Installing

Jump to bottom

Nicholas Marriott edited this page on Nov 29, 2024 · 26 revisions

Installing tmux

Binary packages

Many platforms provide prebuilt packages of tmux, although these are often out of date. Details of the commands to discover and install these can be found in the documentation for the platform package management tools, for example:

Platform	Install Command
Arch Linux	`pacman -S tmux`
Debian or Ubuntu	`apt install tmux`
Fedora	`dnf install tmux`
RHEL or CentOS	`yum install tmux`
macOS (using Homebrew)	`brew install tmux`
macOS (using MacPorts)	`port install tmux`
openSUSE	`zypper install tmux`

Some thirdparty binary packages are available: AppImage and RPMs.

From source tarball

tmux requires two libraries to be available:

1. libevent
2. ncurses

In addition, tmux requires a C compiler, make, yacc (or bison) and pkg-config.

On most platforms, these are available as packages. This table lists the packages needed to run or to buld tmux:

Platform	Command	Run Packages	Build Packages
Debian	`apt-get install`	`libevent ncurses`	`libevent-dev ncurses-dev build-essential bison pkg-config`
RHEL or CentOS	`yum install`	`libevent ncurses`	`libevent-devel ncurses-devel gcc make bison pkg-config`

If libevent and ncurses are not available as packages, they can be built from source, see this section.

tmux uses autoconf so it provides a configure script. To build and install into `/usr/local` using sudo, run:

```
tar -zxf tmux-*.tar.gz
cd tmux-*/
./configure
make && sudo make install
```

To install elsewhere add `--prefix` to configure, for example for `/usr` add `--prefix=/usr`.

Building dependencies

If the dependencies are not available, they can be built from source and installed locally. This is not recommended if the dependencies can be installed from system packages.

Building requires a C compiler, make, automake, autoconf and pkg-config to be installed. It is more common to need to build libevent than ncurses.

Full instructions can be found on the project sites but this is a summary of how to install libevent and ncurses into `~/local` for a single user. To install system-wide into directories under `/opt` or into `/usr/local`, substitute the required path for for `$HOME/local` in each case and run `make install` as root (for example with sudo: `make && sudo make install`).

For libevent:

```
tar -zxf libevent-*.tar.gz
cd libevent-*/
```

```
./configure --prefix=$HOME/local --enable-shared
make && make install
```

For ncurses:

```
tar -zxf ncurses-*.tar.gz
cd ncurses-*/
./configure --prefix=$HOME/local --with-shared --with-termlib --enable-pc-files --
with-pkg-config-libdir=$HOME/local/lib/pkgconfig
make && make install
```

Then the tmux configure script needs to be pointed to the local libraries using `PKG_CONFIG_PATH`:

```
tar -zxf tmux-*.tar.gz
cd tmux-*/
PKG_CONFIG_PATH=$HOME/local/lib/pkgconfig ./configure --prefix=$HOME/local
make && make install
```

If ncurses and libevent were installed in different directories rather than all together in `~/local`, both their `lib/pkgconfig` directories will need to be in `PKG_CONFIG_PATH`, for example:

```
PKG_CONFIG_PATH=/opt/libevent/lib/pkgconfig:/opt/ncurses/lib/pkgconfig ./configu
--prefix=$HOME/local
```

The newly built tmux can be found in `~/local/bin/tmux`.

When tmux is installed locally on Linux, the runtime linker may need to be told where to find the libraries using the `LD_LIBRARY_PATH` environment variable, for example:

```
LD_LIBRARY_PATH=$HOME/local/lib $HOME/local/bin/tmux -V
```

And to view the manual page, `MANPATH` must be set:

```
MANPATH=$HOME/local/share/man man tmux
```

Most users will want to configure these in a shell profile, for example in `.profile` for ksh or `.bash_profile` for bash:

```
export PATH=$HOME/local/bin:$PATH
export LD_LIBRARY_PATH=$HOME/local/lib:$LD_LIBRARY_PATH
export MANPATH=$HOME/local/share/man:$MANPATH
```

From version control

Building tmux from Git has the same dependencies as building from tarball plus also autoconf and automake. Building is the same as from a tarball except first the configure script must be generated. To install into `/usr/local`:

```
git clone https://github.com/tmux/tmux.git
cd tmux
sh autogen.sh
./configure
make && sudo make install
```

Configure options

tmux provides a few configure options:

Option	Description
`--enable-debug`	Build with debug symbols
`--enable-static`	Create a static build
`--enable-utempter`	Use the utempter library if it is installed
`--enable-utf8proc`	Use the utf8proc library if it is installed

Note that `--enable-static` may require static libraries to be installed, for example on RHEL or CentOS the `glibc-static` package is required.

Common problems

configure says: `libevent not found` or `ncurses not found`

The libevent library or its headers are not installed. Make sure the appropriate packages are installed (some platforms split libraries from headers into a `-dev` or `-devel` package).

configure says: `must give --enable-utf8proc or --disable-utf8proc`

macOS's builtin UTF-8 support is very poor, so it is best to use the utf8proc library if possible. Once it is installed, pass `--enable-utf8proc` to configure.

To force tmux to build without utf8proc, use `--disable-utf8proc`.

tmux won't run from `~/local`

On Linux, make sure `LD_LIBRARY_PATH` is set, or try a static build instead (give `--enable-static` to configure).

`autogen.sh` complains about `AM_BLAH` or `PKG_MODULES`

Make sure pkg-config is installed.

configure says: `C compiler cannot create executables`

Either no C compiler (gcc, clang) is installed, or it doesn't work - check there is nothing stupid in `CFLAGS` or `CPPFLAGS`.

The build fails with an error about "conflicting type for `forkpty`"

For static builds, make sure a static libc is available. On RHEL or CentOS the `glibc-static` package is required.

AppImage package

Instructions and scripts on building an AppImage package for tmux are available from Nelson Enzo here. Prebuilt AppImage packages are also available here.

Docker script

A Docker install script is available here.

Red Hat Enterprise Linux / CentOS RPMs

The tmux packages available from the main repositories are often quite out of date, especially for long-term support distributions. RPMs for newer tmux versions can be obtained from here.

For example to set up a repository and install on RHEL8:

```
sudo yum install http://galaxy4.net/repo/galaxy4-release-8-current.noarch.rpm
sudo yum install tmux
```

Or to install an RPM directly on RHEL6:

```
sudo rpm -ivh http://galaxy4.net/repo/RHEL/6/x86_64/tmux-3.1b-2.el6.x86_64.rpm
```

The repository method is recommended to automatically receive future package updates. See this page for more details.

AppImage package

Docker script

Red Hat Enterprise Linux / CentOS RPMs

▸ **Modifier Keys**

▸ **Recipes**

Clone this wiki locally

```
https://github.com/tmux/tmux.wiki.git
```