在线代理网页

首先介绍一些代理网站及vpn网站或工具



NB155代理

<http://www.nb155.com/>

### 米度代理！在线代理,永久地址:Www.Midu.Me

<http://www.midu.me/>

bestfreevpn

<http://www.bestfreevpn.com/.com>

pptp

二师兄vpn

<http://www.2-vpn1.org/home.action>

installl openssh-server on fedora

yum install openssh-server

# netstat -ant | grep 22

# [SSH](https://help.ubuntu.com/community/SSH)/[OpenSSH](https://help.ubuntu.com/community/SSH/OpenSSH)/[Configuring](https://help.ubuntu.com/community/SSH/OpenSSH/Configuring?action=fullsearch&value=linkto%3A%22SSH%2FOpenSSH%2FConfiguring%22&context=180)

<https://help.ubuntu.com/community/SSH/OpenSSH/Configuring>

sudo apt-get install openssh-server

# Ubuntu Linux: Start / Stop / Restart OpenSSH ( SSH ) Server

<http://www.cyberciti.biz/faq/howto-start-stop-ssh-server/>

$ sudo /etc/init.d/ssh start

$ sudo /etc/init.d/ssh stop

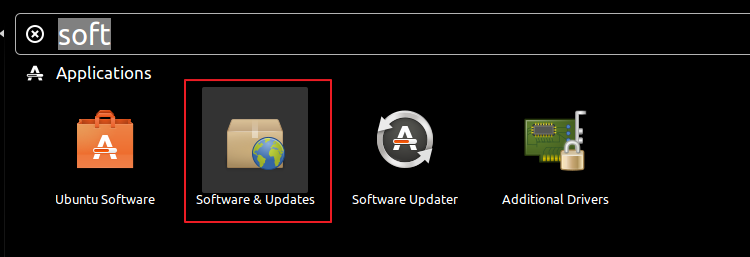
$ sudo /etc/init.d/ssh restart

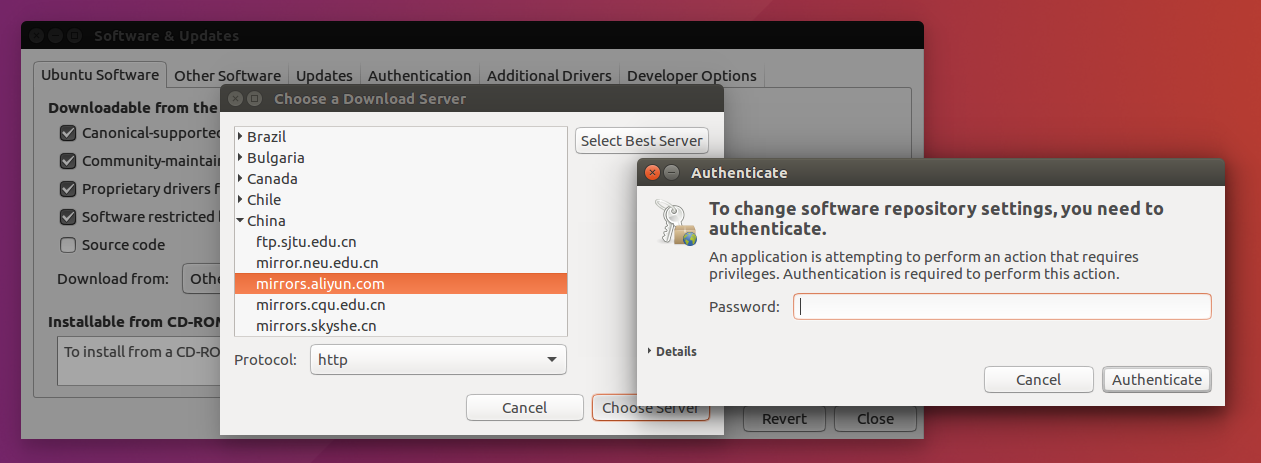
$ sudo /etc/init.d/ssh status

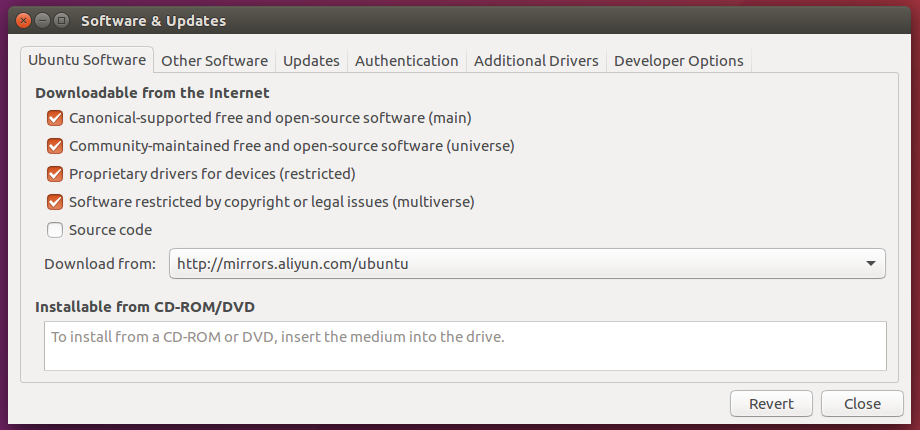
<http://mirrors.aliyun.com/>

<http://mirrors.aliyun.com/help/ubuntu>

(注意：此处只有成功了才能成功执行sudo apt-get install screen等这样的安装操作)

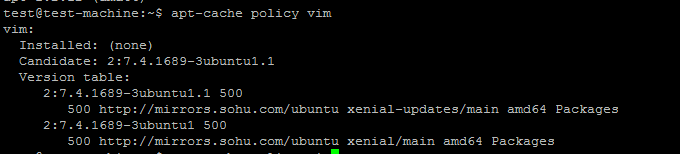






<http://unix.stackexchange.com/questions/6284/check-package-version-using-apt-get-aptitude>

# [Check package version using apt-get/aptitude?](http://unix.stackexchange.com/questions/6284/check-package-version-using-apt-get-aptitude)



**apt-get**

You can run a simulation to see what would happen if you upgrade/install a package:

apt-get -s install <package>

To see all possible upgrades, run a upgrade in verbose mode and (to be safe) with simulation, press n to cancel:

apt-get -V -s upgrade

**apt-cache**

The option policy can show the installed and the remote version (install candidate) of a package.

apt-cache policy <package>

**apt-show-versions**

If installed, shows version information about one or more packages:

apt-show-versions <package>

Passing the -u switch with or without a package name will only show upgradeable packages.

**aptitude**

The console GUI of aptitude can display upgradeable packages with new versions. Open the menu 'Upgradable Packages'. Pressing v on a package will show more detailed version information.

Or on the command-line:

aptitude versions <package>

Passing -V will show detailed information about versions, again to be safe with the simulation switch:

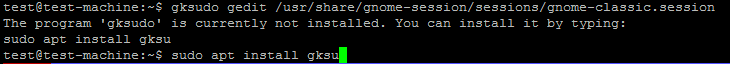
aptitude -V -s install <package>

Substituting install <package> with upgrade will show the versions from all upgradeable packages

<http://askubuntu.com/questions/58172/how-to-revert-to-gnome-classic-desktop>

# [How to revert to GNOME Classic Desktop?](http://askubuntu.com/questions/58172/how-to-revert-to-gnome-classic-desktop)

sudo apt-get install gnome-shell compizconfig-settings-manager gnome-tweak-tool

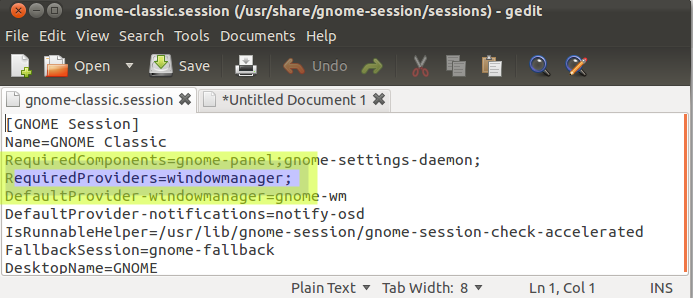


sudo apt install gksu

gksudo gedit /usr/share/gnome-session/sessions/gnome-classic.session

gksudo gedit /usr/share/gnome-session/sessions/gnome.session (本是没找到现有的gnome-classic.session，所以修改的是gnome.session)

输入RequiredProviders=windowmanager;



安装git

sudo apt-get install git



Install screen tools

**sudo apt-get install screen**

screen的使用参考<https://www.digitalocean.com/community/tutorials/how-to-install-and-use-screen-on-an-ubuntu-cloud-server>

# How to Install and Use Screen on an Ubuntu Cloud Server

<http://unix.stackexchange.com/questions/1045/getting-256-colors-to-work-in-tmux>

# [Getting 256 colors to work in tmux](http://unix.stackexchange.com/questions/1045/getting-256-colors-to-work-in-tmux)

<http://stackoverflow.com/questions/6787734/strange-behavior-of-vim-color-inside-screen-with-256-colors>

# [Strange behavior of vim color inside screen with 256 colors](http://stackoverflow.com/questions/6787734/strange-behavior-of-vim-color-inside-screen-with-256-colors)

如下的配置可以解决vim在screen中 颜色不一致的问题。

.screenrc

hardstatus alwayslastline "%{.bW}%-w%{.rW}%n %t%{-}%+w %=%{..G} %H %{..Y} %Y-%m-%d %c"

shell "bash"

startup\_message off

vbell off

altscreen on

attrcolor b ".I"

defbce "on"

termcapinfo xterm\* 'is=\E[r\E[m\E[2J\E[H\E[?7h\E[?1;4;6l'

termcapinfo xterm 'Co#256:AB=\E[48;5;%dm:AF=\E[38;5;%dm'

term screen-256color

解决vim在screen中不能用鼠标拖动来改变split window大小的问题。

<http://superuser.com/questions/549930/cant-resize-vim-splits-inside-tmux>

It appears that dragging the status line to resize a split is not possible when the *Vim* option ttymouseis xterm; it does work when the value is xterm2 though. The latter value configures Vim to ask for an extended mouse reporting mode that (among other things) provides better dragging support. This extended mode only works with newer versions of *xterm* (and other compatible terminal emulators, including *tmux*), so it is not the default value.

You could use something like the following in your .vimrc to set the option: （本人采用的是这种方法,其他方法有缺点）

set mouse+=a

if &term =~ '^screen'

" tmux knows the extended mouse mode

set ttymouse=xterm2

endif

(Though, I am not sure how this will affect actual *screen* instances, which also use a TERM that starts with screen.)

When you are outside of *tmux*, the TERM environment variable is probably an xterm-ish value, and*Vim* will probe for the *xterm* version by using the t\_RV control sequence.

# [Can't resize vim splits inside tmux](http://superuser.com/questions/549930/cant-resize-vim-splits-inside-tmux)

In my case it resolved both cases: mouse split resize and mouse position problem for wide screen.

[The fix is:](http://stackoverflow.com/questions/7000960/vim-mouse-problem)

if has("mouse\_sgr")

set ttymouse=sgr

else

set ttymouse=xterm2

end

# [In Vim, why doesn't my mouse work past the 220th column?](http://stackoverflow.com/questions/7000960/in-vim-why-doesnt-my-mouse-work-past-the-220th-column)

This has been fixed in [Vim 7.3.632](ftp://ftp.vim.org/pub/vim/patches/7.3/7.3.632). See :h sgr-mouse. Or just put this in your ~/.vimrc:

set ttymouse=sgr

If you want to be compatible with versions that don't have mouse\_sgr compiled in, use:

if has("mouse\_sgr")

set ttymouse=sgr

else

set ttymouse=xterm2

end

To see if your version of Vim has mouse\_sgr, run vim --version from the command-line, or in Vim, enter :version, and look for +mouse\_sgr.

在vim中的tabline中显示tab number

<http://vim.wikia.com/wiki/Show_tab_number_in_your_tab_line>

Show tab number in your tab line

You're right. Thank you for the correction, I am no Vim expert at all.

I finally ended up with the .vimrc entry below, borrowed (and simplified) from this [vim\_use thread](http://groups.google.com/group/vim_use/browse_thread/thread/de954baf798fc92c) about "setting Vim tabline".

if exists("+showtabline")

function MyTabLine()

let s = ''

let t = tabpagenr()

let i = 1

while i <= tabpagenr('$')

let buflist = tabpagebuflist(i)

let winnr = tabpagewinnr(i)

let s .= '%' . i . 'T'

let s .= (i == t ? '%1\*' : '%2\*')

let s .= ' '

let s .= i . ')'

let s .= ' %\*'

let s .= (i == t ? '%#TabLineSel#' : '%#TabLine#')

let file = bufname(buflist[winnr - 1])

let file = fnamemodify(file, ':p:t')

if file == ''

let file = '[No Name]'

endif

let s .= file

let i = i + 1

endwhile

let s .= '%T%#TabLineFill#%='

let s .= (tabpagenr('$') > 1 ? '%999XX' : 'X')

return s

endfunction

set stal=2

set tabline=%!MyTabLine()

endif

<http://superuser.com/questions/331272/vim-show-the-index-of-tabs-in-the-tabline>

# [Vim: Show the index of tabs in the tabline](http://superuser.com/questions/331272/vim-show-the-index-of-tabs-in-the-tabline)

put this in your vimrc

" Rename tabs to show tab number.

" (Based on http://stackoverflow.com/questions/5927952/whats-implementation-of-vims-default-tabline-function)

if exists("+showtabline")

function! MyTabLine()

let s = ''

let wn = ''

let t = tabpagenr()

let i = 1

while i <= tabpagenr('$')

let buflist = tabpagebuflist(i)

let winnr = tabpagewinnr(i)

let s .= '%' . i . 'T'

let s .= (i == t ? '%1\*' : '%2\*')

let s .= ' '

let wn = tabpagewinnr(i,'$')

let s .= '%#TabNum#'

let s .= i

" let s .= '%\*'

let s .= (i == t ? '%#TabLineSel#' : '%#TabLine#')

let bufnr = buflist[winnr - 1]

let file = bufname(bufnr)

let buftype = getbufvar(bufnr, 'buftype')

if buftype == 'nofile'

if file =~ '\/.'

let file = substitute(file, '.\*\/\ze.', '', '')

endif

else

let file = fnamemodify(file, ':p:t')

endif

if file == ''

let file = '[No Name]'

endif

let s .= ' ' . file . ' '

let i = i + 1

endwhile

let s .= '%T%#TabLineFill#%='

let s .= (tabpagenr('$') > 1 ? '%999XX' : 'X')

return s

endfunction

set stal=2

set tabline=%!MyTabLine()

set showtabline=1

highlight link TabNum Special

endif

## Putty文件夹蓝色太暗问题

<http://blog.sina.com.cn/s/blog_74a7e56e0101bzkk.html>

 (2013-12-30 13:52:44)

用Putty通过ssh登陆Linux服务器时，有时候会发现系统默认的蓝色字体太暗，具体解决方法如下：

(1)打开putty客户端，选择某登陆Session，然后load；

(2)选择左边选项卡Colours，然后选中下拉列表中的ANSI Blue；

(3)修改其RGB value的值为85, 85, 255；使其与ANSI Blue Bold的颜色一致；

(4)返回Session页，save即可。

源码安装vim （当前版本已经为vim8,而apt-get源中却还为7.4版本，所以选择源码安装）

<https://github.com/Valloric/YouCompleteMe/wiki/Building-Vim-from-source>

# [Valloric](https://github.com/Valloric)/[YouCompleteMe](https://github.com/Valloric/YouCompleteMe)

Compiling Vim from source is actually not that difficult. Here's what you should do:

1：First, install all the prerequisite libraries, including Git. For a Debian-like Linux distribution like Ubuntu, that would be the following:

sudo apt-get install libncurses5-dev libgnome2-dev libgnomeui-dev \

libgtk2.0-dev libatk1.0-dev libbonoboui2-dev \

libcairo2-dev libx11-dev libxpm-dev libxt-dev python-dev \

python3-dev ruby-dev git

(If you know what languages you'll be using, feel free to leave out packages you won't need, e.g. Python2 python-dev or Ruby ruby-dev. This principle heavily applies to the whole page.)

2: Remove vim if you have it already.

sudo apt-get remove vim vim-runtime gvim

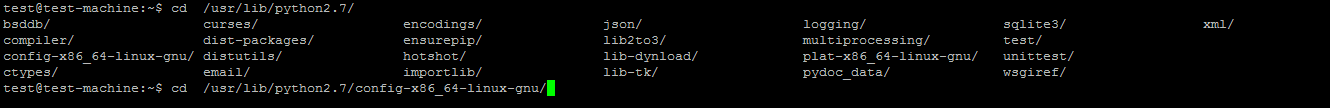
On Ubuntu 12.04.2 you probably have to remove these packages as well:

sudo apt-get remove vim-tiny vim-common vim-gui-common vim-nox

3: Once everything is installed, getting the source is easy.

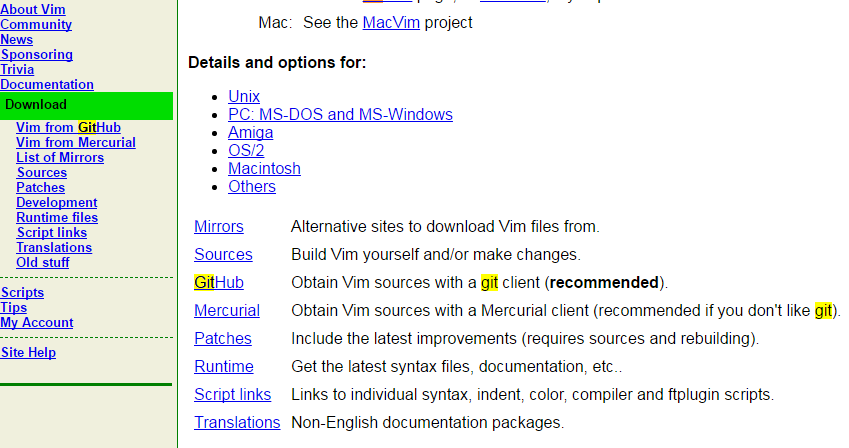
Note: If you are using Python, your config directory might have a machine-specific name (e.g.config-3.5m-x86\_64-linux-gnu). Check in /usr/lib/python[2/3/3.5] to find yours, and change the python-config-dir and/or python3-config-dir arguments accordingly.

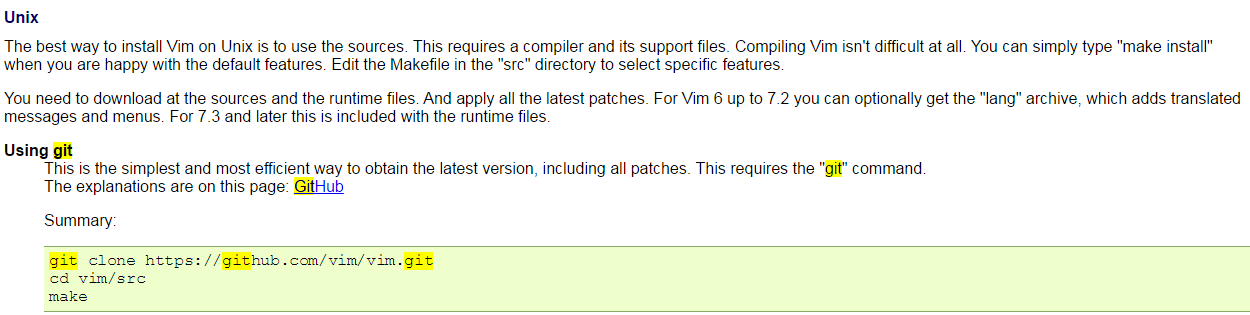
Also, if you're not using vim 7.4, make sure to set the VIMRUNTIMEDIR variable correctly below (for instance, with vim 7.4a, use /usr/share/vim/vim74a). Keep in mind that some vim installations are located directly inside /usr/share/vim; adjust to fit your system:





<http://www.vim.org/download.php>





cd /usr/local/src

sudo git clone https://github.com/vim/vim.git

cd vim

sudo ./configure --with-features=huge \

--enable-multibyte \

--enable-rubyinterp \

--enable-pythoninterp \

--with-python-config-dir=/usr/lib/python2.7/config-x86\_64-linux-gnu \

--enable-python3interp \

--with-python3-config-dir=/usr/lib/python3.5/config-3.5m-x86\_64-linux-gnu \

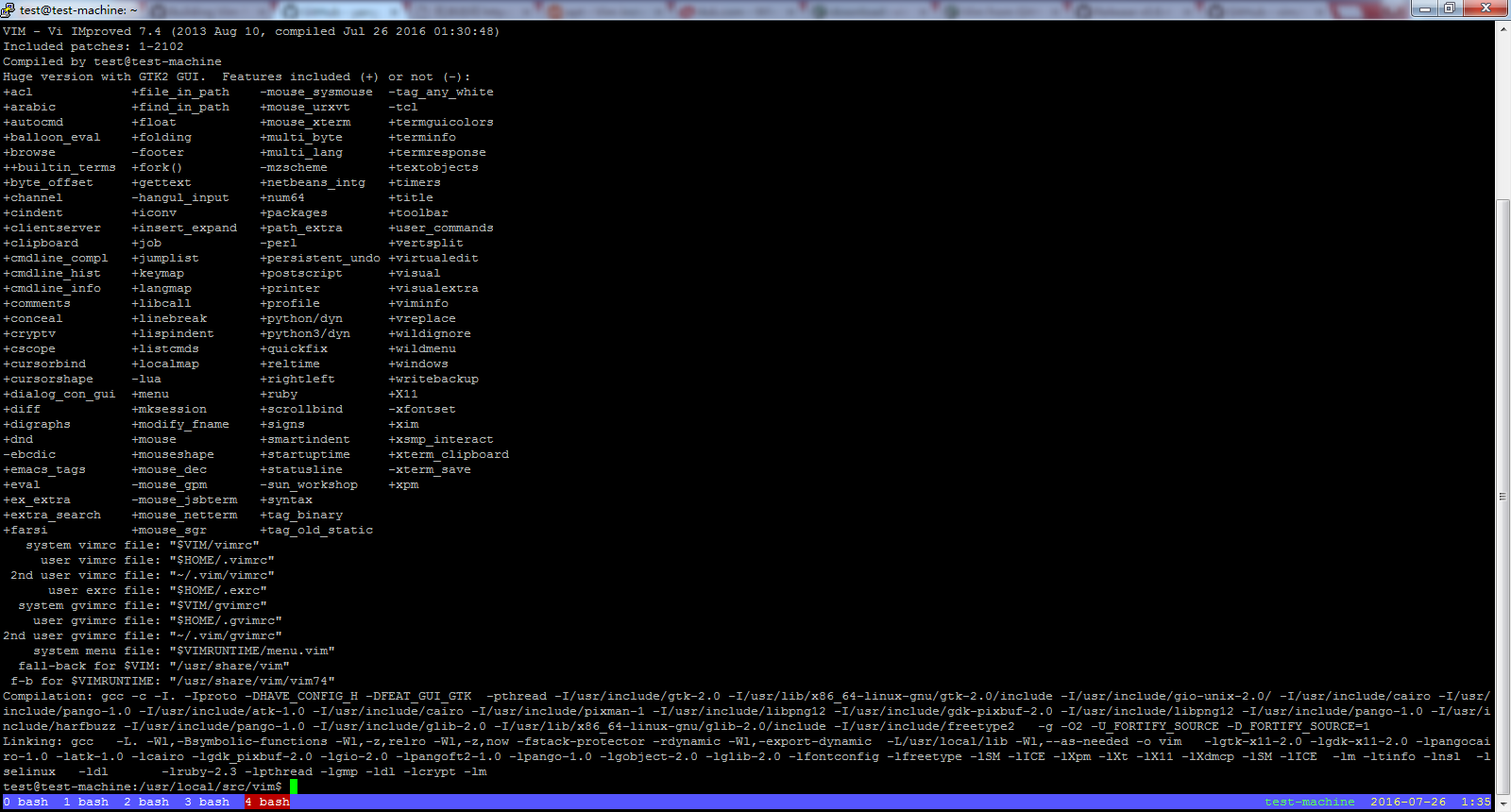
--enable-perlinterp \

--enable-luainterp \

--enable-gui=gtk2 --enable-cscope --prefix=/usr

sudo make VIMRUNTIMEDIR=/usr/share/vim/vim74

sudo make install



# 安装[VundleVim](https://github.com/VundleVim)/[Vundle.vim](https://github.com/VundleVim/Vundle.vim)****插件****

Set up [Vundle](http://github.com/VundleVim/Vundle.vim):

$ git clone https://github.com/VundleVim/Vundle.vim.git ~/.vim/bundle/Vundle.vim

Configure Plugins:：

Put this at the top of your .vimrc to use Vundle. Remove plugins you don't need, they are for illustration purposes.

set nocompatible " be iMproved, required

filetype off " required

" set the runtime path to include Vundle and initialize

set rtp+=~/.vim/bundle/Vundle.vim

call vundle#begin()

" alternatively, pass a path where Vundle should install plugins

"call vundle#begin('~/some/path/here')

" let Vundle manage Vundle, required

Plugin 'VundleVim/Vundle.vim'

" The following are examples of different formats supported.

" Keep Plugin commands between vundle#begin/end.

" plugin on GitHub repo

Plugin 'tpope/vim-fugitive'

" plugin from http://vim-scripts.org/vim/scripts.html

Plugin 'L9'

" Git plugin not hosted on GitHub

Plugin 'git://git.wincent.com/command-t.git'

" git repos on your local machine (i.e. when working on your own plugin)

Plugin 'file:///home/gmarik/path/to/plugin'

" The sparkup vim script is in a subdirectory of this repo called vim.

" Pass the path to set the runtimepath properly.

Plugin 'rstacruz/sparkup', {'rtp': 'vim/'}

" Install L9 and avoid a Naming conflict if you've already installed a

" different version somewhere else.

Plugin 'ascenator/L9', {'name': 'newL9'}

" All of your Plugins must be added before the following line

call vundle#end() " required

filetype plugin indent on " required

" To ignore plugin indent changes, instead use:

"filetype plugin on

"

" Brief help

" :PluginList - lists configured plugins

" :PluginInstall - installs plugins; append `!` to update or just :PluginUpdate

" :PluginSearch foo - searches for foo; append `!` to refresh local cache

" :PluginClean - confirms removal of unused plugins; append `!` to auto-approve removal

"

" see :h vundle for more details or wiki for FAQ

" Put your non-Plugin stuff after this line

Install Plugins:

Launch vim and run :PluginInstall

To install from command line: vim +PluginInstall +qall

tagbar报错的问题的解决办法

<http://stackoverflow.com/questions/7454796/taglist-exuberant-ctags-not-found-in-path>

# [Taglist: Exuberant ctags not found in PATH](http://stackoverflow.com/questions/7454796/taglist-exuberant-ctags-not-found-in-path)

For Ubuntu and derivatives:

sudo apt-get install exuberant-ctags

With yum:

sudo yum install ctags-etags

jshint2.vim插件需要先（依赖）安装node和jshint

<https://github.com/creationix/nvm>

# [creationix](https://github.com/creationix)/[nvm](https://github.com/creationix/nvm) ****（node version manager）****

curl -o- https://raw.githubusercontent.com/creationix/nvm/v0.31.3/install.sh | bash

source ~/.bashrc

test@test-machine:~$ nvm install v0.10.32

test@test-machine:~$ which node

/home/test/.nvm/v0.10.32/bin/node

**使用npm安装一些包失败了的看过来（npm国内镜像介绍）**

 发布于 4 年前  作者 [wppept](https://cnodejs.org/user/wppept)  129790 次浏览  最后一次编辑是 2 年前

这个也是网上搜的，亲自试过，非常好用！

镜像使用方法（三种办法任意一种都能解决问题，建议使用第三种，将配置写死，下次用的时候配置还在）:

1.通过config命令

npm config set registry https://registry.npm.taobao.org

npm info underscore （如果上面配置正确这个命令会有字符串response）

2.命令行指定

npm --registry https://registry.npm.taobao.org info underscore

3.编辑 ~/.npmrc 加入下面内容

registry = https://registry.npm.taobao.org

<http://jshint.com/install/>

sudo npm install -g jshint

### Plugins for text editors and IDEs

#### VIM

* [jshint.vim](https://github.com/walm/jshint.vim), VIM plugin and command line tool for running JSHint.
* [jshint2.vim](https://github.com/Shutnik/jshint2.vim), modern VIM plugin with extra features for running JSHint.
* [Syntastic](https://github.com/scrooloose/syntastic), supports JSHint both older/newer than 1.1.0.

# [ternjs](https://github.com/ternjs)/[tern\_for\_vim](https://github.com/ternjs/tern_for_vim)

## Installation

#### Manual

If you use [Pathogen](https://github.com/tpope/vim-pathogen) or something similar, you can clone this repository to your ~/.vim/bundle (or equivalent) directory. Make sure you have [node.js](http://nodejs.org/) and [npm](https://npmjs.org/) installed (Tern is a JavaScript program), and install the tern server by running npm install in the bundle/tern\_for\_vim directory.

**Caution**: Because the node process is not run using your standard shell, the NVM version of node.js won't work. You need a global node executable.

<http://www.tuicool.com/articles/QBnAr2>

# Vim 自动补全插件 YouCompleteMe 安装与配置

<http://stackoverflow.com/questions/14896327/ultisnips-and-youcompleteme>

# [UltiSnips and YouCompleteMe](http://stackoverflow.com/questions/14896327/ultisnips-and-youcompleteme)

Try this suggestion on a [page](https://github.com/Valloric/YouCompleteMe/issues/36) from the YouCompleteMe issue tracker. In your .vimrc:

let g:UltiSnipsExpandTrigger="<c-j>"

While this setting will make expanding a snippet **share the default mapping for jumping forward within a snippet**, it simulates TextMates' behavior as mentioned in the UltiSnips help tags.

Since I've mapped my Caps Lock key to Ctrl, this mapping works pretty smoothly.

<https://github.com/Valloric/YouCompleteMe/issues/36>

# [Valloric](https://github.com/Valloric)/[YouCompleteMe](https://github.com/Valloric/YouCompleteMe)

# SnipMate/UltiSnips integration #36

[**drasill**](https://github.com/drasill)commented [on 8 Feb 2013](https://github.com/Valloric/YouCompleteMe/issues/36#issuecomment-13284492)

Although, binding <C-j> to ultisnips make it easy to use again.

@Valloric[**Valloric**](https://github.com/Valloric) referenced this issue [on 10 Feb 2013](https://github.com/Valloric/YouCompleteMe/issues/36#ref-issue-10680658)

**Closed**

#### [Intergrate with SnipMate? #47](https://github.com/Valloric/YouCompleteMe/issues/47)