Pasting code with syntax coloring in emails

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

## Creating html[Edit](http://vim.wikia.com/wiki/Pasting_code_with_syntax_coloring_in_emails?action=edit&section=1)

In gvim, use "Convert to HTML" on the Syntax menu. A new file is created, containing the html equivalent of all text from the current buffer.

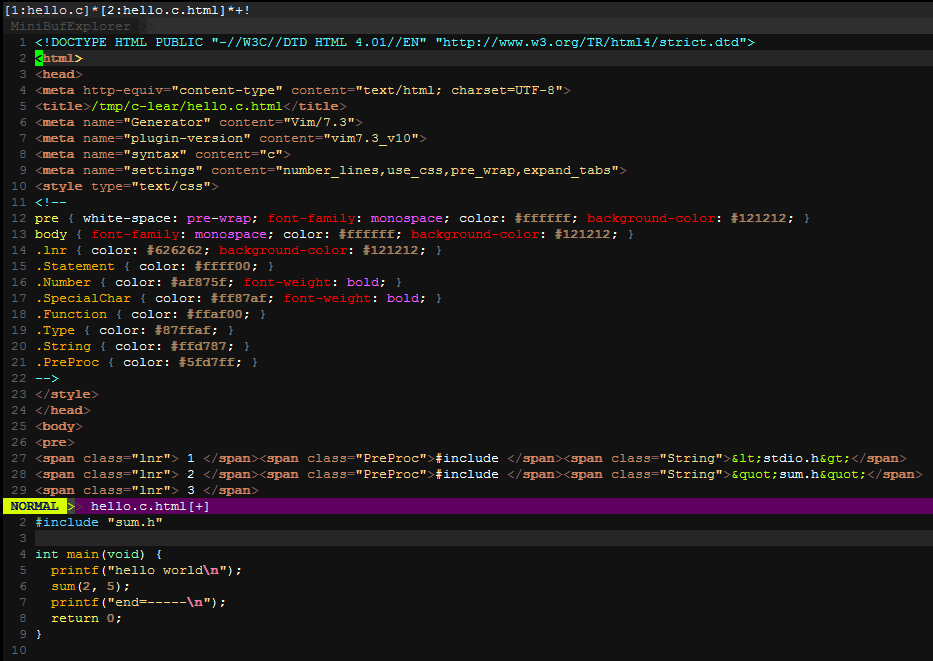
The same operation can be performed by entering a command:

:TOhtml

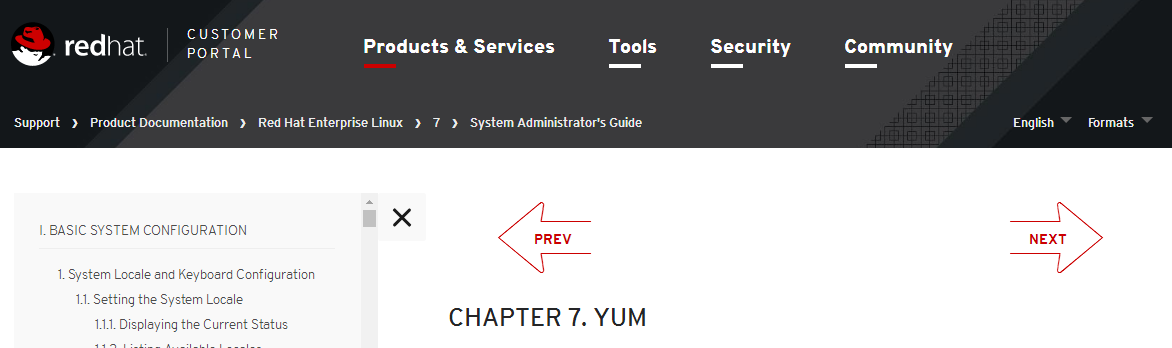
If you visually select some lines before entering :TOhtml, only the selected lines will be converted.

If your color scheme uses a dark background, you may want to temporarily switch to a white background before creating the HTML document. Copy-paste of rich text from the browser into an email client or other application often discards the background color. One way to do that is to use the command:

:colorscheme default



<https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/System_Administrators_Guide/ch-yum.html>



# 20 Linux YUM (Yellowdog Updater, Modified) Commands for Package Management

<http://www.tecmint.com/20-linux-yum-yellowdog-updater-modified-commands-for-package-mangement/>

<https://nazimkuet.wordpress.com/2015/08/26/30-things-to-do-after-minimal-rhelcentos-7-installation/>

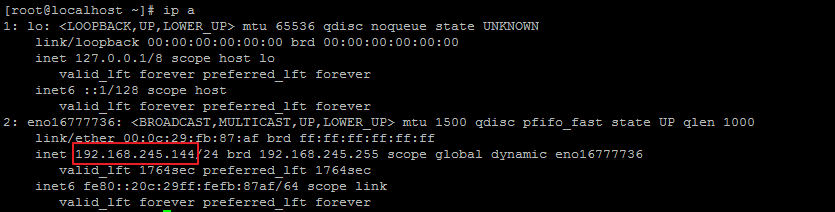
# *30 Things to Do After Minimal RHEL/CentOS 7 Installation*

<http://www.tecmint.com/things-to-do-after-minimal-rhel-centos-7-installation/>

# 30 Things to Do After Minimal RHEL/CentOS 7 Installation

查看ip地址：

ip a



service network restart #启动网路

### 9. Install and Configure SSH Server

<http://www.tecmint.com/things-to-do-after-minimal-rhel-centos-7-installation/2#C9>

**SSH** stands for Secure Shell which is the default protocol in Linux for remote management. SSH is one of those essential piece of software which comes default with CentOS Minimal Server.

Check Currently Installed SSH version.



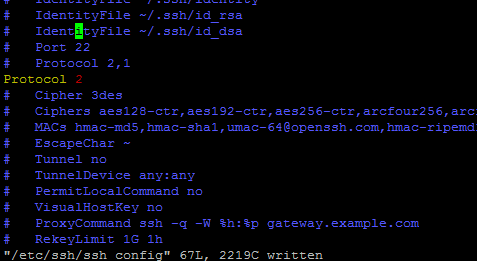
Use Secure Protocol over the default SSH Protocol and change port number also for extra Security. Edit the SSH configuration file ‘/etc/ssh/ssh\_config‘.

Uncomment the line below line or delete 1 from the Protocol string, so the line seems like:

Uncomment the line below line or delete 1 from the Protocol string, so the line seems like:

# Protocol 2,1 (Original)

Protocol 2 (Now)

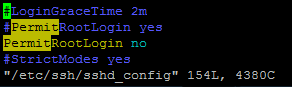


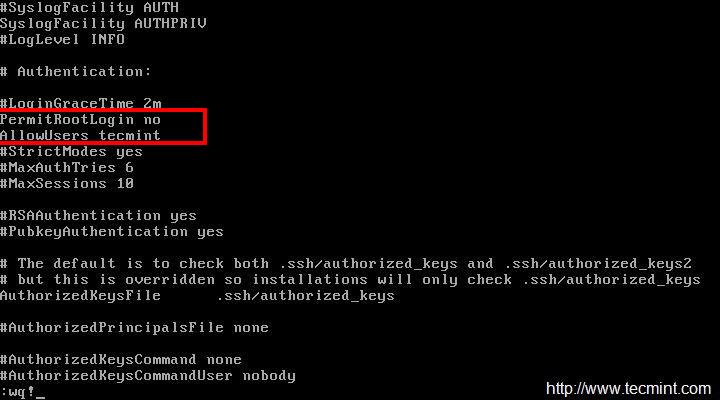
This change force SSH to use Protocol 2 which is considered to be more secure than Protocol 1 and also make sure to change the port number 22 to any in the configuration.

Disable SSH ‘root login‘ and allow to connect to root only after login to normal user account for added additional Security. For this, open and edit configuration file ‘/etc/ssh/sshd\_config‘ and change PermitRootLogin yes tPermitRootLogin no.

# PermitRootLogin yes (Original)

PermitRootLogin no (Now)



[](http://www.tecmint.com/wp-content/uploads/2015/04/Disable-SSH-Root-Login.jpeg)

*Disable SSH Root Login*

Finally, restart SSH service to reflect new changes..

# systemctl restart sshd.service

/bin/systemctl restart sshd.service #启动ssh服务

## 将Centos的yum源更换为国内的阿里云源

<http://www.centoscn.com/CentOS/config/2015/0416/5189.html>

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

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

### 1、备份

mv /etc/yum.repos.d/CentOS-Base.repo /etc/yum.repos.d/CentOS-Base.repo.backup

### 2、下载新的CentOS-Base.repo 到/etc/yum.repos.d/

#### CentOS 5

wget -O /etc/yum.repos.d/CentOS-Base.repo http://mirrors.aliyun.com/repo/Centos-5.repo

#### CentOS 6

wget -O /etc/yum.repos.d/CentOS-Base.repo http://mirrors.aliyun.com/repo/Centos-6.repo

#### CentOS 7

wget -O /etc/yum.repos.d/CentOS-Base.repo http://mirrors.aliyun.com/repo/Centos-7.repo

### 3、之后运行yum makecache生成缓存

yum clean all

yum makecache

yum install epel-release

将epel源设置为阿里云的epel镜像源

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

### 1、备份(如有配置其他epel源)

mv /etc/yum.repos.d/epel.repo /etc/yum.repos.d/epel.repo.backup

mv /etc/yum.repos.d/epel-testing.repo /etc/yum.repos.d/epel-testing.repo.backup

### 2、下载新repo 到/etc/yum.repos.d/

#### epel(RHEL 7)

wget -O /etc/yum.repos.d/epel.repo http://mirrors.aliyun.com/repo/epel-7.repo

#### epel(RHEL 6)

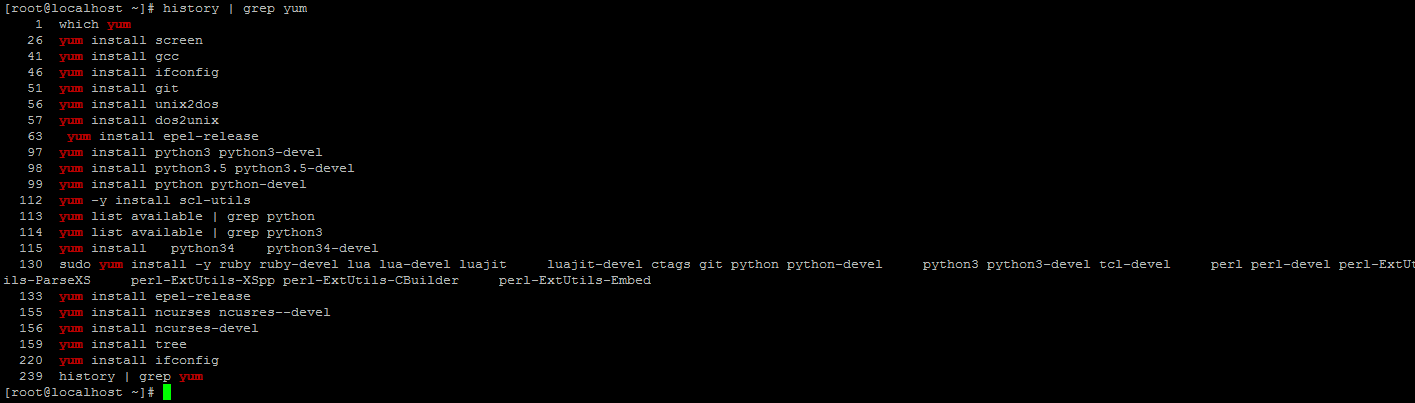
wget -O /etc/yum.repos.d/epel.repo http://mirrors.aliyun.com/repo/epel-6.repo

#### epel(RHEL 5)

wget -O /etc/yum.repos.d/epel.repo http://mirrors.aliyun.com/repo/epel-5.repo

yum makecache

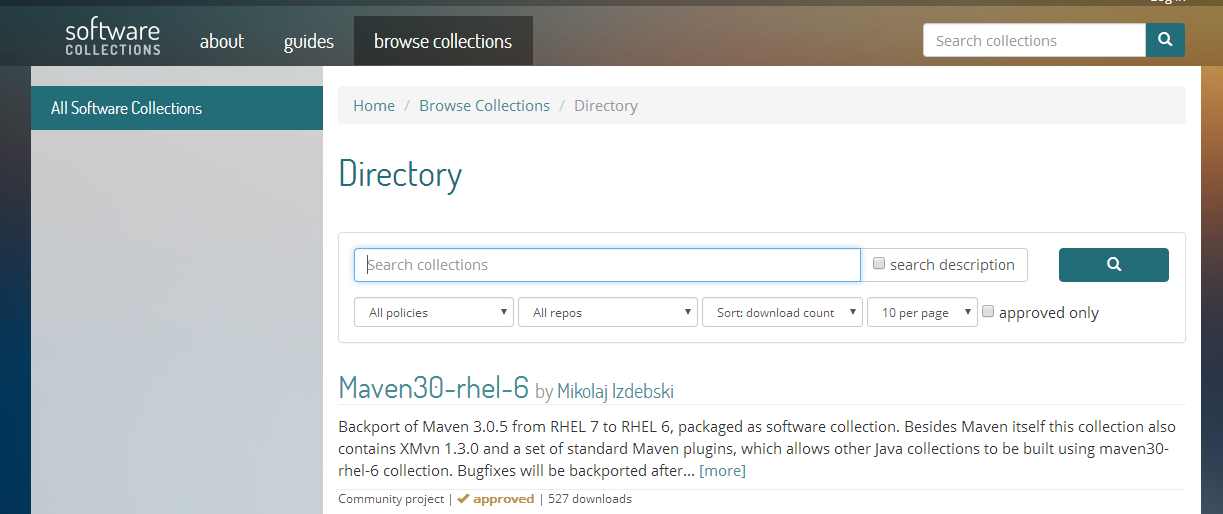
安装某些软件软件：



<https://devops.profitbricks.com/tutorials/install-python-3-in-centos-7/>

[SoftwareCollections.org](https://www.softwarecollections.org/) is a community project built around Red Hat Enterprise Linux, Fedora, CentOS, and Scientific Linux that allows installing and running multiple versions of software on a single system. The available collections typically include newer versions of programming languages and services.

<https://www.softwarecollections.org/en/scls/>



通过鼠标在centos的terminal中的vim程序copy操作（可能有误，以实践为准）：

普通copy: shift + left mouse cursor drag -> right click mouse cursor -> copy menu

块block矩形区域copy: shift + ctrl + left mouse cursor drag -> right click mouse cursor -> copy menu

（注：在centos的terminal中的vim窗口中，如果copy到其他地方后，记得用相同的按键+鼠标键再点击一下来取消选中）

windows系统putty中vim的鼠标copy：

普通copy: shift + left mouse cursor drag

块block矩形区域copy: alt + left mouse cursor drag

<https://www.centos.org/docs/5/html/5.1/Deployment_Guide/index.html>

# Red Hat Enterprise Linux 5.1

## Red Hat Enterprise Linux Deployment Guide

首先介绍一些代理网站及vpn网站或工具



NB155代理

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

bestfreevpn

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

二师兄vpn

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

yum install openssh-server

## 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。

Install screen tools

yum install screen

.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

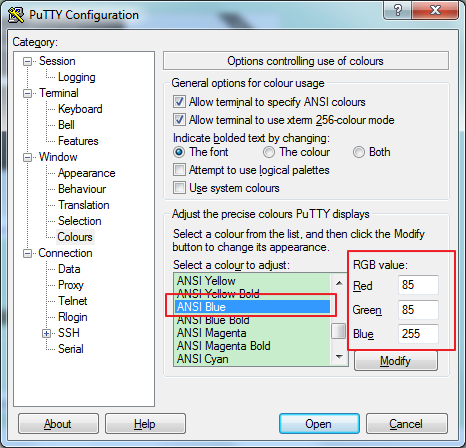
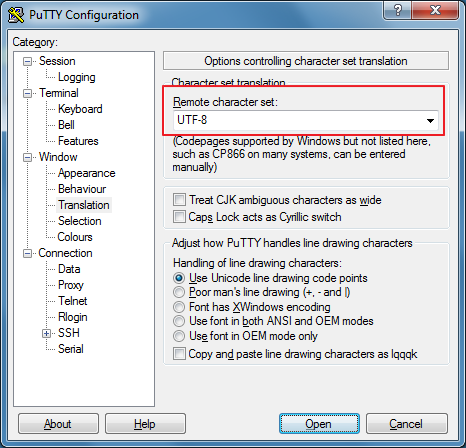
attrcolor b ".I"

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

<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

**nodejs安装**

**step 01: 先安装工具**

<https://linuxmoz.com/centos-install-build-essentials/>

<https://linuxmoz.com/centos-install-development-tools/>

<https://psvmware.wordpress.com/2011/05/13/yum-install-build-essential-centos/>

**//提示：注意网络畅通，可以使用vpn连接保持国外网络通畅**

yum install gcc gcc-c++ kernel-devel

**或者：**

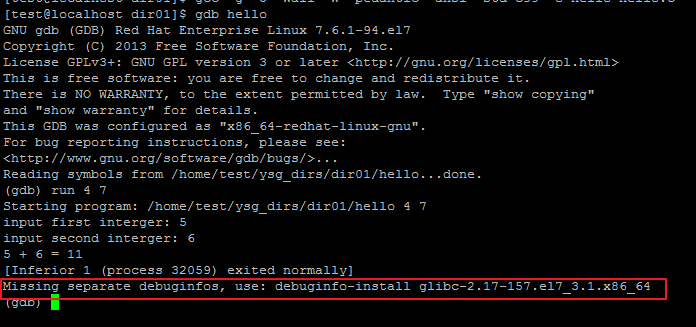
yum groupinstall"Development Tools"

**如果要调试，可以安装gdb**

**[root@localhost ~]# yum install gdb**

[**http://stackoverflow.com/questions/10389988/missing-separate-debuginfos-use-debuginfo-install-glibc-2-12-1-47-el6-2-9-i686**](http://stackoverflow.com/questions/10389988/missing-separate-debuginfos-use-debuginfo-install-glibc-2-12-1-47-el6-2-9-i686)

# [Missing separate debuginfos, use: debuginfo-install glibc-2.12-1.47.el6\_2.9.i686 libgcc-4.4.6-3.el6.i686 libstdc++-4.4.6-3.el6.i686](http://stackoverflow.com/questions/10389988/missing-separate-debuginfos-use-debuginfo-install-glibc-2-12-1-47-el6-2-9-i686)



debuginfo-install is a command of yum-utils, so

1. yum install yum-utils
2. debuginfo-install glibc
3. if the warning's still there, edit /etc/yum.repos.d/CentOS-Debuginfo.repo, set enabled=1

<https://www.digitalocean.com/community/tutorials/how-to-install-and-run-a-node-js-app-on-centos-6-4-64bit>

./configure

make

make install

# yum -y install dos2unix

# yum -y install unix2dos

**安装git（centos7下采用如下此种方式）**

### [Installing from Source](https://git-scm.com/book/en/v2/Getting-Started-Installing-Git#Installing-from-Source)

<https://git-scm.com/book/en/v2/Getting-Started-Installing-Git>

**$** sudo yum install curl-devel expat-devel gettext-devel **\**

openssl-devel perl-devel zlib-devel

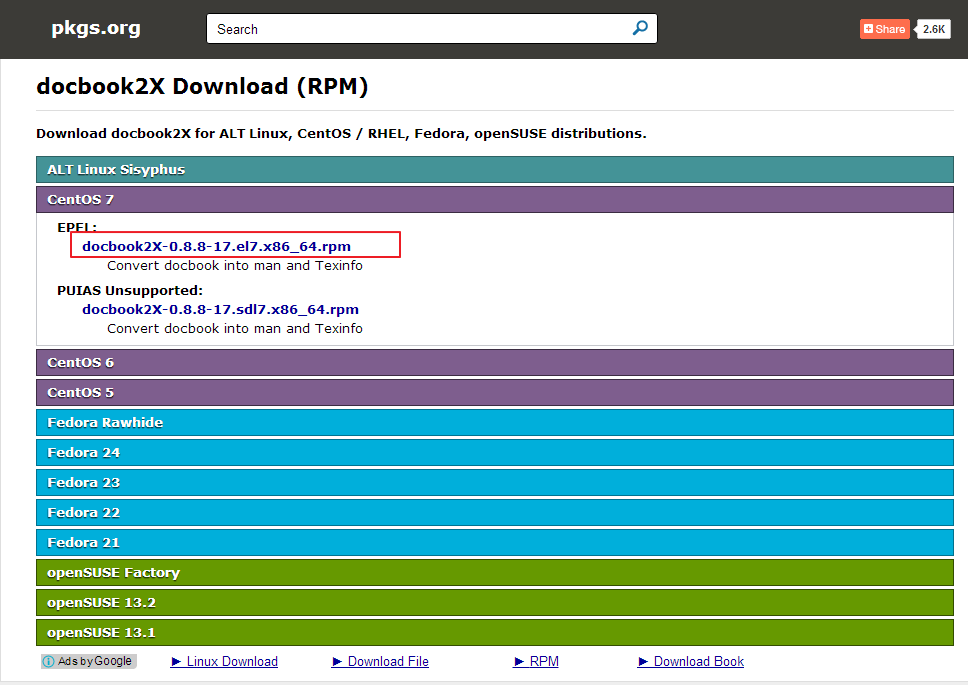
**$** sudo apt-get install libcurl4-gnutls-dev libexpat1-dev gettext **\**

libz-dev libssl-dev

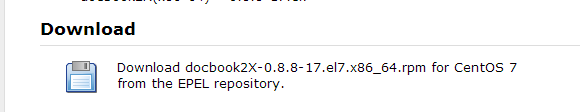
In order to be able to add the documentation in various formats (doc, html, info), these additional dependencies are required (Note: users of RHEL and RHEL-derivatives like CentOS and Scientific Linux will have to [enable the EPEL repository](https://fedoraproject.org/wiki/EPEL#How_can_I_use_these_extra_packages.3F) to download the docbook2X package):

**$** sudo yum install asciidoc xmlto docbook2X #centos7中无法用此方式安装docbook2X,需要自己下载**[docbook2X-0.8.8-17.el7.x86\_64.rpm](http://dl.fedoraproject.org/pub/epel/7/x86_64/d/docbook2X-0.8.8-17.el7.x86_64.rpm)**来进行安装

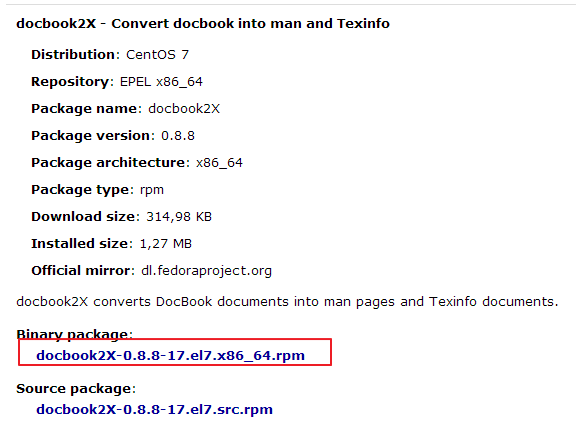
<https://pkgs.org/download/docbook2X>



<https://pkgs.org/centos-7/epel-x86_64/docbook2X-0.8.8-17.el7.x86_64.rpm.html>



<https://pkgs.org/centos-7/epel-x86_64/docbook2X-0.8.8-17.el7.x86_64.rpm/download/>



**下载后执行命令**：yum install docbook2X-0.8.8-17.el7.x86\_64.rpm

**$** sudo apt-get install asciidoc xmlto docbook2x

Additionally, if you’re using Fedora/RHEL/RHEL-derivatives, you need to do this

**$** sudo ln -s /usr/bin/db2x\_docbook2texi /usr/bin/docbook2x-texi

**$** tar -zxf git-2.9.3.tar.gz

**$** cd git-2.9.3

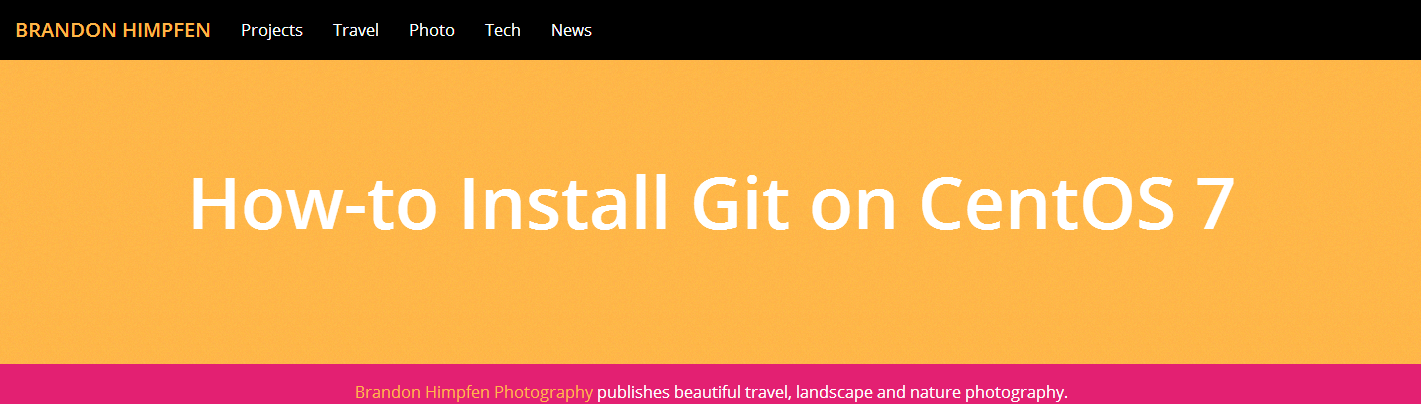
**$** make configure

如遇到下面问题：



解决办法：

<http://www.himpfen.com/install-git-centos-7/>



yum install autoconf #然后再次执行make configure

**$** ./configure --prefix=/usr

**$** make all doc info

**$** sudo make install install-doc install-html install-info

OK,成功

**安装git （此处已经作废）**

<http://stackoverflow.com/questions/21820715/how-to-install-latest-version-of-git-on-centos-6-x>

<http://tecadmin.net/install-git-2-0-on-centos-rhel-fedora/>

<http://www.howtoforge.com/how-to-install-the-latest-git-version-on-centos>

**Step 1:** Install Required Packages

# yum install curl-devel expat-devel gettext-devel openssl-devel zlib-devel

# yum install gcc perl-ExtUtils-MakeMaker

**Step 2:** Uninstall old Git RPM

# yum remove git

**Step 3:** Download and Compile Git Source

# cd /usr/src

# wget https://www.kernel.org/pub/software/scm/git/git-2.9.3.tar.gz

# tar zxvf git-2.9.3.tar.gz

After downloading and extracting Git source code, Use following command to compile source code.

# cd git-2.9.3/

# make prefix=/usr/local/git all

# make prefix=/usr/local/git install

# echo "export PATH=$PATH:/usr/local/git/bin" >> /etc/bashrc

# source /etc/bashrc

<http://git-scm.com/book/en/v2/Getting-Started-Installing-Git>

$ cp git-2.2.0.tar.gz /usr/src/

$ cd /usr/src/

$ tar -zxf git-2.0.0.tar.gz

$ cd git-2.0.0

$ make configure

$ ./configure --prefix=/usr

$ yum install asciidoc #避免在执行make all doc info时出现如下错误

/bin/sh: line 1: asciidoc: command not found

make[1]: \*\*\* [git-check-attr.html] Error 127

make[1]: Leaving directory `/usr/src/git-2.2.0/Documentation'

make: \*\*\* [doc] Error 2

$ yum install xmlto #避免在执行make all doc info时出现如下错误

/bin/sh: line 1: xmlto: command not found

make[1]: \*\*\* [git-check-attr.1] Error 127

make[1]: Leaving directory `/usr/src/git-2.2.0/Documentation'

make: \*\*\* [doc] Error 2

<http://stackoverflow.com/questions/897477/installing-git-on-os-x>

$ yum install docbook2X

$ ln -s /usr/bin/db2x\_docbook2texi /usr/bin/docbook2x-texi #避免在执行make all doc info时出现如下错误

/bin/sh: line 1: docbook2x-texi: command not found

make[1]: \*\*\* [user-manual.texi] Error 127

make[1]: Leaving directory `/usr/src/git-2.2.0/Documentation'

make: \*\*\* [info] Error 2

$ yum install epel-release

$ yum install gettext

$ make all doc info

$ make install install-doc install-html install-info

**Step 4.** Check Git Version

# git --version

git version 2.0.4

# Extra Packages for Enterprise Linux (EPEL)

**NOTE for CentOS users**  
You can install EPEL by running **yum install epel-release**. The package is included in the CentOS Extras repository, enabled by default.

If you are running an EL7 version, please visit here to get the newest 'epel-release' package for EL7: [The newest version of 'epel-release' for EL7](http://download.fedoraproject.org/pub/epel/7/x86_64/repoview/epel-release.html) (注意安装最新版本的，因为旧版本的epel-release可能找不到一些软件包,如python34之类)

yum install <https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm>

If you are running an EL6 version, please visit here to get the newest 'epel-release' package for EL6: [The newest version of 'epel-release' for EL6](http://download.fedoraproject.org/pub/epel/6/i386/repoview/epel-release.html)

If you are running an EL5 version, please visit here to get the newest 'epel-release' package for EL5: [The newest version of 'epel-release' for EL5](http://download.fedoraproject.org/pub/epel/5/i386/repoview/epel-release.html)

You can verify these packages and their keys from the Fedora project's keys page: <https://fedoraproject.org/keys>

<http://stackoverflow.com/questions/9500898/compiler-error-msgfmt-command-not-found-when-compiling-git-on-a-shared-hosting>

# [Compiler error - msgfmt command not found when compiling git on a shared hosting](http://stackoverflow.com/questions/9500898/compiler-error-msgfmt-command-not-found-when-compiling-git-on-a-shared-hosting)

I had the same issue. Thanks to your work on finding it was related to gettext, a simple apt-get install gettext fixed it for me.

## [CentOS下Git命令行自动补全](http://ipple.net/lanmp/centos-git-bash-completion.html)

<http://ipple.net/lanmp/centos-git-bash-completion.html>

<http://www.360doc.com/content/13/0116/12/10058718_260492185.shtml>

在《[Git](http://ipple.net/tag/git" \t "_blank" \o "View all posts in Git)权威指南》一书中看到Git可以配置命令自动补齐功能，真不错，不过按照原书的操作说明，查看了一下，CentOS的etc目录下无bash\_completion文件，故没有按照他写的方法操作，而是直接把git提供的bash脚本拷贝到用户的目录下就可以了，具体操作方法：

$ cd git-1.7.7.4  
$ cp contrib/completion/git-completion.bash ~/.git-completion.bash  
$ vi ~/.bashrc

最后一行加入  
source ~/.git-completion.bash

关闭SSH终端，再重新登入，检测一下，Git输入的时候，按tab键，Git命令是不是能自动补全了？

另外Debian/Ubuntu里有个包就叫git-completion，这个包安装完成后会自动把这个补全脚本放到/etc/bash\_completion.d/下，由bash-compleletion载入执行。  
就是前面那本书中提到的方式

[【Git笔记】运行man git时报错"No manual entry for git"的解决方法](http://blog.csdn.net/slvher/article/details/17968499)

<http://blog.csdn.net/slvher/article/details/17968499>

从github下载git源码(version 1.8.5.2)解压，执行以下步骤后安装git成功：  
          a. make configure  
          b. ./configure --prefix=xxx (这里的xxx为自定义的安装路径)  
          c. make  
          d. make install  
          e. 在~/.bash\_profile中将GITPATH/bin加入PATH保存退出后，执行. ~/.bash\_profile

        但运行man git时提示"No manual entry for git"

        这是由于git安装时并不默认安装manual docs，如果想要安装git时同时安装manpages，则需执行make all doc && make install install-doc，而非简单的make && make install

**解决方法：**  
从git-core的[downloadlist](https://code.google.com/p/git-core/downloads/list" \t "_blank)下载与git版本一致的manpages，比如我拉取的是git-manpages-1.8.5.2.tar.gz  
       下载完后执行tar -zxf git-manpages-1.8.5.2.tar.gz解压，将解压结果mv至~/tools/git-1.8.5.2/share/man  
        在~/.bash\_profile中，添加export MANPATH=$HOME/tools/git-1.8.5.2/share/man:$MANPATH后保持退出，执行. ~/.bash\_profile

        然后在shell终端运行man git可以验证结果。

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

# Building Vim from source

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

For Fedora 20, that would be the following:

sudo yum install -y ruby ruby-devel lua lua-devel luajit \

luajit-devel ctags mercurial python python-devel \

python3 python3-devel tcl-devel \

perl perl-devel perl-ExtUtils-ParseXS \

perl-ExtUtils-XSpp perl-ExtUtils-CBuilder \

perl-ExtUtils-Embed

This step is needed to rectify an issue with how Fedora 20 installs XSubPP:

# symlink xsubpp (perl) from /usr/bin to the perl dir

sudo ln -s /usr/bin/xsubpp /usr/share/perl5/ExtUtils/xsubpp (cento7中不需要这一步骤)



## Installing latest Vim on CentOS from source

<http://www.fullybaked.co.uk/articles/installing-latest-vim-on-centos-from-source>

# [No Terminal Library Found when Compiling Vim](http://askubuntu.com/questions/158344/no-terminal-library-found-when-compiling-vim)

I think you should install a ncurses-dev library.

yum install ncurses ncurses-devel #如果不安装ncurses ncurses-devel,后续执行./configure时很可能出现如下错误

no terminal library found

checking for tgetent()... configure: error: NOT FOUND!

You need to install a terminal library; for example ncurses.

Or specify the name of the library with --with-tlib.

# Compiling Vim on Fedora 20

<http://ciarand.me/posts/compiling-vim-on-fedora-20/>

yum remove vim

**Using git**

This is the simplest and most efficient way to obtain the latest version, including all patches. This requires the "git" command.   
The explanations are on this page: [GitHub](http://www.vim.org/git.php)

Summary:

su - root

cd /usr/local/src

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

[root@localhost config]# whereis python2.7



/usr/lib64/python2.7/config

<http://www.csdn.net/article/2014-04-24/2819472-softwarecollections>

# SCL项目：在Red Hat系列系统上安装同一软件的不同版本

<https://devops.profitbricks.com/tutorials/install-python-3-in-centos-7/>

Install Python 3 in CentOS 7

## Introduction

[SoftwareCollections.org](https://www.softwarecollections.org/) is a community project built around Red Hat Enterprise Linux, Fedora, CentOS, and Scientific Linux that allows installing and running multiple versions of software on a single system. The available collections typically include newer versions of programming languages and services.

This tutorial will describe setting up Software Collections (SCLs) in CentOS 7 and installing Python 3.3 along side the Python 2.7 distribution default.

## Requirements

* CentOS 7 or Red Hat Enterprise Linux 7

## Install Software Collections

The first step is to install the utilities needed to manage collections.

yum -y install scl-utils

The next step is to identify which software collection you want and to install the associated repository. A full list of available collections can be found at the following URL.

<https://www.softwarecollections.org/en/scls/>

The **Python33** collection repository package can be installed directly using RPM.

rpm -Uvh https://www.softwarecollections.org/en/scls/rhscl/python33/epel-7-x86\_64/download/rhscl-python33-epel-7-x86\_64.noarch.rpm

<http://serverfault.com/questions/710354/repository-for-python3-devel-on-centos-7>

# [repository for python3-devel on CentOS 7?](http://serverfault.com/questions/710354/repository-for-python3-devel-on-centos-7)

You can always activate the [EPEL Repository](https://fedoraproject.org/wiki/EPEL), it has a [python34-devel package](https://dl.fedoraproject.org/pub/epel/7/x86_64/repoview/python34-devel.html) available.

yum install <https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm>

yum install python34 python34-devel

/usr/lib64/python3.4/config-3.4m

<https://www.softwarecollections.org/en/docs/>

# 

# Quick Start

## Using Software Collections

1. Find a Software Collection (SCL) you are interested in. You can browse through the [available SCLs in our Directory](https://www.softwarecollections.org/en/scls/). We will use the [python33 SCL](https://www.softwarecollections.org/en/scls/rhscl/python33/) as an example.
2. Download and install the chosen Software Collection's repository RPM file for your system (EPEL 6 on AMD 64 or Intel® 64, in our case):

yum install rhscl-python33-epel-6-x86\_64-1-2.noarch.rpm

1. Install the main SCL package (its name is identical to the name of the Software Collection):

yum install python33

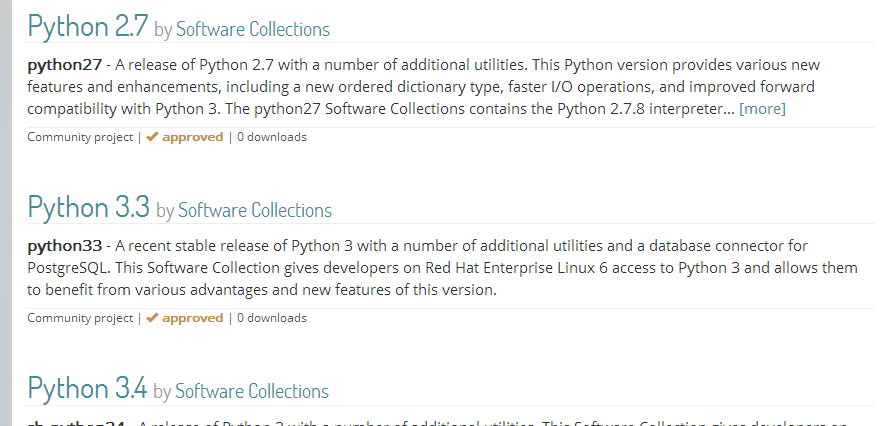
1. Start using the newly installed Software Collection:

scl enable python33 bash

For more usage examples and documentation of more complex scenarios, see the [Software Collections Packaging Guide](https://www.softwarecollections.org/en/docs/guide/#chap-Introducing_Software_Collections).

If you need any help, you can ask your question on StackOverflow using the [software-collections](http://stackoverflow.com/questions/tagged/software-collections) tag.

<https://www.softwarecollections.org/en/scls/?search=python>



cd vim/

./configure --with-features=huge \

--enable-multibyte \

--enable-rubyinterp \

--enable-pythoninterp \

--with-python-config-dir=/usr/lib64/python2.7/config \

--enable-python3interp \

--with-python3-config-dir=/usr/lib64/python3.4/config-3.4m \

--enable-perlinterp \

--enable-luainterp \

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

[root@localhost vim]# yum remove vim



make VIMRUNTIMEDIR=/usr/share/vim/vim74

make install

vundle:

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

<https://nodejs.org/en/download/package-manager/#enterprise-linux-and-fedora>

## Enterprise Linux and Fedora

Run as root on RHEL, CentOS or Fedora, for Node.js v4 LTS Argon:

curl --silent --location https://rpm.nodesource.com/setup\_4.x | bash -

Alternatively for Node.js v6:

curl --silent --location https://rpm.nodesource.com/setup\_6.x | bash -

Alternatively for Node.js 0.10:

curl --silent --location https://rpm.nodesource.com/setup | bash -

Then install, as root:

yum -y install nodejs

jshint2.vim插件需要先（依赖）安装node和jshint

<https://github.com/Shutnik/jshint2.vim>

# [Shutnik](https://github.com/Shutnik)/[jshint2.vim](https://github.com/Shutnik/jshint2.vim)

# jshint2.vim

**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

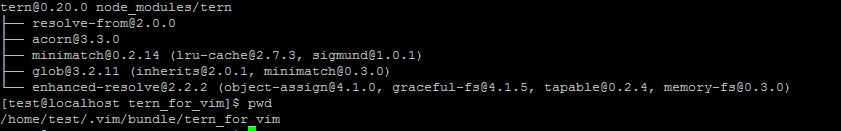
registry = https://registry.npm.taobao.org

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

npm install -g jshint

cd vim/bundle/tern\_for\_vim

npm install



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

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

<https://github.com/Valloric/YouCompleteMe#fedora-linux-x64>

### Fedora Linux x64

Make sure you have Vim 7.3.598 with Python 2 or Python 3 support. Fedora 21 and later have a Vim that's recent enough. You can see the version of Vim installed by running vim --version. If the version is too old, you may need to [compile Vim from source](https://github.com/Valloric/YouCompleteMe/wiki/Building-Vim-from-source) (don't worry, it's easy).

Install YouCompleteMe with [Vundle](https://github.com/VundleVim/Vundle.vim" \l "about).

**Remember:** YCM is a plugin with a compiled component. If you **update** YCM using Vundle and the ycm\_core library APIs have changed (happens rarely), YCM will notify you to recompile it. You should then rerun the install process.

Install development tools and CMake: sudo dnf install automake gcc gcc-c++ kernel-devel cmake

我执行的命令： yum install automake gcc gcc-c++ kernel-devel cmake

Make sure you have Python headers installed: sudo dnf install python-devel python3-devel.

Compiling YCM **with** semantic support for C-family languages:

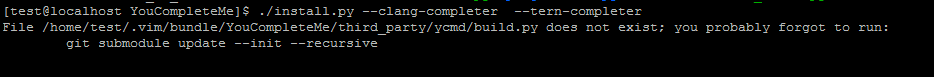
cd ~/.vim/bundle/YouCompleteMe

./install.py --clang-completer --tern-completer

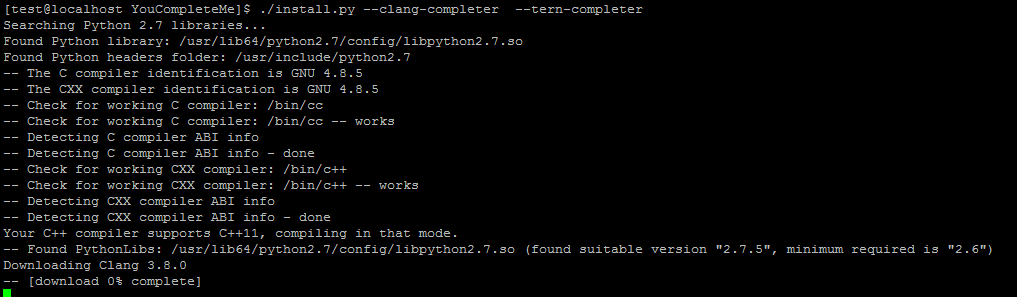
Compiling YCM **without** semantic support for C-family languages:

The following additional language support options are available:

* C# support: install [Mono](http://www.mono-project.com/docs/getting-started/install/linux/#centos-7-fedora-19-and-later-and-derivatives) and add --omnisharp-completer when calling ./install.py.
* Go support: install [Go](https://golang.org/doc/install) and add --gocode-completer when calling ./install.py.
* TypeScript support: install [Node.js and npm](https://docs.npmjs.com/getting-started/installing-node) then install the TypeScript SDK with npm install -g typescript.
* JavaScript support: install [Node.js and npm](https://docs.npmjs.com/getting-started/installing-node) and add --tern-completer when calling ./install.py.
* Rust support: install [Rust](https://www.rust-lang.org/) and add --racer-completer when calling ./install.py.



git submodule update --init --recursive



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

Node Version Manager - Simple bash script to manage multiple active node.js versions

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

source ~/.bashrc

$ wget http://www.my-guides.net/en/images/stories/fedora12/msttcore-fonts-2.0-3.noarch.rpm

$ yum install msttcore-fonts-2.0-3.noarch.rpm

<http://blog.csdn.net/sunylat/article/details/8880550>

# [CentOS中安装Courier New字体](http://blog.csdn.net/sunylat/article/details/8880550)

打开终端，输入：wget <http://www.my-guides.net/en/images/stories/fedora12/msttcore-fonts-2.0-3.noarch.rpm>随后回车，下载此安装包，下载完毕后，找到保存此文件目录，双击此文件，随后就把这些微软字体安装上去了，也包括Courier New字体

<http://www.itzgeek.com/how-tos/linux/centos-how-tos/how-to-install-microsoft-truetype-fonts-in-centos-6-rhel-6.html#axzz3P9SXxTqn>

# How to install Microsoft TrueType Fonts in CentOS 6 / RHEL 6

<http://help.accusoft.com/PCC/v8.2/HTML/How%20to%20Install%20Microsoft%20Fonts%20on%20Linux.html>

**How to Install Microsoft Fonts on Linux**

<http://www.2cto.com/os/201303/198055.html>

<http://raincoder.iteye.com/blog/1670867>

### [主机无法访问虚拟机中Linux上的tomcat服务](http://raincoder.iteye.com/blog/1670867)

  在wmware中安装linux后安装好数据库，JDK及tomcat后启动服务，虚拟机中可以访问，但是主机却无法访问，但是同时主机和虚拟机之间可以ping的通，网上查阅资料后，解决方法是关闭虚拟机中的防火墙服务。桌面--管理--安全级别与防火墙，将防火墙设置为：disable即可。或用root登录后，执行   
service iptables stop --停止   
service iptables start --启动   
  
但是在实际应用中，关闭防火墙降低的服务器的安全性，不能关闭防火墙。   
如果在宿主机的dos窗口下telnet虚拟机的8080窗口，会失败，由此可以确定是虚拟机的8080窗口有问题，应该是被防火墙堵住了。因此修改防火墙设置即可。   
  
修改Linux系统防火墙配置需要修改 /etc/sysconfig/iptables 这个文件，如果要开放哪个端口，在里面添加一条   
-A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp --dport 8080 -j ACCEPT   
就可以了，其中 8080 是要开放的端口号，然后重新启动linux的防火墙服务，   
/etc/init.d/iptables restart。

# [CentOs中iptables配置允许mysql远程访问](http://www.cnblogs.com/ccdc/archive/2012/04/24/2468048.html)

**mysql访问配置：**

iptables -A INPUT -p tcp -m state --state NEW -m tcp --dport 3306 -j ACCEPT

service iptables save

service iptables restart

<http://www.aichengxu.com/view/32778>

# centos6.5下安装mysql

10.CentOS6.5开放mysql端口3306  
CentOS6.5默认是不开放端口的，如果要让外部的系统访问CentOS6.5上的[mysql](http://cpro.baidu.com/cpro/ui/uijs.php?rs=1&u=http%3A%2F%2Fwww%2Eaichengxu%2Ecom%2Fview%2F32778&p=baidu&c=news&n=10&t=tpclicked3_hc&q=50099048_cpr&k=mysql&k0=mysql&kdi0=8&k1=server&kdi1=1&k2=%D2%FD%C7%E6&kdi2=1&k3=memory&kdi3=1&sid=4973828ce4634bc2&ch=0&tu=u1703034&jk=a994250acd9138dd&cf=29&fv=16&stid=9&urlid=0&luki=1&seller_id=1&di=128" \t "_blank)，必须开放mysql的端口3306  
10.1 修改/etc/sysconfig/iptables  
添加下面一行

-A INPUT -m state --state NEW -m tcp -p tcp --dport 3306 -j ACCEPT

/etc/sysconfig/iptables

<http://www.cyberciti.biz/faq/rhel-fedorta-linux-iptables-firewall-configuration-tutorial/>

# CentOS / Redhat Iptables Firewall Configuration Tutorial

<http://stackoverflow.com/questions/8588789/linux-find-and-replace>

# [Linux find and replace](http://stackoverflow.com/questions/8588789/linux-find-and-replace)

Try the following command for the file file.txt:

sed -i 's/abc/abcd/g' file.txt

Try the following command for all files in the current folder:

find . -maxdepth 1 -type f -exec sed -i 's/abc/abcd/g' {} \;

For the files in the current directory and all subdirectories:

find . -type f -exec sed -i 's/abc/abcd/g' {} \;

Or if you are fan of xargs:

find . -type f | xargs -I {} sed -i 's/abc/abcd/g' {}

<https://yoyoclouds.wordpress.com/2012/05/19/best-practices-installing-centos-vm-on-vmware-workstation-part-1/>

<https://yoyoclouds.wordpress.com/2012/05/19/best-practices-installing-centos-vm-on-vmware-workstation-part-2/>

## Installing VMware Tools from the Command Line with the Tar Installer

# Linux centos 下 eclipse 打开文件时关闭

http://processors.wiki.ti.com/index.php/Linux\_Host\_Support#cairo-misc.c:380:\_cairo\_operator\_bounded\_by\_source:\_Assertion\_.60NOT\_REACHED.27\_failed

[http](http://cpro.baidu.com/cpro/ui/uijs.php?rs=1&u=http%3A%2F%2Fwww%2Eth7%2Ecn%2Fsystem%2Flin%2F201412%2F78392%2Eshtml&p=baidu&c=news&n=10&t=tpclicked3_hc&q=smileking_cpr&k=http&k0=java&kdi0=8&k1=eclipse&kdi1=8&k2=%BD%E2%BE%F6%B7%BD%B0%B8&kdi2=8&k3=http&kdi3=8&k4=%CF%B5%CD%B3&kdi4=8&sid=19903660dc994e19&ch=0&tu=u1682280&jk=cb127cde8a87b8dd&cf=29&fv=16&stid=9&urlid=0&luki=4&seller_id=1&di=128)://processors.wiki.ti.com/index.php/Linux\_Host\_Support

现象描述：在 centos [系统](http://cpro.baidu.com/cpro/ui/uijs.php?rs=1&u=http%3A%2F%2Fwww%2Eth7%2Ecn%2Fsystem%2Flin%2F201412%2F78392%2Eshtml&p=baidu&c=news&n=10&t=tpclicked3_hc&q=smileking_cpr&k=%CF%B5%CD%B3&k0=java&kdi0=8&k1=eclipse&kdi1=8&k2=%BD%E2%BE%F6%B7%BD%B0%B8&kdi2=8&k3=http&kdi3=8&k4=%CF%B5%CD%B3&kdi4=8&sid=19903660dc994e19&ch=0&tu=u1682280&jk=cb127cde8a87b8dd&cf=29&fv=16&stid=9&urlid=0&luki=5&seller_id=1&di=128)下，打开eclipse后，打开文件时，eclipse退出。

                    在命令行下启动时，会打印异常信息：

[java](http://cpro.baidu.com/cpro/ui/uijs.php?rs=1&u=http%3A%2F%2Fwww%2Eth7%2Ecn%2Fsystem%2Flin%2F201412%2F78392%2Eshtml&p=baidu&c=news&n=10&t=tpclicked3_hc&q=smileking_cpr&k=java&k0=java&kdi0=8&k1=eclipse&kdi1=8&k2=%BD%E2%BE%F6%B7%BD%B0%B8&kdi2=8&k3=http&kdi3=8&k4=%CF%B5%CD%B3&kdi4=8&sid=19903660dc994e19&ch=0&tu=u1682280&jk=cb127cde8a87b8dd&cf=29&fv=16&stid=9&urlid=0&luki=1&seller_id=1&di=128): cairo-misc.c:380: \_cairo\_operator\_bounded\_by\_source: Assertion `NOT\_REACHED' failed.

问题分析：该问题会在Centos6.6及更高版本出现。也会在其他版本中出现。

[解决方案](http://cpro.baidu.com/cpro/ui/uijs.php?rs=1&u=http%3A%2F%2Fwww%2Eth7%2Ecn%2Fsystem%2Flin%2F201412%2F78392%2Eshtml&p=baidu&c=news&n=10&t=tpclicked3_hc&q=smileking_cpr&k=%BD%E2%BE%F6%B7%BD%B0%B8&k0=java&kdi0=8&k1=eclipse&kdi1=8&k2=%BD%E2%BE%F6%B7%BD%B0%B8&kdi2=8&k3=http&kdi3=8&k4=%CF%B5%CD%B3&kdi4=8&sid=19903660dc994e19&ch=0&tu=u1682280&jk=cb127cde8a87b8dd&cf=29&fv=16&stid=9&urlid=0&luki=3&seller_id=1&di=128)：在[eclipse](http://cpro.baidu.com/cpro/ui/uijs.php?rs=1&u=http%3A%2F%2Fwww%2Eth7%2Ecn%2Fsystem%2Flin%2F201412%2F78392%2Eshtml&p=baidu&c=news&n=10&t=tpclicked3_hc&q=smileking_cpr&k=eclipse&k0=java&kdi0=8&k1=eclipse&kdi1=8&k2=%BD%E2%BE%F6%B7%BD%B0%B8&kdi2=8&k3=http&kdi3=8&k4=%CF%B5%CD%B3&kdi4=8&sid=19903660dc994e19&ch=0&tu=u1682280&jk=cb127cde8a87b8dd&cf=29&fv=16&stid=9&urlid=0&luki=2&seller_id=1&di=128)安装目录下的eclipse.ini配置文件中添加：-Dorg.eclipse.swt.internal.gtk.cairoGraphics=false

## FAQ

### cairo-misc.c:380: \_cairo\_operator\_bounded\_by\_source: Assertion `NOT\_REACHED' failed

A: This is an issue on CentOS 6.6 and later. it could also occur on other distributions. To work around this problem add the following line to your ccstudio.ini file (/ccsv6/eclipse)

-Dorg.eclipse.swt.internal.gtk.cairoGraphics=false

<http://www.cnblogs.com/candle806/p/3439469.html>

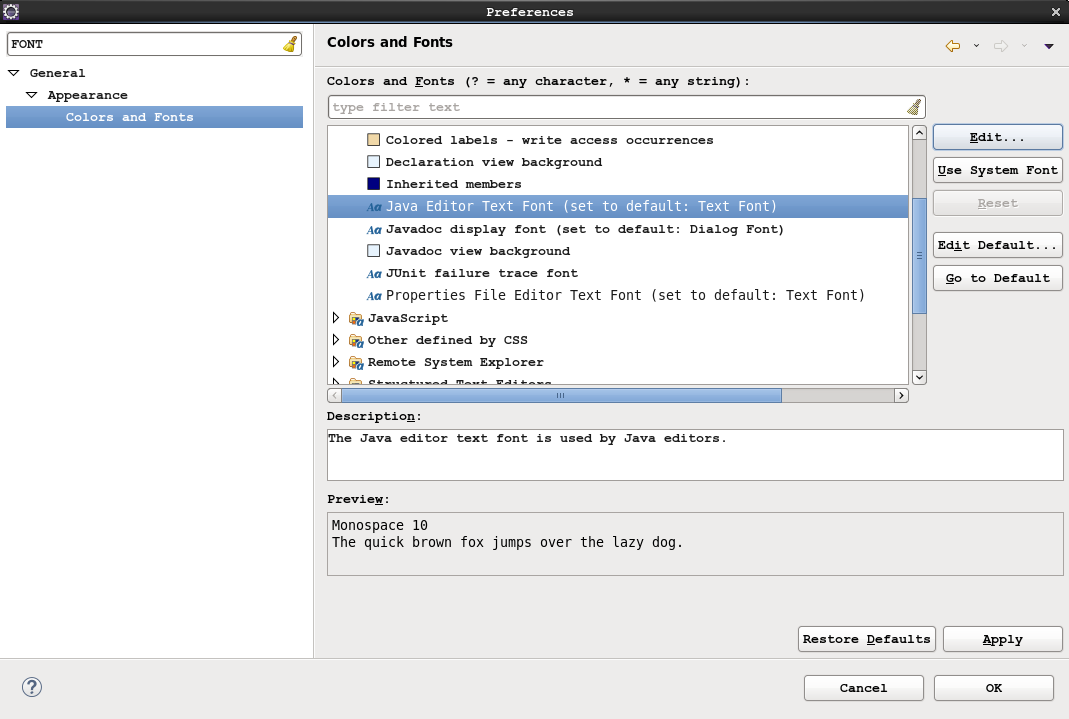
[**Eclipse+Maven创建webapp项目<一>**](http://www.cnblogs.com/candle806/p/3439469.html)

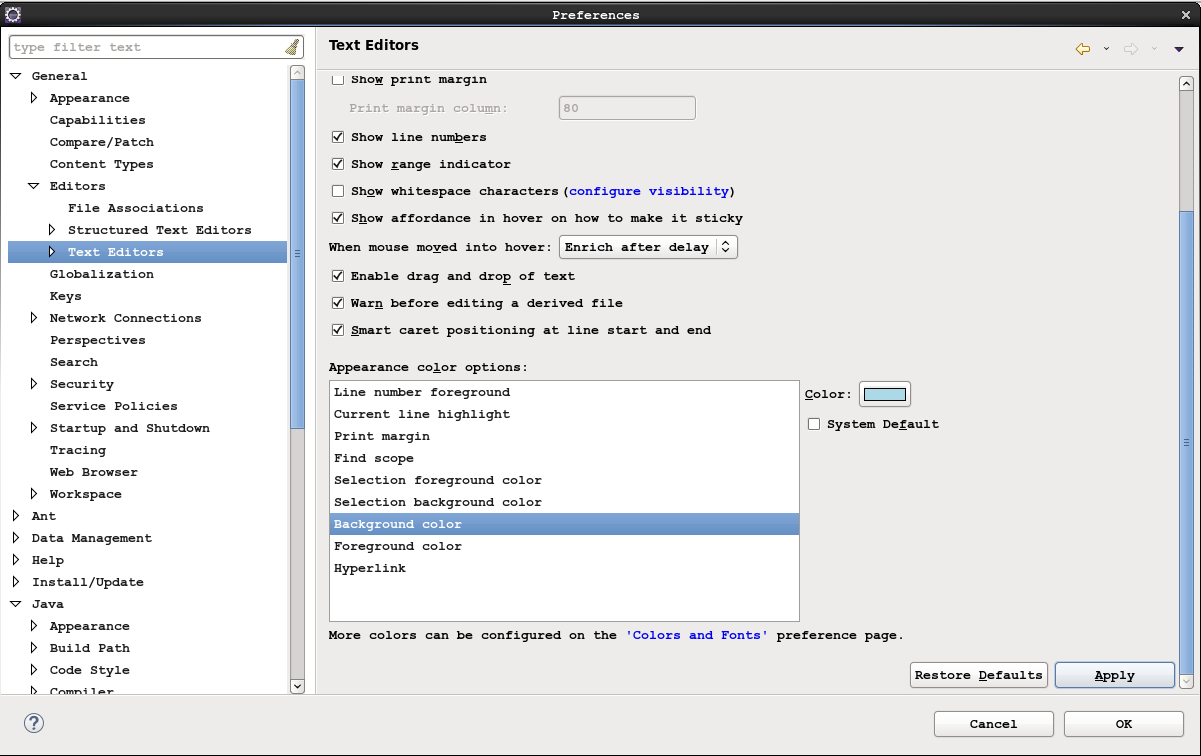
<http://blog.csdn.net/smilevt/article/details/8215558/>

[eclipse构建maven的web项目](http://blog.csdn.net/smilevt/article/details/8215558)

<http://jingyan.baidu.com/article/9f63fb91a7d2a5c8400f0e20.html>

# 使用Eclipse创建Maven Web工程





<http://www.cyberciti.biz/faq/linux-delete-user-password/>

## Command to delete user password under Linux

Type the following command to delete a user password:  
# passwd --delete username  
OR  
# passwd -d username

[**nixCraft**](http://www.cyberciti.biz/faq)

<http://www.cyberciti.biz/faq/unix-create-user-account/>

# UNIX Create User Account

http://nixcraft.cn/

<http://geeksterminal.com/how-to-install-glib-glibc/1392/>

### To install glib on CentOS/RedHat/Fedora: (tagbar依赖)

**yum install** glib**\***

<http://ask.xmodulo.com/check-glibc-version-linux.html>

# [How to check glibc version on Linux](http://ask.xmodulo.com/check-glibc-version-linux.html)

## Method One

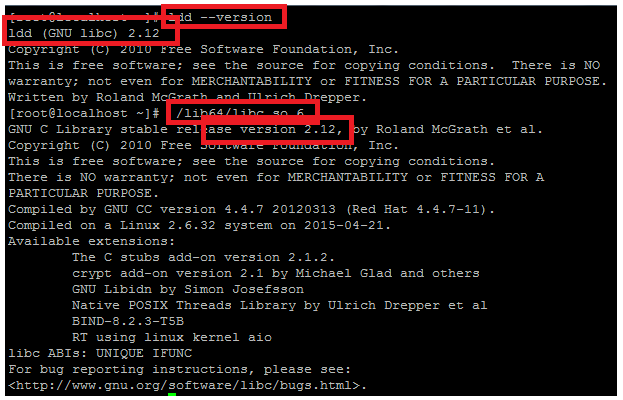
A simple command-line to check the version of the GNU C library is as follows.

$ ldd --version

## Method Two

On 64-bit Red Hat based system:

$ /lib64/libc.so.6



<http://www.liquidweb.com/kb/how-to-check-the-glibc-gnu-libc-version-on-centos-6-and-centos-7/>

[How to Check the glibc (GNU libc) Version on CentOS 6 and CentOS 7](http://www.liquidweb.com/kb/how-to-check-the-glibc-gnu-libc-version-on-centos-6-and-centos-7/)

<http://stackoverflow.com/questions/16678487/wget-command-to-download-a-file-and-save-as-a-different-filename>

Use the -O option.

E.g.

wget google.com

...

16:07:52 (538.47 MB/s) - `index.html' saved [10728]

vs.

wget google.com -O foo.html

...

16:08:00 (1.57 MB/s) - `foo.html' saved [10728]

<http://stackoverflow.com/questions/5732696/centos-directory-structure-as-tree>

# [CentOS directory structure as tree?](http://stackoverflow.com/questions/5732696/centos-directory-structure-as-tree)

**# yum install tree -y**

<http://www.cyberciti.biz/faq/unix-linux-appleosx-bsd-bash-script-find-what-directory-itsstoredin/>

# Bash Script: Find Out In What Directory Script Is Stored Under Unix Or Linux

*#!/bin/bash*

*# Name: /tmp/demo.bash :*

*# Purpose: Tell* ***in*** *what directory $0 is stored in*

*# Warning: Not tested* ***for*** *portability*

*# ------------------------------------------------*

*##* ***who*** *am i? ##*

\_script="$(readlink -f ${BASH\_SOURCE[0]})"

*## Delete* ***last*** *component from $\_script ##*

\_base="$(dirname $\_script)"

*## Okay, print it ##*

**echo** "Script name : $\_script"

**echo** "Current working dir : $PWD"

**echo** "Script location path (dir) : $\_base"

<http://www.unixmen.com/install-apache-ant-maven-tomcat-centos-76-5/>

# How To Install Apache Ant, Maven And Tomcat On CentOS 7/6.5

Make sure you’ve allowed the Tomcat default port **8080** through iptables.

##### On CentOS 6.x systems:

Edit file **/etc/sysconfig/iptables**,

vi /etc/sysconfig/iptables

Add the following line:

[...]

-A INPUT -m state --state NEW -m tcp -p tcp --dport 8080 -j ACCEPT

[...]

Save and close the file and restart the iptables service.

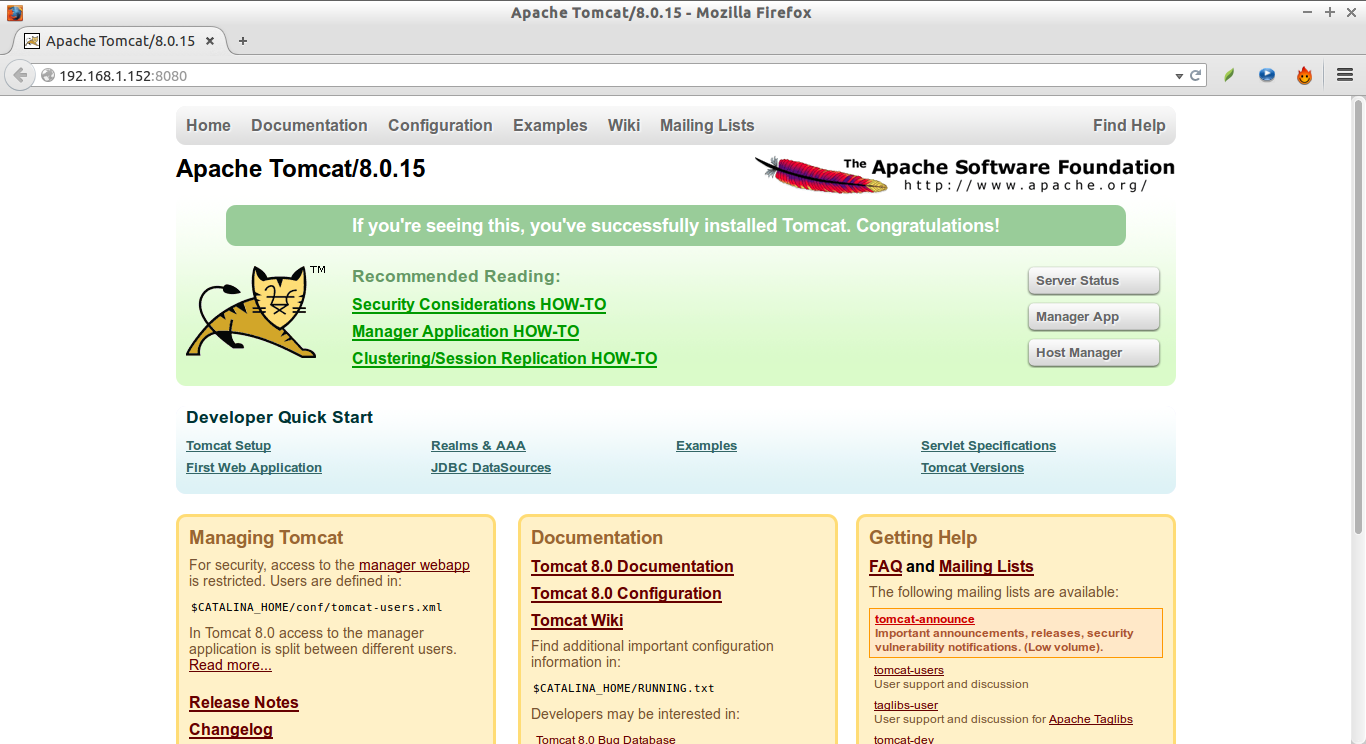
service iptables restart

##### On CentOS 7 systems:

firewall-cmd --permanent --add-port=8080/tcp

firewall-cmd --reload

Now, navigate to **http://ip-address:8080** from your web browser. You should see the following screen.



<http://www.idevelopment.info/data/Unix/General_UNIX/GENERAL_RedirectingMANPagestoaFile.shtml>

Redirecting MAN Pages to a File

*by Jeff Hunter, Sr. Database Administrator*

Trying to redirect a man page to a file using only the redirection command will produce many unreadable characters. Before redirecting the file, pipe the man page through the "col" command as show below.

# **man rcs | col -bx > rcs.txt**

<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)

The tmux FAQ ( <https://raw.githubusercontent.com/tmux/tmux/master/FAQ> ) explicitly advises against setting TERM to anything other than screen or screen-256color in your shell init file, so don't do it!

Here's what I use:

~$ which tmux

tmux: aliased to TERM=xterm-256color tmux

and in in my .tmux.conf:

set -g default-terminal "screen-256color"

Aliasing tmux to "tmux -2" should also do the trick

<http://www.cnblogs.com/itech/archive/2012/12/17/2822170.html>

# [为什么使用tmux](http://www.cnblogs.com/itech/archive/2012/12/17/2822170.html)

为什么使用tmux：

tmux比screen有更多的功能，能够保持你的工作环境连续性。例如tmux解决如下的问题：

1）下班后，你需要断开ssh或关闭电脑，你的ssh连接将丢失；

2）在公司打开的ssh，在家里也需要访问；

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

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

The tmux FAQ ( <https://raw.githubusercontent.com/tmux/tmux/master/FAQ> ) explicitly advises against setting TERM to anything other than screen or screen-256color in your shell init file, so don't do it!

Here's what I use:

~$ which tmux

tmux: aliased to TERM=xterm-256color tmux

and in in my .tmux.conf:

set -g default-terminal "screen-256color"

Aliasing tmux to "tmux -2" should also do the trick.

可以参考一些参见：

<http://chibicode.com/vimrc/>

# Goodbye Vim, Hello Sublime Text, and My Final .vimrc

<https://www.douban.com/note/486635690/>

# 逼格提高指南

✓ sheerun/vim-polyglot：几乎所有程序语言的源文件syntax/tab等的支持。有此一个插件，就不再需要 vim-ruby，vim-go等一票单独的语言插件了。

<https://davidosomething.com/blog/vim-for-javascript/>

# [Setting up Vim for JavaScript development](https://davidosomething.com/blog/vim-for-javascript/)

# [How to show only hidden files in Terminal?](http://askubuntu.com/questions/468901/how-to-show-only-hidden-files-in-terminal)

<http://askubuntu.com/questions/468901/how-to-show-only-hidden-files-in-terminal>

The command :

ls -ld .?\*

Will only list hidden files .

Explain :

-l use a long listing format

-d, --directory

list directory entries instead of contents, and do not derefer‐

ence symbolic links

.?\* will only state hidden files

<http://help.accusoft.com/PCC/v8.3/HTML/How%20to%20Install%20Microsoft%20Fonts%20on%20Linux.html>

**安装微软的Courier New字体：**

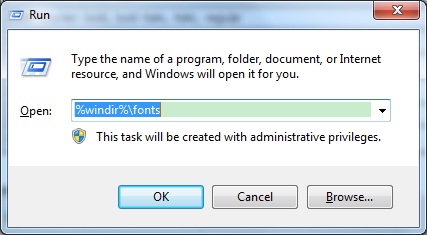
在win7系统下执行：

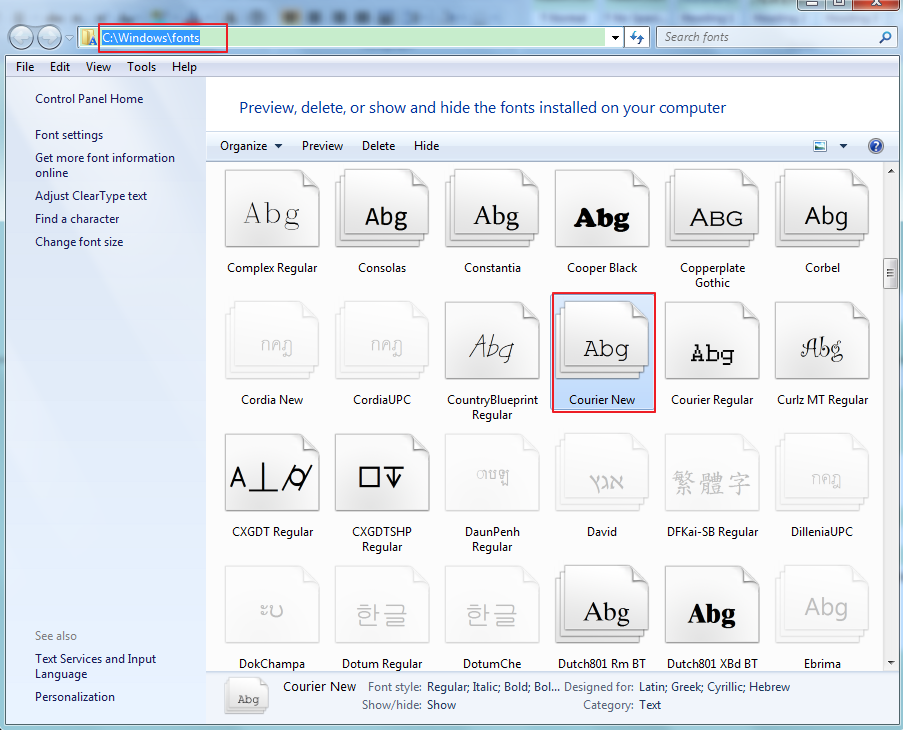
<https://support.microsoft.com/en-us/kb/314960>

首先在windows 7系统下找到Courier New字体文件，

1. Click **Start**, and then click **Run**.
2. Type the following command, and then click **OK**:

%windir%\fonts





# How to Manually Install Fonts in Ubuntu 16.04

<http://ubuntuhandbook.org/index.php/2016/05/manually-install-fonts-ubuntu-16-04/>

mkdir /usr/share/fonts/ms\_fonts #然后将Courier New字体文件文件copy到该目录下

[root@localhost ms\_fonts]# fc-cache –fv

<https://der-linux-admin.de/2014/12/centos-installation-der-microsoft-truetype-fonts-msttcorefonts/>

# CentOS – Installation der Microsoft TrueType Fonts (msttcorefonts)

wget ftp://rpmfind.net/linux/sourceforge/r/re/repo-1c/RPMS/msttcorefonts-2.5-1.noarch.rpm

yum install msttcorefonts-2.5-1.noarch.rpm

/usr/share/fonts/msttcorefonts/

<http://www.itzgeek.com/how-tos/linux/centos-how-tos/how-to-install-microsoft-truetype-fonts-in-centos-6-rhel-6.html>

# How to install Microsoft TrueType Fonts in CentOS 6 / RHEL 6

[root@geeksite~/]# wget http://www.itzgeek.com/msttcore-fonts-2.0-3.noarch.rpm

[root@geeksite~/[# rpm -Uvh msttcore-fonts-2.0-3.noarch.rpm

<https://ask.fedoraproject.org/en/question/8331/the-printscreen-key-on-my-keyboard-does-not-take-a-screenshot/>

# [The printscreen key on my keyboard does not take a screenshot.](https://ask.fedoraproject.org/en/question/8331/the-printscreen-key-on-my-keyboard-does-not-take-a-screenshot/)

The default behaviour when pressing the PrintScreen key is to automatically place your screenshot in the **Pictures** folder in your home directory (i.e. "~/Pictures"). The click and the flash mean that the screenshot has been taken, so just check the Pictures folder for your screenshot.

Other than just the "Print Screen' key, which saves your whole Desktop to the Pictures folder, GNOME3 also has the following shortcuts enabled by default for screenshot actions:

* **PrintScreen** -- Takes a screenshot of your entire desktop and saves it to the Pictures folder.
* **Alt + PrintScreen** -- Saves a screenshot of the focused window to the Pictures Folder
* **Shift + PrintScreen** -- Lets you select an area of the screen, and saves to the Pictures Folder
* **Ctrl + PrintScreen** -- Takes a screenshot of your entire desktop and copies it to the clipboard.
* **Ctrl + Alt + PrintScreen** -- copies a screenshot of the focused window to the clipboard.
* **Ctrl + Shift + PrintScreen** -- Lets you select an area of the screen, and copies it to the clipboard.
* **Ctrl + Shift + Alt + R** -- Records a Screencast) of your entire desktop and saves it to your Videos folder.

These shortcuts also can be changed in the GNOME keyboard settings if your keyboard does not have a PrintScreen key.

**Is there a GUI for taking screenshots, rather than keyboard shortcuts?**

If you want a little bit more control over your screenshots, you can also use the GNOME screenshots tool that is included in Fedora by default (search for "Screenshot" in the overview). It provides you a simple GUI for changing settings such as showing the cursor, and screenshot delays:

# How To Use SSH to Connect to a Remote Server in Ubuntu

<https://www.digitalocean.com/community/tutorials/how-to-use-ssh-to-connect-to-a-remote-server-in-ubuntu>

ssh remote\_username@remote\_host

ssh –p 22 remote\_username@remote\_host #同时指定remote server端口号

# SSH Passwordless Login Using SSH Keygen in 5 Easy Steps

<http://www.tecmint.com/ssh-passwordless-login-using-ssh-keygen-in-5-easy-steps/>

# [how to install filezilla on CentOS?](http://stackoverflow.com/questions/19586904/how-to-install-filezilla-on-centos)

<http://stackoverflow.com/questions/19586904/how-to-install-filezilla-on-centos>

For Cent OS 7

sudo yum -y install epel-release

sudo yum -y install filezilla

Go to **Application > Internet > Filezilla**

# Google Chrome 52 Released – Install on RHEL/CentOS 7/6 and Fedora 23-15

<http://www.tecmint.com/install-google-chrome-on-redhat-centos-fedora-linux/>

**Update** : Sadly, the **Google Chrome** browser no longer supports the most famous commercial distribution **RHEL 6.x** and its free clones such as **CentOS** and **Scientific Linux**.

Luckily, there is a script developed by [Richard Lloyd](http://chrome.richardlloyd.org.uk/), that automatically download and install latest **Google Chrome** browser by picking libraries from a more recent released distro and put those libraries in (**/opt/google/chrome/lib**) directory and then you can able to run **Google Chrome** on **RHEL/CentOS 6.x** versions.

# wget http://chrome.richardlloyd.org.uk/install\_chrome.sh

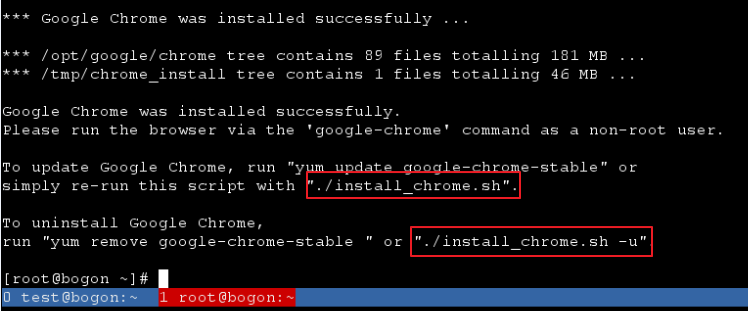
# chmod u+x install\_chrome.sh

# ./install\_chrome.sh

#### Step 3: Starting Chrome Web Browser

Start browser with non-root user.

# google-chrome &



[http://tecadmin.net/install-google-chrome-in-centos-rhel-and-fedora/#](http://tecadmin.net/install-google-chrome-in-centos-rhel-and-fedora/)

# How to Install Google Chrome 52 in CentOS/RHEL 7/6 & Fedora 23/22

# Google Chrome Cheat Sheet

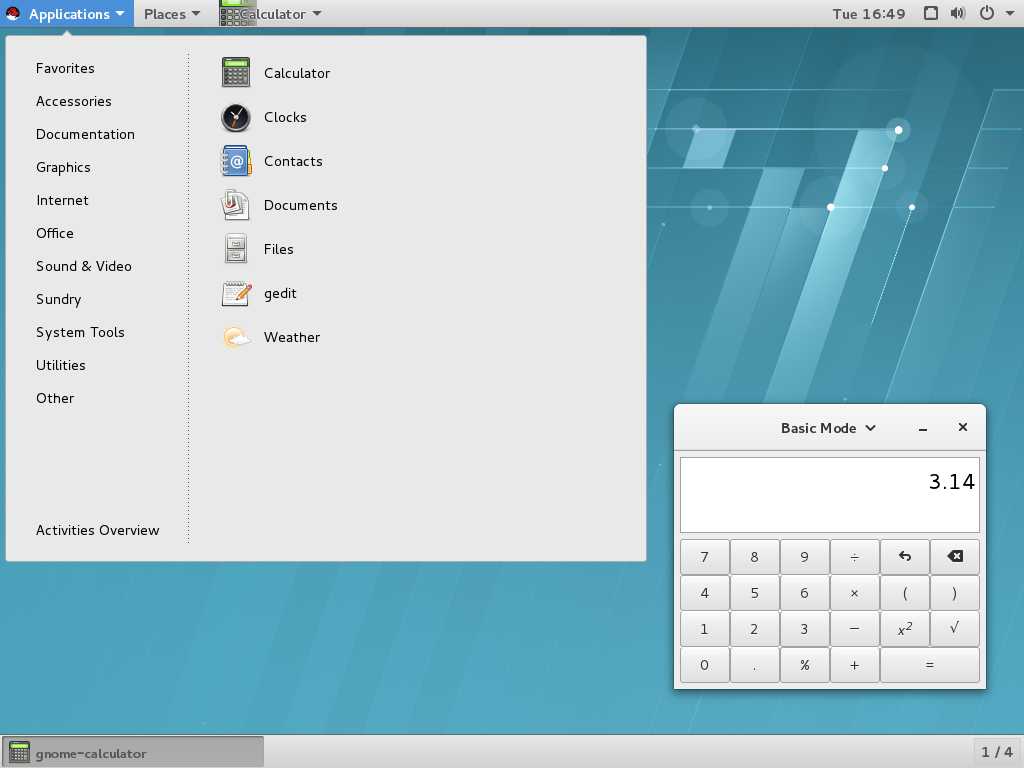
<http://www.shakeuplearning.com/blog/google-chrome-cheat-sheet/>

<http://www.shakeuplearning.com/wp-content/uploads/2014/03/ChromeCheatSheet.pdf>

<https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/Desktop_Migration_and_Administration_Guide/what-is-gnome-classic.html>

## 1.3. WHAT IS GNOME CLASSIC?

GNOME Classic is a GNOME Shell feature and mode for users who prefer a more traditional desktop experience. While GNOME Classic is based on GNOME 3 technologies, it provides a number of changes to the user interface:



### 1.3.1. The GNOME Classic Extensions

GNOME Classic is distributed as a set of [GNOME Shell extensions](https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/Desktop_Migration_and_Administration_Guide/GNOME-shell-extensions.html). The GNOME Classic extensions are installed as dependencies of the *gnome-classic-session* package, which provides components required to run a GNOME Classic session. Because the GNOME Classic extensions are enabled by default on Red Hat Enterprise Linux 7, GNOME Classic is the default Red Hat Enterprise Linux 7 desktop user interface.

[root@localhost ~]# **yum install clang**

