Monitor processes, CPU and RAM with htop

I've been using **htop** for so long that it's now my go-to tool for the visual representation of key process performance metrics on a server: CPU usage, RAM, Swap, average load and most resource-hungry processes.

htop command for process monitoring

This is how a default **htop** screen looks on a properly configured colour-capable terminal: just run "**htop**" wihtout any parameters.

How To Install htop in Linux

htop is available via [EPEL repository](#) for CentOS/RedHat/Fedora projects:

```
reys@rhel:~ $ yum whatprovides htop
Loaded plugins: fastestmirror, langpacks
Determining fastest mirrors
```

* base: centos.quelquesmots.fr
* epel: mirror.ibcp.fr
* extras: centos.mirrors.proxad.net
* updates: centos.crazyfrogs.org
htop-2.2.0-3.el7.x86_64 : Interactive process viewer
Repo : epel

Once EPEL is activated, you'll be able to just install htop
with yum.

How To Install htop in MacOS

On MacOS I've been using brew to install htop:

greys@maverick:~ $ **brew install htop**

or

greys@maverick:~ $ **brew upgrade htop**
==> Upgrading 1 outdated package:
htop 2.0.2 -> 2.2.0_1
==> Upgrading htop
==> Installing dependencies for htop: ncurses
==> Installing htop dependency: ncurses
==>   Downloading
https://homebrew.bintray.com/bottles/ncurses-6.1.mojave.bottle
.tar.gz
##
########## 100.0%
==> Pouring ncurses-6.1.mojave.bottle.tar.gz
==> Caveats
ncurses is keg-only, which means it was not symlinked into
/usr/local,
because macOS already provides this software and installing
another version in
parallel can cause all kinds of trouble.

If you need to have ncurses first in your PATH run:
echo 'export PATH="/usr/local/opt/ncurses/bin:$PATH"' >>
~/.bash_profile

For compilers to find ncurses you may need to set:

```
export LDFLAGS="-L/usr/local/opt/ncurses/lib"
export CPPFLAGS="-I/usr/local/opt/ncurses/include"

For pkg-config to find ncurses you may need to set:
export PKG_CONFIG_PATH="/usr/local/opt/ncurses/lib/pkgconfig"

==> Summary
🍺 /usr/local/Cellar/ncurses/6.1: 3,869 files, 8.3MB
==> Installing htop
==>                                          Downloading
https://homebrew.bintray.com/bottles/htop-2.2.0_1.mojave.bottl
e.tar.gz
################################################################
########## 100.0%
==> Pouring htop-2.2.0_1.mojave.bottle.tar.gz
==> Caveats
htop requires root privileges to correctly display all running
processes,
so you will need to run `sudo htop`.
You should be certain that you trust any software you grant
root privileges.
==> Summary
🍺 /usr/local/Cellar/htop/2.2.0_1: 11 files, 188KB
Removing: /usr/local/Cellar/htop/2.0.2... (11 files, 185KB)
==> Caveats
==> ncurses
ncurses is keg-only, which means it was not symlinked into
/usr/local,
because macOS already provides this software and installing
another version in
parallel can cause all kinds of trouble.

If you need to have ncurses first in your PATH run:
echo  'export  PATH="/usr/local/opt/ncurses/bin:$PATH"'  >>
~/.bash_profile

For compilers to find ncurses you may need to set:
export LDFLAGS="-L/usr/local/opt/ncurses/lib"
export CPPFLAGS="-I/usr/local/opt/ncurses/include"

For pkg-config to find ncurses you may need to set:
```

```
export PKG_CONFIG_PATH="/usr/local/opt/ncurses/lib/pkgconfig"

==> htop
```

htop requires root privileges to correctly display all running processes,
so you will need to run `sudo htop`.
You should be certain that you trust any software you grant root privileges.

That's it for today. Hope you find **htop command** useful!

See Also

- [htop — official page](#)
- [Unix Commands](#)
- [top command](#)
- [Basic Unix Commands](#)
- [Advanced Unix Commands](#)