

Install Basic System

download CentOS

```
http://isoredirect.centos.org/centos/6/isos/x86_64/  
http://isoredirect.centos.org/centos/6/isos/i386/
```

set network

add ssh server

```
# yum -y install openssh-server  
# chkconfig sshd on  
# /etc/init.d/sshd start
```

disable SELinux changing to "SELINUX=disabled" in

```
# vim /etc/selinux/config
```

Enable other Repositories

import the GPG keys for software packages:

```
# rpm --import /etc/pki/rpm-gpg/RPM-GPG-KEY*
```

RPMForge and EPEL (please verify versions before install):

```
# rpm --import http://dag.wieers.com/rpm/packages/RPM-GPG-KEY.dag.txt  
# wget  
http://pkgs.repoforge.org/rpmforge-release/rpmforge-release-0.5.2-2.el6.rf.x  
86_64.rpm  
# rpm -ivh rpmforge-release-0.5.2-2.el6.rf.x86_64.rpm
```

```
# rpm --import https://fedoraproject.org/static/0608B895.txt  
# wget  
http://dl.fedoraproject.org/pub/epel/6/x86_64/epel-release-6-5.noarch.rpm  
# rpm -ivh epel-release-6-5.noarch.rpm
```

```
# yum install yum-priorities  
# vim /etc/yum.repos.d/epel.repo
```

add "priority=10" to the [epel] section.

```
# yum update
```

note: this will upgrade your system to CentOS 6.2 !!

Remote Desktop / VNC server

CentOS 6 comes with vncserver.

Set your user password

```
$ vncpassword
```

Set your display and geometry:

```
# vim /etc/sysconfig/vncservers
```

add vnc user and options:

```
VNCSERVERS="1:suporte"  
VNCSERVERARGS[1]="-geometry 1024x768"
```

enable vncserver service

```
# chkconfig vncserver on
```

start service

```
# service vncserver start
```

if your vncserver not start, login as vncuser (in my case suporte), execute the command:

```
# $ vncserver
```

And open firewall ports (5901 tcp/udp) with

```
# system-config-firewall
```

KVM Qemu Virt-Manager

install packages

```
# yum groupinstall "Virtualization" "Virtualization Tools" "Virtualization  
Client" "Virtualization Platform"
```

Bridge your network ⇒⇒ you'll loose connection !!!

disable NetworkManager and enable simple network

```
# chkconfig NetworkManager off  
# chkconfig network on  
# service NetworkManager stop  
# service network start
```

go to network config dir

```
# cd /etc/sysconfig/network-scripts
```

modify interface to make bridge **ifcfg-eth0**

```
DEVICE=eth0
# change the hardware address to match the hardware address your NIC uses
HWADDR=00:16:76:D6:C9:45
ONBOOT=yes
BRIDGE=br0
```

create bridge **ifcfg-br0**

```
DEVICE=br0
TYPE=Bridge      # case sensitive
BOOTPROTO=dhcp # or static
ONBOOT=yes
DELAY=0
```

restart network service

```
# service network restart
```

reference: [RHEL6](#)

ObsPy

```
yum install ipython scipy numpy python-matplotlib python-matplotlib-tk
python-lxml python-setuptools
```

```
easy_install -U distribute
```

```
easy_install -N -U obspy.core
easy_install -N -U obspy.mseed
easy_install -N -U obspy.sac
easy_install -N -U obspy.gse2
easy_install -N -U obspy.imaging
```

```
easy_install -N -U obspy.signal
easy_install -N -U obspy.arclink
easy_install -N -U obspy.xseed
easy_install -N -U obspy.seishub
easy_install -N -U obspy.iris
easy_install -N -U obspy.seisan
easy_install -N -U obspy.wav
easy_install -N -U obspy.fissures
easy_install -N -U obspy.sh
```

```
easy_install -N -U obspy.taup
```

```
easy_install -N -U suds
```

SAC no CentOS

Igual pra todos os linux (=

Baixar o arquivo de acordo com a arquitetura do sistema:

```
wget ftp://trovador.iag.usp.br/tecnico/sac-101.4-linux_x86_64.tar.gz
```

Ir ao diretório de instalação

```
cd /path/to/install (ex /usr/local)
```

Descompactar o arquivo

```
tar -zxvf /path/to/file/sac*.tar.gz
```

Adicionar as seguintes linhas no ~/.bashrc ou no /etc/profile

```
#SAC
export SACHOME=/path/where/you/install/sac # (ex /usr/local/sac)
export PATH=${PATH}:${SACHOME}/bin
export SACAUX=${SACHOME}/aux
```

GMT

Respeitando a arquitetura, faça:

```
sudo yum install GMT.x86_64
```

Não é necessário adicionar paths no bashrc. Contudo, o GMT pede um arquivo de configuração que deve ficar na pasta home do usuário e deve ser salvo como .gmtdefaults. Sim “.gmt”, oculto mesmo.

Uma cópia deste arquivo pode ser adquirida em:

```
ftp://trovador.iag.usp.br/pub/gmtdefaults.txt
```

Para testar se está bem, faça:

```
psxy
```

E veja se recebe uma mensagem de ajuda. (=

PGPLOT

Always download the original source code of PGPLOT from its [homepage](#), and install the package manually (at the time of writing these notes, the latest version is `pgplot5.2.tar.gz`). For this to work properly, the appropriate X11 files must be available in your machine. I also install `gfortran` as the Fortran compiler. Both tasks can be performed executing:

```
$ sudo yum install libX11-devel
$ sudo yum install gcc-gfortran
```

1 - Download the distribution file `pgplot5.2.tar.gz` 2 - Decompress and extract the contents of the distribution file in a source directory:

```
# cd /opt/src
# mv ~/Downloads/pgplot5.2.tar.gz .
# tar zxvf pgplot5.2.tar.gz
```

The last command will create `/usr/local/src/pgplot` and subdirectories. 3. Create the directory where PGPLOT will be actually installed:

```
# mkdir /opt/pgplot
# cd /opt/pgplot
```

4-Copy the file `drivers.list` from the source directory to the installation directory:

```
# cp /opt/src/pgplot/drivers.list .
```

5-Edit that file and remove the exclamation mark (first column of each row) in front of the following graphic devices: `/PS`, `/VPS`, `/CPS`, `/VCPS` and `/XServe`. 6-Create the makefile. From the installation directory `/usr/local/pgplot` execute:

```
# /opt/src/pgplot/makemake /opt/src/pgplot linux g77_gcc_aout
```

7-Edit the file `makefile` and change the line:

```
FCOMPL=g77
by
FCOMPL=gfortran
```

8-Compile the source files:

```
# make
# make cpg
# make clean
```

After the installation, make sure that the expected environment variables are properly set (at least `PGPLOT_DIR` must exist; `PGPLOT_DEV` is also quite useful):

```
$ export PGPLOT_DIR=/opt/pgplot
```

```
$ export PGPLOT_DEV=/Xserve
```

It is useful to introduce those definitions in the `.bashrc` file at the home directory of the user.

Finally, execute one of the demos in order to check that everything is fine:

```
$ /opt/pgplot/pgdemo1
```

FFTW

Instalar as bibliotecas FFT: Fast Fourier Transform library

```
sudo yum install fftw-devel.x86_64
```

Easy like that.

NTFS Filesystem

Install the following packages:

```
$ yum install fuse fuse-ntfs-3g
```

If the rpmforge repo is disabled by default,

```
$ yum --enablerepo=rpmforge install fuse fuse-ntfs-3g
```

NAWK

Instalando o nawk : “The one true awk” descended from UNIX V7

```
$ sudo yum install nawk.x86_64
```

Q: Why not Gawk !?!

```
# yum install gawk
```

Q2: What's difference !?!

```
[[http://www.thegeekstuff.com/2011/06/awk-nawk-gawk/]]  
[[http://www.well.ox.ac.uk/~johnb/comp/awk/awk.html]]
```



TestDisk

Tool to check and undelete partition, PhotoRec recovers lost files

```
$ sudo yum install testdisk
```

f77

```
$ setenv INSTALL /usr/local
$ curl
"http://netlib.sandia.gov/cgi-bin/netlib/netlibfiles.tar?filename=netlib/f2c"
" -o "f2c.tar"
$ tar -xvf f2c.tar
$ gunzip -rf f2c/*
$ cd f2c
$ mkdir libf2c
$ mv libf2c.zip libf2c
$ cd libf2c
$ unzip libf2c.zip
$ cp makefile.u Makefile
$ make
$ cp f2c.h $INSTALL/include
$ cp libf2c.a $INSTALL/lib
$ cd ../src
$ cp makefile.u Makefile
$ chmod +x sum
$ make
$ cp f2c $INSTALL/bin
$ cd ..
$ mkdir -p $INSTALL/man/man1
$ cp f2c.1t $INSTALL/man/man1
$ cp fc $INSTALL/bin/f77
$ chmod +x $INSTALL/bin/f77
$ cd ..
$ rm -rf f2c
$ echo "=====SUMMARY======"
$ echo $0 " has built and installed:"
$ find $INSTALL -name '*f2c*' -mmin -5
$ find $INSTALL -name '*f77*' -mmin -5
```

Translate, compile, link, and run a program:

```
$ f2c hello.f
$ gcc -c hello.c
$ gcc -o hello hello.o -lf2c -lm
$ ./hello
```

Or combine these into a single command:

```
$ f77 -o hello hello.f
```

From:

<https://www.moho.iag.usp.br/wiki/> - Wiki@CSUSP

Permanent link:

https://www.moho.iag.usp.br/wiki/doku.php?id=soft:howto_centos_server

Last update: **2017/06/05 12:06**

