PASSWORDLESS LOGIN

Here is a set of commands to configure dc machines so that you don't have to type your password every time you log in.

(1) login to a dc machine and go to your home directory

{dc02:~} cd ~

(2) generate the ssh-key pairs - private and public keys in your $HOME/.ssh directory

{dc02:~} ssh-keygen -t rsa

//Just press enter for passphrase

…

…

The key fingerprint is:

… something …

The key's randomart image is:

…

… (special image) …

…

(3) go to your .ssh directory to see those pairs

{dc02:~} cd .ssh

// first is the private key, second is the public key

{dc02:~/.ssh} ls

id_rsa id_rsa.pub

(4) add the public key to the authorized_keys file

{dc02:~/.ssh} cat id_rsa.pub >> authorized_keys

// Now, since the private key is on the same directory you can ssh from any dcXX

// server to any other dcXX server without entering your password

Also, save the complete path to your project directory. It would be something like:

/people/cs/s/sxg122830/TestProj

To find out what the path is, run the ‘pwd’ command once you are inside your project directory on a dcXX machine.

This saved path would have to be put in your launcher script so that it will know the location of the project.

(5) Now, in order to do ssh from other linux systems (such as your local linux machine), just download the id_rsa file (not the id_rsa.pub) to your local linux system and put it in ~/.ssh directory. (This is important for the scripts to work!)

You can do this using the scp command e.g.

{:~} scp path_of_id_rsa_file/id_rsa .ssh/

The path can be obtained using pwd inside the .ssh directory on the dcXX machine.

(6) create a directory, say “launch”, on your local machine and put the config file in that directory.

(7) edit the launcher and cleanup scripts to put in your details. Also, give execute permissions to the scripts.

{:~} chmod +x launcher.sh cleanup.sh

(8) compile your project on the dc server & place your .class files in /bin directory or an appropriate place and edit the scripts accordingly

(9) launch your project. Script will work provided you are using a gnome-terminal on your linux system. If you use any other type of terminal, you will have to substitute the command to 'gnome-terminal –e' with what you would use to invoke a new terminal window on your machine.

{:~} ./launcher.sh

// program will run...

(10) run the cleanup script to delete any zombie processes if present on the dcXX machines

{:~} ./cleanup.sh