↖ Topic Index

GNU screen 🚧

	Description
GNU Screen • **GNU Screen Manual** • **GNU Screen @ ArchLinux Wiki** • **GNU Screen @ Gentoo Linux Wiki** See also: • **Tmux**	When using Emacs in a ssh terminal session connected to a remote computer, the GNU screen utility 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 GNU screen session and continue without loosing anything. That feature alone is worth learning to use GNU screen. ☝PEL provides a the **.screenrc** GNU screen configuration file. • It sets **C-z** as the GNU screen control command key sequence. • **C-z** is a better choice than screen's default of **C-a** when using Emacs as it is used less often. See the notes in PEL's **.screenrc** file. • It also configures a nice status line, prevents the appearance of screen splash screen and some other niceties.
List existing screen sessions	`screen -ls` — Use this to list attached and detached screen sessions.
Start a screen session	
Start unnamed session	`screen` — Create a new session.
Start a named session	`screen -S NAME` — Create a new session named NAME and update the terminal title with it.
Attach another client to currently attached session	Multiple GNU screen clients can attach to the same session, allowing multiple users to access the same remote shell.
Multiple-user attach to existing attached GNU screen session	`screen -x NAME` — Attach a new client to an already attached GNU screen session. This allows multiple users to connect to the same shell.
Restore 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.
Restore detached session	`screen -r NAME` — Re-attach screen to session specified by NAME.
Inside Emacs	Since **C-z** is used for GNU screen command key, the Emacs command to suspend Emacs becomes the following under PEL
Suspend Emacs session	`C-x C-z z` — Suspends Emacs session when screen is running.
GNU Screen operations	Once a GNU screen session is started, use the **C-z** key to issue a GNU screen command.
GNU screen help	`C-z ?` — Show a list of GNU screen command key bindings.
Scroll Up half screen	`C-z [ C-u` — To scroll up or down: first activate GNU screen *copy mode* with **C-z [** and then use **C-u** and **C-d** to scroll up or down. Repeat **C-u** and **C-d** as required. • These scroll the display up/down by the specified amount of lines while preserving the cursor position. (Default: half screen-full
Scroll Down half screen	`C-z [ C-d`
Enter Copy mode	`C-z [` — Once in Copy mode, you can use the **C-d** and **C-u**, **<up>** and **<down>** keys to scroll and the following other keys
• **Scroll Up full screen**	`C-b` — Scroll the display up a full screen
• **Scroll Down full screen**	`C-f` — Scroll the display down a full screen
• **Exit Copy mode**	`q` — Exit back to the terminal
References:	• GNU Screen @ Wikipedia : start here if you do not know what this program is. • GNU Screen home page • GNU Screen Manuals • **GNU Screen Manual - all in 1 HTML Page** (useful to search)
GNU Screen source code	• GNU Screen Git Repository - Savannah

Compile GNU Screen:	`git clone https://git.savannah.gnu.org/git/screen.git`
	`cd screen/src`
	`./autogen.sh`
	`./configure --prefix=/usr/local \`
	`  --enable-pam \`
	`  --enable-colors256 \`
	`  --enable-rxvt_osc \`
	`  --enable-use-locale \`
	`  --enable-telnet`
	`make && make install`

Using Emacs within an GNU Screen Session	• By default GNU Screen uses the **C-a** key as the Screen command key.
	  • To pass **C-a** to Emacs running inside a GNU Screen session: type **C-a** followed by **a**
	  • Screen command key can be changed with the **escape** setting in the ~/.screenrc file. See next lines for 2 examples:
	  • To change it to **C-^**, write: **escape ^^^**
	  • The first ^^ is the caret representation of Control-^. The last ^ is the single key to type after to pass **C-^** to the program running under Screen (like Emacs). Another character could be used, 6 for example.
	  • To change it to **C-z**, write: **escape ^zz**

Logging with Screen	GNU screen supports dumping the current content of the screen to a file or **log the complete window session to a file**. • This second feature is quite useful when running long lasting commands like software builds preformed from a shell. • The session can be started inside a screen window, and hidden to speed it up while logging all the details inside the log file. • The log file will contain the entire output to stdout and stderr. It will also contain all the escape sequence codes printed on your shell to colonize it for example. • You can view this log file inside Emacs and use the **pel-screen-log-fix-rendering** command (bound to **<f11> t s**) to filter these escape codes out of the buffer and render the colours. See also: **Buffers** , **Text Modes**
Multi-user screen	Use GNU screen to allow simultaneous access to a shell for several users! See: • GNU Screen Manual - Multiuser Session • https://aperiodic.net/screen/multiuser • Unix & Linux: Sharing a terminal with multiple users (with screen or otherwise) • 2012 UTOSC - Screen vs. tmux faceoff - Jon Jensen - Youtube video