↖ Topic Index

Tmux 🚧

	Description	

Tmux
• **tmux @ home**
  • **Getting started**
  • **tmux man page**
• **Tmux Cheat Sheet**
• **Tmux Intro @ Red Hat**
 	
See also:
• **GNU screen**
	When using Emacs in a ssh terminal session connected to a remote computer, Tmux provides protection against network issues such as VPN network disconnections.
	• If you get disconnected you can re-establish the connection and re-connect to your original Tmux screen session and continue without loosing anything. That feature alone is worth learning to use Tmux.
	
	Reason for using Tmux with Emacs.
	• The main reason to use tmux (or GNU screen) with Emacs occurs when connecting to a remote host and establishing a session on that host over ssh. Using tmux allows you to directly interact with the shell, providing full control, including the ability to run Emacs on that remote host.
	  • You don't need tmux (nor GNU screen) to establish multiple windows, local sessions, etc… Emacs does all of that.
	  • You can use Emacs with Tramp to access files on the remote host and edit them, establishing protected session, since the content of remote files are copied to your local system.
	  • You can use Emacs Tramp to open a shell on the remote host too (see here for some examples).
	  • However, there's several constraints imposed by Tramp that may prevent you to control the remote shell behaviour. Like the shell prompt, ability to use various keys (since Tramp can only use the shell command). These limitations are not present when using a terminal multiplexer like tmux (or GNU screen).
	• Another great use of tmux is to allow more than 1 person to use the same shell at the same time.
	
	Tmux configuration file is: **~/.tmux.conf**	

List existing Tmux sessions	**`tmux ls`**	Use this to list tmux sessions. • It lists all sessions from all terminals on this host.
Start a Tmux session		
Start unnamed session	• **`tmux`** • **`tmux new`** • **`tmux new-session`**	Create a new session.
Start a named session	**`tmux new -s`** NAME	Create a new session named NAME and update the terminal title with it.
Detach a Tmux session		
Detach current session	**`C-b d`**	Assuming the tmux *action key* is **`C-b`** (the default), **`C-b d`** detaches the tmux session.
Restore/Attach-to Tmux session	List existing sessions with. screen -ls and pick a name from the list. You can type only the prefix portion that uniquely identifies it. ☝ More than 1 tmux client can attach to a tmux session. You can use this to allow multiple people interact on the same shell.	
Restore detached session	• **`tmux new-session -A -s`** NAME • **`tmux a -t`** NAME • **`tmux at -t`** NAME • **`tmux attach -t`** NAME • **`tmux attach-session -t`** NAME	Re-attach tmux to session specified by NAME.
Kill Tmux session		
	`tmux kill-session -t` NAME	Kill session specified by its NAME