Linux Software Installation – Exercises 2

Part 1. Install PYTHON software with PIP

1.1 Check the version of python and pip

On BioHPC computers, the "python" command points to "python2.7.5", with "pip" command linked to this default python. The "python3" command points to "python3.6.7", with "pip3" command linked to this default python3.

Before you install any python modules, it is always a good idea to ask these three questions.

1. Which copy of Python installation you will be running?
2. What is the version of the Python?
3. Whether the pip command you are running is associated with the Python you will be running?

Use the following commands to address these questions:

```
which python3
ls -l /usr/local/bin/python3
python3 -V
which pip3
head -n1 /usr/local/bin/pip3
```

1.2 Install and use python module deepTools
Install deepTools in the directory ~/.local/

```
pip3 install deepTools --user
```

Run deepTools installed in the directory ~/.local/

```
export PATH=$HOME/.local/bin:$PATH
which deeptools
export LC_ALL=en_US.utf-8
export LANG=en_US.utf-8
deeptools
```

- The new deeptools requires Locale to be set as en_US.utf-8.

1.3 Check the version for deeptool and numpy python modules

```
python3
import deeptools
deeptools.__file__

import numpy
numpy.__file__
numpy.__version__
```

After you are done, press “Ctrl-D” to exit python prompt.

- Make sure that you type **double-underline** for the command “__version__” .
- The "__file__" command checks the files from which the modules are loaded;
- The deeptools module does not have the “__version__” property;
- An old version of pysam could cause deeptools to fail, in that case you will need to update the pysam by “--upgrade” option;

1.4 (optional) Install a different version of deeptools in an alternative location

Install deepTools v3.3.2 in /workdir/$USER

```
pip3 install deepTools==3.3.2 --prefix /workdir/$USER --ignore-installed
```

Run deepTools in /workdir/$USER

```
export PATH=/workdir/$USER/bin:$PATH
export PYTHONPATH=/workdir/$USER/lib/python3.6/site-packages
which deeptools
export LC_ALL=en_US.utf-8
export LANG=en_US.utf-8
deeptools
```

Part 2. Install PERL software with CPAN

2.1 Configure CPAN

```
mkdir ~/perl
export PERL5LIB=~/perl/lib/perl5
cpan
```

At cpan pompt “cpan[1]>”, type the following commands:

```
o conf makepl_arg INSTALL_BASE=~/perl
o conf mbuild_arg INSTALL_BASE=~/perl
o conf prefs_dir ~/perl /prefs
o conf commit
```

2.2 Install PERL modules with CPAN

```
install XML::Simple
```

This module is only accessible if you “export PERL5LIB=~/perl/lib/perl5”. Delete the whole directory /perl if installation goes wrong.

Part 3. Install C software

Normally, the software web site or the README (sometimes INSTALL) file in the source code directory provides step-by-step instructions. As a non-root user, quite often you need to modify the instructions, e.g. adding “--prefix=~/mydirectory” at the configuration step.

3.1 Download the source code

```
cd /workdir/$USER
wget http://catchenlab.life.illinois.edu/stacks/source/stacks-2.53.tar.gz
tar xvfz stacks-2.53.tar.gz
```

- If the source code is available from github, use “git clone https://github.com/lh3/bwa.git” to get the latest source code.
- If the source code you downloaded is a *tar.bz2 file, use “tar xvfj myfile.tar.bz2” command to decompress.

3.2 Configure

```
mkdir $HOME/stacks
cd /workdir/$USER/stacks-2.53
export LD_LIBRARY_PATH=/usr/local/gcc-7.3.0/lib64:/usr/local/gcc-7.3.0/lib
export PATH=/usr/local/gcc-7.3.0/bin:$PATH
./configure --prefix=$HOME/stacks
```

- The default gcc compiler would not work with stacks v2. If you use the default gcc, you will get an error message complaining your gcc is too old. The two commands here “export LD_LIBRARY_PATH..” “export PATH...” switch the default gcc to v7.3.0.
- The “--prefix=“ parameter instruct the software to be installed in "$HOME/stacks".
- After this step, a new Makerfile is created with instructions how to compile and install the software. The Makerfile is a text file. If you want, you can use any text editor to examine the file, you will find lines like “prefix = /home/xxxxx/stacks”.

3.3 Compile

```
make
make install
```

·After this step, you will find binary executables and libraries in your installation directory: $HOME/stacks.

3.4 Run software

```
export LD_LIBRARY_PATH=/usr/local/gcc-7.3.0/lib64:/usr/local/gcc-7.3.0/lib
export PATH=$HOME/stacks/bin:$PATH
sstacks
```

Part 4. Install R package

First, start R by command “R”. If you want to use a different version of R, you need to modify PATH, e.g. “export PATH=/programs/R-3.4.2s/bin:$PATH”.

At R prompt “>”, type the following command to install R package “qtl”

```
install.packages("qtl")
```

You will be prompted for two questions. Answer "yes" and press "Enter" to both questions.

Would you like to use a personal library instead? (yes/No/cancel) yes
Would you like to create a personal library? yes

Still at R prompt, load qtl package, get the version and physical path of the package

```
library(qtl)
packageVersion("qtl")
find.package("qtl")
```

The package is installed in $HOME/R. If something goes wrong, you can delete the whole directory and start from scratch.