A Brief Introduction to Blue Waters

September 2, 2019

Contents

This is a brief documentation about how to use Blue Waters.

1 Connect to Blue Waters

1.1 Check your SSH client

For Window user, first run Command Prompt(cmd.exe) and inside Command Prompt run ssh. If there is no error message you can go to the next step. Otherwise, please refer the link to enable ssh client on Windows.
Similarly, for Mac user, if you cannot run ssh in your terminal,please refer the link to enable ssh client on your machine.

1.2 Logging in

Use the following command:

```
ssh YOUR_USER_ID@bwbay.ncsa.illinois.edu
```

An official documentation on how to logging in is in here.

2 Transfer Files

2.1 Transfer Files between your local machine and Blue Waters

You can copy files less than 1GB and directories less than 10GB using the command scp. This command must be run from a login node (which is the node you will be on when you first login) and cannot be run from your local machine.

Transfer from your local machine to BW

```
scp localuser@localIP:~/codeForBlueWaters.py ~/scratch/.
scp -r localuser@localIP:~/directoryForBluewaters ~/scratch/
```

Transfer from BW to your local machine

```
scp  codeForLocalMachine.py localuser@localIP:LocalMachineDirectory/.
scp -r DirectoryForLocalMachine localuser@localIP:LocalMachineDirectory/.
```

The first command is for copying a single file from your local machine to the scratch directory in Blue Waters. The second command is for copying an entire directory (the -r option is to do so recursively). Type ifconfig at the command line on your local machine to get your IP address. Another way is if you type exit at your Blue Waters login (user@h2ologin3) and then log back in, you will be prompted with a message such as the following:

Last login: Thu Sep 15 13:41:06 2018 from YOURIPADDRESS

If your Internet service provider(ISP) does not provide static IP, you can login to Engineering Workstations(EWS). And then transfer file between EWS and Blue Waters.

For files bigger than 1GB and/or directories bigger than 10GB (which should only really be datasets for this class), let the TAs or the professor know what it is you need to transfer and I believe we will have to do this ourselves since the student accounts do not allow it. We can put the datasets in the shared projects directory for the class so everyone can have access to them.

2.2 Copy Files from Shared Directory to Your Local Director on Blue Waters

The shared projects directory for the class is

```
/projects/training/bayw.
```

You can use the following command to copy the 'tutorial' folder into your scratch folder.

```
cp -r /projects/training/bayw/tutorial ~/scratch/.
```

Note 1: /scratch is different than ∼/scratch. The ∼ is shorthand for your own local directory where /scratch is on the root directory. For example, my ∼/scratch is actually located at /u/training/instr030/scratch. Do not try to do anything with the root /scratch.

Note 2: The scratch directory is supposed to be wiped occasionally (I believe every 30 days although I do not think I have ever actually seen it wiped). Its probably not a good idea to keep important files in here. You can scp files back from Blue Waters to your local computer for storage if they are small enough.

An official documentation on how to transfer data is in here, which includes how to setup WinSCP.

3 Run Job in Interactive Mode

3.1 Submit an Interactive Job

Once you login the Blue Waters, you can submit an interactive job.

Blue Waters batch system provide interactive mode for interactive debugging and/or optimization activities. Long running jobs or production runs should not be run from interactive sessions. To start an interactive session, use the following command.

```
qsub -I -l nodes=1:ppn=32:xe -l walltime=01:00:00
```

- -I: iterative job
- nodes=1:ppn=32:xe: 1 cpu node
- walltime=01:00:00: use maximum for 1 hour

The job will go into the queue. When the command returns, you'll be in a shell running on a "MOM" node. MOM node is a shared service resource that manages job execution. Use aprun command to send the job to compute nodes.

If the intended use of the interactive session is to work directly on a compute node (for example: to access the GPU in an XK node), consider using CCM mode, which provides the opportunity to do computations interactively on a dedicated compute node rather than on a shared MOM node.

```
qsub -I -l gres=ccm -l nodes=1:ppn=16:xk -l walltime=01:00:00
```

Once the request goes through, if you only requested one node, you will immediately be at a terminal for a PBS MOM node (user@nid###). Just like the login node, you cannot perform any calculations here as its against Blue Waterss usage policy.

Type the following commands to login to the compute node from the PBS MOM node:

```
module add ccm
ccmlogin
```

You will once again be at a terminal indicated by user@nid###. however, this number should be different than the previous number indicating the PBS MOM node. You are now at an actual compute node where computations can be performed.

An official documentation on how about interactive mode can be found in here and about ccm mode can be found in here.

3.2 Run Python on Interactive Mode

Once you are in the interactive mode, run the following command to load python into your environment.

```
module load python/2.0.1
```

Note: there are different Pythons available in Blue Waters, and we recommend python/2.0.1 because it contains Pytorch 0.4.0.

Once you successfully load Python, you can run the following command to run your Python code.

```
aprun -n 1 -N 1 python main.py
```

- -n :Number of processing elements PEs for the application
- -N :Number of PEs to place per node

4 Submit a Job

The procedure to submit a job is as following

1. create the job script(or called pbs script).
2. Submit the job script using qsub.

   ```
   qsub run.pbs
   ```

Here is an example of job script named run.pbs for GPU job.

```
#!/bin/bash
#PBS -l nodes=01:ppn=16:xk
#PBS -l walltime=02:00:00
#PBS -N your_job_name
#PBS -e $PBS_JOBID.err
#PBS -o $PBS_JOBID.out
#PBS -m bea
#PBS -M YOUR_NETID@illinois.edu

cd /u/training/instr030/code_directory

. /opt/modules/default/init/bash # NEEDED to add module commands to shell
module load python/2.0.1
#module load cudatoolkit
aprun -n 1 -N 1 python main.py
```

Here is an example of job script named run.pbs for CPU job.

```
#!/bin/bash
#PBS -l nodes=01:ppn=32:xe
#PBS -l walltime=02:00:00
#PBS -N your_job_name
#PBS -e $PBS_JOBID.err
#PBS -o $PBS_JOBID.out
#PBS -m bea
#PBS -M YOUR_NETID@illinois.edu

cd /u/training/instr030/code_directory

. /opt/modules/default/init/bash # NEEDED to add module commands to shell
module load python/2.0.1
#module load cudatoolkit
aprun -n 1 -N 1 python main.py
```

Here is some explanation on the job script.

- `#PBS -l nodes=01:ppn=16:xk`

  This is used to declare a GPU node. Here xk is to indicate it is a GPU node. if you plan to use CPU, you can use

  ```
  #PBS -l nodes=01:ppn=32:xe
  ```

- `#PBS -l walltime=02:00:00`

  This is used to set a limitation on the running time of your code. The job will stop at the time of min(your code running time, walltime)

- `#PBS -N your_job_name`

  This is to set your job name, which could be used when you check your job status.

- `#PBS -e $PBS_JOBID.err`

  This is to set your job error message file. If you code has some error, the error message will be shown up in this file.

- `#PBS -o $PBS_JOBID.out`

  This is to set your job output file. If you have print function in your code, the print output will be shown in this file.

- `#PBS -m bea`

- `#PBS -M YOUR_NETID@illinois.edu`

  This is to set an Email notification when your job begins to run and finish.

- `cd /u/training/instr030/code_directory`

  The script will run in the 'code_directory'. You can store this script anywhere, but this script will run the file in the directory you defined.

- `module load python/2.0.1`

- `aprun -n 1 -N 1 python main.py`

  This part is similar to the command when you run python in interactive mode.

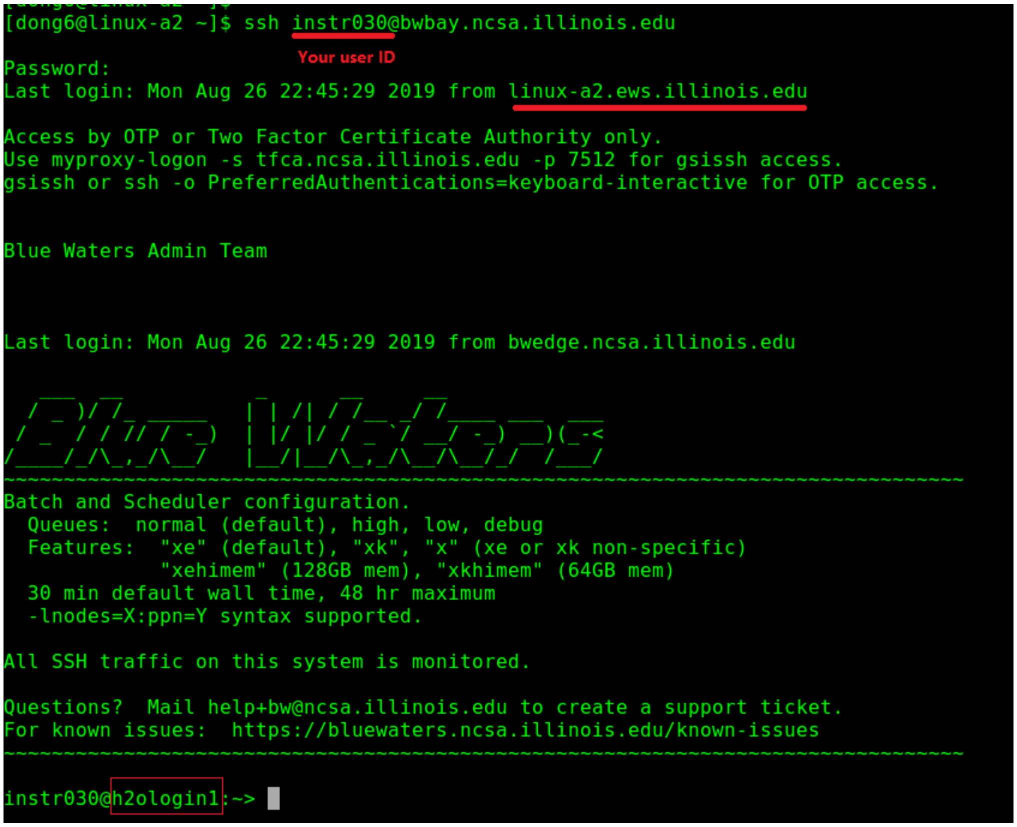
Note 1:Once you submitted a job, you can check its status by the following command.

```
qstat -u YOUR_BW_ID
```

Note 2: You can delete your job by the following command.

```
qdel XXXXXXX.bw
```

where XXXXXX is the job ID.
An official documentation on how to submit job can be found in here

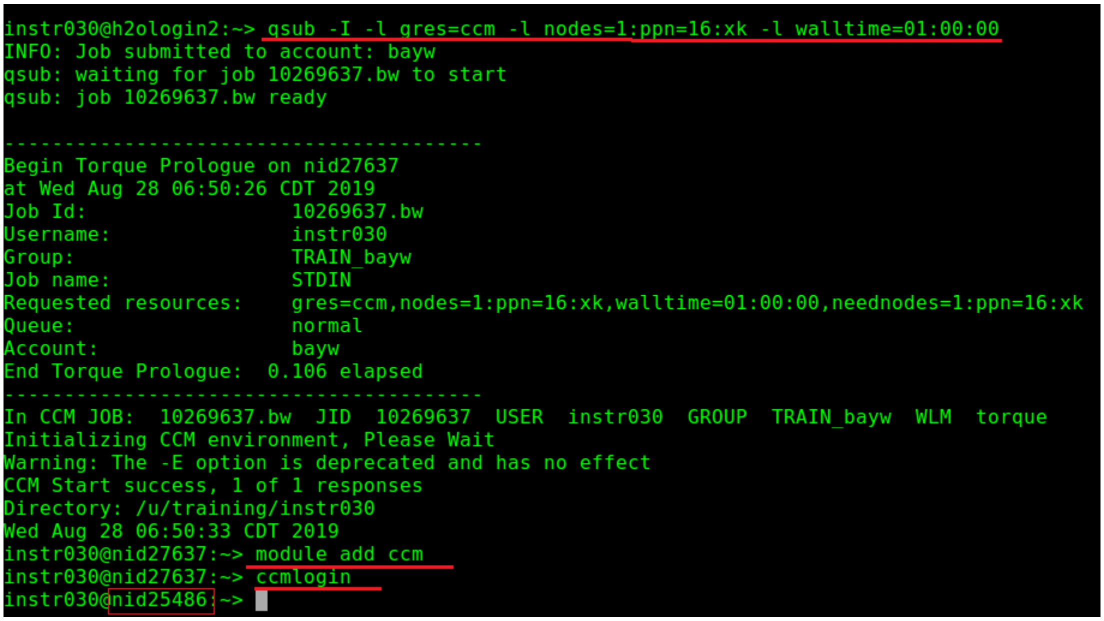


```
[dong6@linux-a2 ~]$
[dong6@linux-a2 ~]$ ssh instr030@bwbay.ncsa.illinois.edu

Password:
Last login: Mon Aug 26 22:45:29 2019 from linux-a2.ews.illinois.edu

Access by OTP or Two Factor Certificate Authority only.
Use myproxy-logon -s tfca.ncsa.illinois.edu -p 7512 for gsissh access.
gsissh or ssh -o PreferredAuthentications=keyboard-interactive for OTP access.


Blue Waters Admin Team



Last login: Mon Aug 26 22:45:29 2019 from bwedge.ncsa.illinois.edu

   ___  __            _      __     __
  / _ )/ /_ _____    | | /| / /__ _/ /____ _______
 / _  / / // / -_)   | |/ |/ / _ `/ __/ -_) __(_-<
/____/_/\_,_/\__/    |__/|__/\_,_/\__/\__/_/ /___/
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Batch and Scheduler configuration.
  Queues:  normal (default), high, low, debug
  Features:  "xe" (default), "xk", "x" (xe or xk non-specific)
             "xehimem" (128GB mem), "xkhimem" (64GB mem)
  30 min default wall time, 48 hr maximum
  -lnodes=X:ppn=Y syntax supported.

All SSH traffic on this system is monitored.

Questions?  Mail help+bw@ncsa.illinois.edu to create a support ticket.
For known issues:  https://bluewaters.ncsa.illinois.edu/known-issues
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

instr030@h2ologin1:~>
```

Your user ID

Figure 1: Screen shot on login Blue Waters in Linux

List of Figures



```
instr030@h2ologin2:~> qsub -I -l gres=ccm -l nodes=1:ppn=16:xk -l walltime=01:00:00
INFO: Job submitted to account: bayw
qsub: waiting for job 10269637.bw to start
qsub: job 10269637.bw ready

----------------------------------------
Begin Torque Prologue on nid27637
at Wed Aug 28 06:50:26 CDT 2019
Job Id:                 10269637.bw
Username:               instr030
Group:                  TRAIN_bayw
Job name:               STDIN
Requested resources:    gres=ccm,nodes=1:ppn=16:xk,walltime=01:00:00,neednodes=1:ppn=16:xk
Queue:                  normal
Account:                bayw
End Torque Prologue:  0.106 elapsed
----------------------------------------
In CCM JOB:  10269637.bw  JID  10269637  USER  instr030  GROUP  TRAIN_bayw  WLM  torque
Initializing CCM environment, Please Wait
Warning: The -E option is deprecated and has no effect
CCM Start success, 1 of 1 responses
Directory: /u/training/instr030
Wed Aug 28 06:50:33 CDT 2019
instr030@nid27637:~> module add ccm
instr030@nid27637:~> ccmlogin
instr030@nid25486:~> 
```

Figure 2: Screen shot on ccm login