy fogadd el az alapbeállítást, ami a legutolsó létrehozott puffer lesz.

A másolási módba belépés a „**C-b** [”-vel lesz, ami után menj a másolandó rész elejére. Nyomj „**C-szóközt**” (**C-b** nem kell elé) a másolás indításához, majd ahogy mozogsz a szöveg kijelöltre vált. Nyomj „**Alt-w**”-t (ismét nincs **C-b**) ahhoz, hogy a szöveget elfogadd és a beillesztő pufferbe másolódjon, vagy nyomj „**Esc**”-et az eldobáshoz.

```

john@myplace:/home/john
File Edit View Search Terminal Help
[john@myplace ~]$

Swedish colonists built Nyenskans, a fortress, at the mouth of the Neva River in 1611, in a land
called Ingermanland, that was inhabited by Finnic tribe of Ingrians.[15] A small town called "Nyen" gr
ew up around it.
Peter the Great was interested in seafaring and maritime affairs, and he intended to have Russia gain
a seaport in order to be able to trade with other maritime nations.[16] He needed a better seaport tha
n Arkhangelsk, which was on the White Sea to the north and closed to shipping for months during the wi
nter.
The Bronze Horseman, monument to Peter the Great.
On May 12 [O.S. 1] 1703, during the Great Northern War, Peter the Great captured Nyenskans, and soon r
eplaced the fortress. On May 27 [O.S. 16] 1703,[17] closer to the estuary 5 km (3 mi) inland from the
gulf), on Zayachy (Hare) Island, he laid down the Peter and Paul Fortress, which became the first bric
k and stone building of the new city.[18].
@
"Documents/St Petersburg.txt" 40L, 7414C
2,1 Top
[admin] 0:log-files* 1:[tmux]- "john@myplace:/home/jo" 11:09 03-Sep-15
    
```

Most menj oda, ahová a szöveget beillesztenéd, másik táblába, vagy ablakba. A szöveget illeszd szövegszerkesztőbe, vagy hasonlóba, ne közvetlenül a héj promptjába, mivel a bash megpróbálja végrehajtani egyenként a sorokat, ami ártalmas eredménnyel járhat. A szöveg beillesztéséhez „**C-b ]**”-t írd be.



```

john@myplace:/home/john
File Edit View Search Terminal Help
GNU nano 2.4.2 File: research1.txt Modified

Peter the Great was interested in seafaring and maritime affairs, and he intended to have Russia
gain a seaport in order to be able to trade with other maritime nations.[16] He needed a better
seaport than Arkhangelsk, which was on the White Sea to the north and closed to shipping for months
during the winter.
The Bronze Horseman, monument to Peter the Great.
On May 12 [O.S. 1] 1703, during the Great Northern War, Peter the Great captured Nyenskans, and soon
replaced the fortress. On May 27 [O.S. 16] 1703,[17] closer to the estuary 5 km (3 mi) inland from
the gulf), on Zayachy (Hare) Island, he laid down the Peter and Paul Fortress, which became the first
k and stone building of the new city.[18].
@
2,1 Top
[admin] 0:log-files* 1:[tmux]- "john@myplace:/home/jo" 11:14 03-Sep-15
    
```

Ha egynél több kiválasztott szöveget másoltál, minden egyes kijelölés saját pufferbe kerül és ezek megjeleníthetők a „**C-b =**” parancssal. Használd a fel és le nyilakat a szöveg kiválasztásához, és nyomd le az Enter-t a szöveg azonnali beillesztéséhez. A szöveg, további használatra a pufferben megmarad.

A tartalmak kilistázásához anélkül, hogy egyet is kiválasztanál, használd a „**C-b #**”-et.

```

john@myplace:/home/john
File Edit View Search Terminal Help
(0) buffer0004: 555 bytes: During its first few years, the city developed around Trinity Square on t
(1) buffer0003: 528 bytes: The city was built by conscripted peasants from all over Russia; a number
(2) buffer0002: 710 bytes: Peter the Great was interested in seafaring and maritime affairs, and he
(3) buffer0001: 98 bytes: --profile filemanagement&'012# fif=Find in File (maik3531)012# e.g. to s
(4) buffer0000: 260 bytes: export GTK2_RC_FILES=$HOME/.gtkrc-2.0\012alias konq='konqueror --profile
    
```

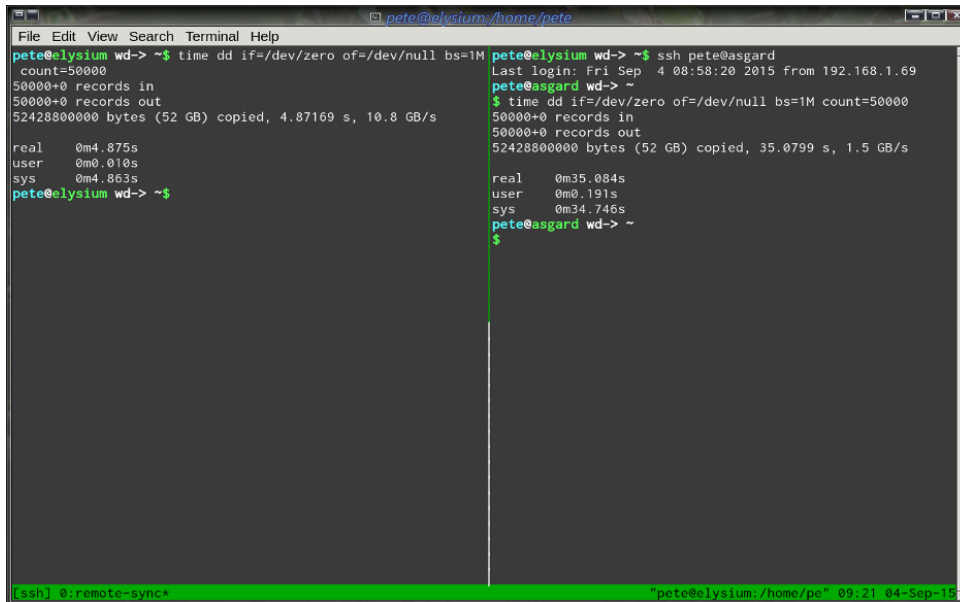
## Mindenféle

A tmux-hoz hasonló eszközöket sokszor használják hálózati kapcsolaton keresztül, úgymint ssh-val (secure shell connections – biztonságos héjkapcsolat), ahol leválasztható, és később újra csatlakozhat, amikor új kapcsolat jön létre.

A tmux egy hasznos jellemzője a synchronize-pane (táblák szinkronizálása) parancs („ **C-b :** ”-ot használj a prompthoz, majd géped be a parancsot. A

## Több parancssori ablak használata Tmux-szal

parancs lehetővé teszi, hogy utasításokat írj be egy táblába, miközben a szöveg az ablak összes táblájában megjelenik. Ideális könyvtártartalmak, vagy futási idők összehasonlítására számítógépek között.)



```
pete@elysium wd-> ~$ time dd if=/dev/zero of=/dev/null bs=1M count=50000
50000+0 records in
50000+0 records out
5242880000 bytes (52 GB) copied, 4.87169 s, 10.8 GB/s

real    0m4.875s
user    0m0.010s
sys     0m4.863s
pete@elysium wd-> ~$

pete@elysium wd-> ~$ ssh pete@asgard
Last login: Fri Sep  4 08:58:20 2015 from 192.168.1.69
pete@asgard wd-> ~$ time dd if=/dev/zero of=/dev/null bs=1M count=50000
50000+0 records in
50000+0 records out
5242880000 bytes (52 GB) copied, 35.0799 s, 1.5 GB/s

real    0m35.084s
user    0m0.191s
sys     0m34.746s
pete@asgard wd-> ~$
```

Úgy tűnik, a távoli gépre ráfér egy frissítés.

Van két elég korlátozottan használható parancs a „C-b t” és a „C-b q”.

Az első az időt mutatja:



```
"john@myplace:/home/jo" 11:37 03-Sep-15
```

A második parancs az egyes táblák indexét jeleníti meg nagy méretben, az aktuális piros, a többi kék színnel. Ez lehetővé teszi, hogy táblát válts az adott táblához tartozó szám lenyomásával – mindaddig, amíg a számok láthatóak. Gyorsnak kell lenned, mivel a számok egy másodpercre, vagy annyira se láthatóak!



```
john@myplace:/home/john
[john@myplace ~]$ nano research1.txt
[john@myplace ~]$

[john@myplace ~]$
[john@myplace ~]$
```

### A tmux beállítása

Négyféle dolgot kell (legalább) beállítani a tmux-ban:

- \* a tmux alapvető opcióinak beállításai;
- \* a státuszsor;
- \* a kedvenc billentyűkombinációk;
- \* döntés arról, hogy emacs, vagy vi beállításokat használjon-e alapból a másolási módban, azokra, amiket magad nem állítasz be.

Ha otthon vagy a vi-ban, akkor azokat a kombinációkat használd. Ez különösen azért hasznos, mert a vi mozgató parancsai kifejezetten gyorsak. Ha a vi nem a kedvenced, és én megértem, miért nem az, akkor, még ha nem is hallottál sosem az emacs-ról, használd az emacs billentyűpárokat, mivel a bash, vagy readline is alapbeállítás szerint ezeket használja a parancssor szerkesztésére: **C-e** a sor végére ugrás; **C-a** a sor elejére ugrás és így tovább. Ugyanakkor ne sokat törődj ezzel, mivel nem kell új nyelvet megtanulnod, csak egyszerűen állítsd be a dolgokat úgy, ahogy neked kényelmes.

A tmux megjelenésének és viselkedésének megváltoztatásához készíts egy `.tmux.conf` nevű fájlt a home könyvtárban. Minden, ami ebben kerül, módosítja a tmux alapműködését. Jóllehet sok beállítható opció van, a többség csak néhányat használ, a többi az alapbeállításon hagyja.

Vannak opciók szerverre, munkamenetre és ablakra, és ezek egyaránt lehetnek globálisak, vagy helyiek. Léteznek felhasználói opciók is. Ha ez egy kicsit bonyolultnak hangzik, ne ijedj meg, a kézikönyv rengeteg példát és magyarázó megjegyzést tartalmaz. Az Interneten is számos beállítási fájl mintája található.

A „**C-b**” előtag billentyű-kombináció GNU screen-hez hasonló „**C-a**”-ra cseréléshez add a **set -g prefix C-a** sort a beállító fájlodhoz. A „**-g**”-tól lesz globális érvényű. A tmux vi, vagy emacs mód választáshoz egy alábbi sort szintén add hozzá

```
setw -g mode-keys vi          vagy
setw -g mode-keys emacs
```

Amikor egy ablak számos táblára osztott, nem mindig egyértelmű, hogy melyik tábla aktív. A könnyebb láthatósága érdekében az alap táblaszegélyt kékre, míg az aktív tábla szegélyét világossárgára változtattam.

```
set-option -g pane-border-fg blue
set-option -g pane-active-border-fg brightyellow
set-option -g pane-active-border-bg default
```

Megjegyzéseket fűzhetsz hozzá, hogy a konfigurációs fájl jobban olvasható legyen. A megjegyzés a `#` jelet követi a sor végéig terjedően.

A tmux egeret korlátozottan támogat, én így engedélyeztem tábla kiválasztását.

```
setw -g mode-mouse on
set -g mouse-select-pane on
```

A man oldalak szerint az egér sokkal többre használható, de én nem fárasztottam magamat más funkciók engedélyezésével. Hé – végül is ez a parancssor.

Az ablakfelosztás parancsának könnyebb megjegyzéséhez a `%`-ot és `-et` `-ra` és `|-re` cseréltem, ami számomra logikusabbnak tűnik. Az ezt végrehajtó sorok

```
bind-key - split-window -v # tábla osztása vízszintesen
bind-key | split-window -h # tábla osztása függőlegesen
```

Ha megnézed „**C-b ?**”-lel az alapbeállítás szerinti billentyűkombinációkat, észreveheted, hogy néhány „**bind-key -r**”-rel kezdődik. Az **-r** azt jelenti, hogy a billentyűt lenyomva tartva ismételi. Ez különösen tábla átméretezésekor hasznos.

A konfigurációs fájlt akkor olvassa be, amikor új tmux munkamenet indul, ám hasznos lehet, ha a fájl szerkesztése után frissülne, különösen a beállítást során. Az új „**C-a**” előtagunkkal ez lehetséges lesz a „**C-a r**” hatására, ha ezt a sort hozzáfűzzük

```
bind-key r source-file ~/.tmux.conf
```

A státuszsor bal oldali és jobb oldali részből áll. Az, ami itt megjeleníthető, elég meggyőző, és alapvetően szövegből és néhány tmux változó értékéből áll, ezek mindegyike a kézikönyvben, használatuk módjával ki van listázva. Egyes változóknak van `#`-tel jelzett rövidítése, mint pl. a `#S`, a munkamenet neve; a `##` megjelenít egy `#`-et. Színeket és attribútumait (félkövér stb.) egy `#[]` kifejezés írja le, mint pl. a `#[fg=blue, bold]` (előtér, kék, félkövér).

A `#[default]` kifejezés visszaállítja a színek és attribútumok beállításait a globális alapértékre. Az idő és dátum formátuma az `strftime` szabványt követi. A többi megérzés, izzadság és káromkodás.

Az én konfigurációs fájlom így állt össze és így néz ki (én vi-t, vim-et használok, még ha nem különösebben szeretem is, ugyanakkor, be kell ismernem, hogy jó (valójában több, mint jó)) – te hogyan csinálnád?



**A konfigurációs fájl – ~/.tmux.conf**

```

setw -g mode-keys vi
set -g prefix C-a
bind-key a send-prefix
set-option -g history-limit 10000
setw -g mode-bg black
set-option -g default-terminal "screen-256color"
set-option -g pane-border-fg blue
set-option -g pane-active-border-fg brightyellow
set-option -g pane-active-border-bg default
set-window-option -g utf8 on
set-window-option -g xterm-keys on # for vim
set-window-option -g mode-keys vi # vi key
set-window-option -g monitor-activity on
set-window-option -g window-status-current-fg white
setw -g window-status-current-attr bold
setw -g automatic-rename
setw -g mode-mouse on
set -g mouse-select-pane on
set -g terminal-overrides 'xterm*'
set-option -g status-utf8 on
set-option -g status-justify right
set-option -g status-bg black
set-option -g status-interval 5
set-option -g status-left-length 30
set-option -g status-left '[fg=blue,bold]#T#[default]'
set-option -g status-right '[fg=blue,bold]#S [fg=green]%R %a %d %b#[default]'
set-option -g visual-activity on
bind-key r source-file ~/.tmux.conf
bind-key - split-window -v
bind-key | split-window -h
bind-key Space list-panes
bind-key Enter break-pane
bind-key -n C-up prev
bind-key -n C-left prev
bind-key -n C-right next
bind-key -n C-down next
set-window-option -g window-status-current-bg red
bind A command-prompt "rename-window %"
setw -g aggressive-resize on
bind-key -r Left select-pane -L
bind-key -r Down select-pane -D
bind-key -r Up select-pane -U
bind-key -r Right select-pane -R
setw -g mode-keys vi
    
```

