

tmux is a terminal multiplexer: it enables a number of terminals to be created, accessed, and controlled from a single screen. tmux may be detached from a screen and continue running in the background then later reattached.

```
tmux [command]      # Run a command
                    # 'tmux' with no commands will create a new session

new                 # Create a new session
-s "Session"        # Create named session
-n "Window"          # Create named Window
-c "/dir"           # Start in target directory

attach             # Attach last/available session
-t "#"             # Attach target session
-d                 # Detach the session from other instances

ls                 # List open sessions
-a                 # List all open sessions

lsw                # List windows
-a                 # List all windows
-s                 # List all windows in session

lsp                # List panes
-a                 # List all panes
-s                 # List all panes in session
-t                 # List all panes in target

kill-window         # Kill current window
-t "#"             # Kill target window
-a                 # Kill all windows
-a -t "#"           # Kill all windows but the target

kill-session        # Kill current session
-t "#"             # Kill target session
-a                 # Kill all sessions
-a -t "#"           # Kill all sessions but the target
```

Key Bindings

The method of controlling an attached tmux session is via key combinations called 'Prefix' keys.

(C-b) = Ctrl + b # 'Prefix' combination required to use keybinds

(M-1) = Meta + 1 -or- Alt + 1

```
?                 # List all key bindings
:                 # Enter the tmux command prompt
r                 # Force redraw of the attached client
c                 # Create a new window
```

```

!           # Break the current pane out of the window.
%           # Split the current pane into two, left and right
"           # Split the current pane into two, top and bottom

n           # Change to the next window
p           # Change to the previous window
{           # Swap the current pane with the previous pane
}           # Swap the current pane with the next pane

s           # Select a new session for the attached client
            interactively
w           # Choose the current window interactively
0 to 9      # Select windows 0 to 9

d           # Detach the current client
D           # Choose a client to detach

&           # Kill the current window
x           # Kill the current pane

Up, Down    # Change to the pane above, below, left, or right
Left, Right

M-1 to M-5  # Arrange panes:
            # 1) even-horizontal
            # 2) even-vertical
            # 3) main-horizontal
            # 4) main-vertical
            # 5) tiled

C-Up, C-Down # Resize the current pane in steps of one cell
C-Left, C-Right

M-Up, M-Down # Resize the current pane in steps of five cells
M-Left, M-Right

```

Configuring ~/.tmux.conf

tmux.conf can be used to set options automatically on start up, much like how .vimrc or init.el are used.

```

# Example tmux.conf
# 2014.10

### General
#####

# Enable UTF-8
setw -g utf8 on
set-option -g status-utf8 on

# Scrollback/History limit

```

```

set -g history-limit 2048

# Index Start
set -g base-index 1

# Mouse
set-option -g mouse-select-pane on

# Force reload of config file
unbind r
bind r source-file ~/.tmux.conf

### Keybinds
#####

# Unbind C-b as the default prefix
unbind C-b

# Set new default prefix
set-option -g prefix `

# Return to previous window when prefix is pressed twice
bind C-a last-window
bind ` last-window

# Allow swapping C-a and ` using F11/F12
bind F11 set-option -g prefix C-a
bind F12 set-option -g prefix `

# Keybind preference
setw -g mode-keys vi
set-option -g status-keys vi

# Moving between panes with vim movement keys
bind h select-pane -L
bind j select-pane -D
bind k select-pane -U
bind l select-pane -R

# Window Cycle/Swap
bind e previous-window
bind f next-window
bind E swap-window -t -1
bind F swap-window -t +1

# Easy split pane commands
bind = split-window -h
bind - split-window -v
unbind '"'
unbind %

# Activate inner-most session (when nesting tmux) to send commands
bind a send-prefix

```

Theme

#####

Statusbar Color Palatte

```
set-option -g status-justify left
set-option -g status-bg black
set-option -g status-fg white
set-option -g status-left-length 40
set-option -g status-right-length 80
```

Pane Border Color Palette

```
set-option -g pane-active-border-fg green
set-option -g pane-active-border-bg black
set-option -g pane-border-fg white
set-option -g pane-border-bg black
```

Message Color Palette

```
set-option -g message-fg black
set-option -g message-bg green
```

Window Status Color Palette

```
setw -g window-status-bg black
setw -g window-status-current-fg green
setw -g window-status-bell-attr default
setw -g window-status-bell-fg red
setw -g window-status-content-attr default
setw -g window-status-content-fg yellow
setw -g window-status-activity-attr default
setw -g window-status-activity-fg yellow
```

UI

#####

Notification

```
setw -g monitor-activity on
set -g visual-activity on
set-option -g bell-action any
set-option -g visual-bell off
```

Automatically set window titles

```
set-option -g set-titles on
set-option -g set-titles-string '#H:#S.#I.#P #W #T' # window number,program name,active (or not)
```

Statusbar Adjustments

```
set -g status-left "[fg=red] #H#[fg=green]:#[fg=white]#S#[fg=green] |[default]"
```

Show performance counters in statusbar

Requires <https://github.com/thewtex/tmux-mem-cpu-load/>

```
set -g status-interval 4
set -g status-right "[fg=green] | #[fg=white]#(tmux-mem-cpu-load)#[fg=green] | #[fg=cyan]%H:%M #[default]"
```

References

[Tmux | Home](#)

[Tmux Manual page](#)

[Gentoo Wiki](#)

[Archlinux Wiki](#)

[Display CPU/MEM % in statusbar](#)

[tmuxinator - Manage complex tmux sessions](#)