Introduction to GACRC Teaching Cluster
PHYS8602

Georgia Advanced Computing Resource Center (GACRC)
EITS/University of Georgia
Zhuofei Hou  zhuofei@uga.edu
Outline

• GACRC

• Overview

• Computing Resources
  ➢ Three Folders
  ➢ Three Computational Queues
  ➢ Software

• Submit Batch Job

• GACRC Wiki and Support
We are a high-performance-computing (HPC) center at UGA

We provide to the UGA research and education community an advanced computing environment:

- HPC computing and networking infrastructure located at the Boyd Data Center
- Comprehensive collection of scientific, engineering and business applications
- Consulting and training services

http://wiki.gacrc.uga.edu (GACRC Wiki)
https://wiki.gacrc.uga.edu/wiki/Getting_Help (GACRC Support)
http://gacrc.uga.edu (GACRC Web)
Please note:
You need to connect to the UGA VPN when accessing from outside of the UGA main campus.

1. ssh with MyID and password
2. Verify with Archpass Duo two-factor authentication

- Node: Computer for a specific function on cluster, e.g., login node
- Queue: Collection of compute nodes for specific computing need
- Cluster: Nodes + Drives, all connected by network
Computing Resources

- Two Nodes:
  1. Login node (MyID@teach.gacrc.uga.edu): for submitting computational jobs
  2. Transfer node (MyID@txfer.gacrc.uga.edu): for transferring data files

- Three Directories:
  1. /home/MyID: working space for computational jobs
  2. /work/phys8602/MyID: data parking for individual user in the class
  3. /work/phys8602/instructor_data: data shared with class by the instructors

- Queue for your class: fsr8602
Computing Resources (cont.)

- Software
  1. Software names are long and have a Easybuild toolchain name associated to it
  2. Complete module name: Name/Version-toolchain, e.g., Python/2.7.14-foss-2016b
  3. Software names are case-sensitive!
     - module avail: List all available software modules installed on cluster
     - module load moduleName: Load a module into your working environment
     - module list: List modules currently loaded
     - module unload moduleName: Remove a module from working environment
     - ml spider pattern: Search module names matching a pattern (case-insensitive)
Submit Batch Job

1. Log on to Login node using MyID and password, and two-factor authentication with Archpass Duo: 
   
   ```bash
   ssh MyID@teach.gacrc.uga.edu
   ```

2. Create a working subdirectory for a job: `mkdir ./workDir`

3. Change directory to `workDir`:
   ```bash
   cd ./workDir
   ```

4. Transfer data from local computer to `workDir`: use `scp` or `SSH File Transfer` to connect Transfer node
   Transfer data on cluster to `workDir`: log on to Transfer node and then use `cp` or `mv`

5. Compile your source codes `phys8602_mult.c` into binary

6. Make a job submission script in `workDir`: `nano ./phys8602_sub.sh`

7. Submit a job from `workDir`: `sbatch ./phys8602_sub.sh`

8. Check job status: `squeue` or Cancel a job: `scancel JobID`
Step 1: Log on to Login node - Mac/Linux using ssh

1. Open **Terminal** utility

2. Type command line: `ssh MyID@teach.gacrc.uga.edu`

3. You will be prompted for your **MyID password**

4. Teaching cluster access requires ID verification using two-factor authentication with **Archpass Duo**. If you are not enrolled in Archpass Duo, please refer to

   [https://eits.uga.edu/access_and_security/infosec/tools/archpass_duo/](https://eits.uga.edu/access_and_security/infosec/tools/archpass_duo/) on how to enroll

More information: [https://wiki.gacrc.uga.edu/wiki/Connecting#Connecting_to_the_teaching_cluster](https://wiki.gacrc.uga.edu/wiki/Connecting#Connecting_to_the_teaching_cluster)
Step1 (Cont.) - Mac/Linux

Using ssh in Terminal!

4. Verify login using Duo

SSH Zhuofei@Teach.Gacrc.uga.edu

1. Log on

UGA DUO authentication is required for SSH/SCP access to GACRC systems. For additional help with UGA DUO authentication or to report an issue please visit: https://eits.uga.edu/access_and_security...

Password:  2. Enter your MyID password

When you enter password, no stars or dots will show as you are typing. Please type password carefully!

Duo two-factor login for Zhuofei

Enter a passcode or select one of the following options:

1. Duo Push to XXX-XXX-5758
2. Phone call to XXX-XXX-5758
3. Phone call to XXX-XXX-1925
4. SMS passcodes to XXX-XXX-5758 (next code starts with: 1)

Passcode or option (1-5): 1

Success. Logging you in...

Last login: Fri Aug 3 11:24:43 2018 from 172.22.72.35

[zhuofei@teach ~]$ 5. Logged on!
Step1 (Cont.) - Windows

1. Download and install SSH Secure Utilities: [http://eits.uga.edu/hardware_and_software/software/](http://eits.uga.edu/hardware_and_software/software/)

2. You can use PuTTY as an alternative: [https://www.putty.org/](https://www.putty.org/)
Step1 (Cont.) - Windows using SSH Secure Utilities

Please Note:
Authentication Method needs to be set as
Keyboard Interactive in default <profile Setting>
Step1 (Cont.) - Windows using SSH Secure Utilities

- Enter your UGA MyID password and click OK

Host Name: teach.gacrc.uga.edu
User Name: MyID
Port Number: 22
9. Enter “push” and click OK

10. Verify login using Duo
Step1 (Cont.) - Windows using SSH Secure Utilities

11. Click OK

12. Logged on!
Step 2 - 3: Create and change directory to workDir

```
[zhuofei@teach ~]$ ls
[zhuofei@teach ~]$ mkdir workDir
[zhuofei@teach ~]$ ls
workDir
[zhuofei@teach ~]$ cd workDir/
[zhuofei@teach workDir]$ ls
[zhuofei@teach workDir]$ ls
```

- `ls` command to list folder’s contents
- `mkdir` command to create a subdirectory
- `cd` command to change directory
- It is empty in workDir!
Step 4: Transfer data from local computer to workDir - Mac/Linux

1. Connect to Transfer node (MyID@txfer.gacrc.uga.edu) in Terminal on local computer
2. Type scp command: `scp (-r) [Source] [Target]`
3. Once you input MyID password, scp command will send “push” to your Duo Enrolled mobile device for verification

**E.g. 1:** use scp on local computer, from Local ➔ workDir on cluster

```bash
scp ./file zhuofei@txfer.gacrc.uga.edu:/home/zhuofei/workDir
scp -r ./folder/ zhuofei@txfer.gacrc.uga.edu:/home/zhuofei/workDir
```

**E.g. 2:** use scp on local computer, from workDir on cluster ➔ Local

```bash
scp zhuofei@txfer.gacrc.uga.edu:/home/zhuofei/workDir/file .
scp -r zhuofei@txfer.gacrc.uga.edu:/home/zhuofei/workDir/folder/ .
```

[https://wiki.gacrc.uga.edu/wiki/Transferring_Files#The_File_Transfer_node_for_the_teaching_cluster .28txfer.gacrc.uga.edu.29](https://wiki.gacrc.uga.edu/wiki/Transferring_Files#The_File_Transfer_node_for_the_teaching_cluster .28txfer.gacrc.uga.edu.29)
Step 4 (Cont.) - Windows using SSH Secure Utilities

Please Note:
Authentication Method needs to be set as Keyboard Interactive in default <profile Setting>
Step 4 (Cont.) - Windows using SSH Secure Utilities

4. Enter your UGA MyID password and click OK.

Steps 9 - 11 are the same as listed on page 13 - 14!
12. Logged on!

13. Click yellow button

14. Change local and remote paths
Step 4 (Cont.) - Windows using SSH Secure Utilities

15. Drag data between local computer and remote cluster.
Step 4 (Cont.): Transfer data on cluster to workDir

- Log on to Transfer node (MyID@txfer.gacrc.uga.edu)
  - Mac/Linux: ssh MyID@txfer.gacrc.uga.edu (page 8-9)
  - Windows: use SSH Secure Client app (page 14-16)
- Directories you can access on txfer:
  1. /home/MyID (Landing home)
  2. /work/phys8602/MyID
  3. /work/phys8602/instructor_data
- Transfer data between two folders on cluster using cp or mv, e.g.:
  
  mv /work/phys8602/MyID/datafile /home/MyID/workDir
Step 5: Compile your Fortran program `phys8602_mult.c` into binary

```
[zhufei@teach ~]$ cat phys8602_mult.c
/* Multiply two integer numbers */
#include <stdio.h>
int main(void)
{
int i=3, j=4, iprod;
FILE *fp;
fp = fopen("output.txt","w");
iprod=i*j;
fprintf(fp, "The product of %d and %d is %d \n", i,j,iprod);
fclose(fp);
return 0;
}
[zhufei@teach ~]$ module load PGI/17.9
[zhufei@teach ~]$ pgcc phys8602_mult.c -o phys8602_mult.x
[zhufei@teach ~]$ ./phys8602_mult.x
```

Note:
`phys8602_mult.c` is put in `/usr/local/training/phys`
You can copy it into your working directory for use

- load PGI compilers
- compile into binary
- run binary
Step 6: Make a job submission script `phys8602_sub.sh`

```bash
#!/bin/bash

#SBATCH --job-name=testJob
#SBATCH --partition=fsr8602
#SBATCH --ntasks=1
#SBATCH --cpus-per-task=1
#SBATCH --mem=2gb
#SBATCH --time=00:10:00
#SBATCH --output=log.%j
#SBATCH --mail-type=END,FAIL

cd $SLURM_SUBMIT_DIR

time ./phys8602_mult.x
```

- `#SBATCH --job-name=testJob`: # Job name
- `#SBATCH --partition=fsr8602`: # Partition (queue) for PHYS8602
- `#SBATCH --ntasks=1`: # Single task job
- `#SBATCH --cpus-per-task=1`: # Number of cores per task
- `#SBATCH --mem=2gb`: # Total memory for job
- `#SBATCH --time=00:10:00`: # Time limit hrs:min:sec; TIMELIMIT 10 min
- `#SBATCH --output=log.%j`: # Standard output and error log
- `#SBATCH --mail-user=MyID@uga.edu`: # Where to send mail
- `#SBATCH --mail-type=END,FAIL`: # Mail events (BEGIN, END, FAIL, ALL)

Note:
`phys8602_sub.sh` is put in `/usr/local/training/phys` 
You can copy it into your working directory for use

More Information: [https://wiki.gacrc.uga.edu/wiki/Running_Jobs_on_the_teaching_cluster](https://wiki.gacrc.uga.edu/wiki/Running_Jobs_on_the_teaching_cluster)
Step7: Submit a job from workDir using sbatch

```
$ sbatch phys8602_sub.sh
Submitted batch job 139
```

**Tips:** sub.sh is a job submission script for

1. specifying computing resources
2. loading software using `module load`
3. running any Linux commands you want to run
4. running your compiled binary
Step8: Check job status using squeue

```
$ squeue -l
Wed Aug  8 13:40:02 2018

<table>
<thead>
<tr>
<th>JOBID</th>
<th>PARTITION</th>
<th>NAME</th>
<th>USER</th>
<th>STATE</th>
<th>TIME</th>
<th>TIME_LIMI</th>
<th>NODES</th>
<th>NODELIST</th>
</tr>
</thead>
<tbody>
<tr>
<td>162</td>
<td>fsr8602</td>
<td>testJob</td>
<td>zhuofei</td>
<td>PENDING</td>
<td>0:00</td>
<td>00:10:00</td>
<td>1</td>
<td>(None)</td>
</tr>
<tr>
<td>160</td>
<td>fsr8602</td>
<td>testJob</td>
<td>zhuofei</td>
<td>RUNNING</td>
<td>0:02</td>
<td>00:10:00</td>
<td>1</td>
<td>c2-11</td>
</tr>
<tr>
<td>161</td>
<td>fsr8602</td>
<td>testJob</td>
<td>zhuofei</td>
<td>RUNNING</td>
<td>0:02</td>
<td>00:10:00</td>
<td>1</td>
<td>c2-11</td>
</tr>
</tbody>
</table>

$ squeue

<table>
<thead>
<tr>
<th>JOBID</th>
<th>PARTITION</th>
<th>NAME</th>
<th>USER</th>
<th>ST</th>
<th>TIME</th>
<th>NODES</th>
<th>NODELIST</th>
</tr>
</thead>
<tbody>
<tr>
<td>162</td>
<td>fsr8602</td>
<td>testJob</td>
<td>zhuofei</td>
<td>PD</td>
<td>0:15</td>
<td>1</td>
<td>(None)</td>
</tr>
<tr>
<td>160</td>
<td>fsr8602</td>
<td>testJob</td>
<td>zhuofei</td>
<td>R</td>
<td>0:17</td>
<td>1</td>
<td>c2-11</td>
</tr>
<tr>
<td>161</td>
<td>fsr8602</td>
<td>testJob</td>
<td>zhuofei</td>
<td>R</td>
<td>0:17</td>
<td>1</td>
<td>c2-11</td>
</tr>
</tbody>
</table>
```

Common STATE: R for Running; PD for Pending; TO for TimedOut; S for Suspended; F for FAILED

TIME: the elapsed time used by the job, not remaining time, not CPU time.
Step8 (Cont.): Cancel job using scancel

```
$ squeue -l
Wed Aug  8 14:03:47 2018
JOBID PARTITION   NAME   USER    STATE   TIME   TIME_LIMI   NODES NODELIST
169 fsr8602 testJob zhuofei RUNNING 0:07 00:10:00   1 c1-38
168 fsr8602 testJob zhuofei RUNNING 0:10 00:10:00   1 c1-39

$ scancel 169

[zhuofei@teach workDir]$ squeue -l
Wed Aug  8 14:03:47 2018
JOBID PARTITION   NAME   USER    STATE   TIME   TIME_LIMI   NODES NODELIST
169 fsr8602 testJob zhuofei COMPLETI 0:15 00:10:00   1 c1-39
168 fsr8602 testJob zhuofei RUNNING 0:18 00:10:00   1 c1-38

$ squeue -l
Wed Aug  8 14:04:08 2018
JOBID PARTITION   NAME   USER    STATE   TIME   TIME_LIMI   NODES NODELIST
168 fsr8602 testJob zhuofei RUNNING 0:35 00:10:00   1 c1-38
```
Step 8 (Cont.): Check job details using scontrol show job

$ scontrol show job 174

JobId=174  JobName=testJob
UserId=zhuofei(1772)  GroupId=gacrc-instruction(21004)  MCS_label=N/A
JobState=RUNNING  Reason=None  Dependency=(null)
Requeue=1  Restarts=0  BatchFlag=1  Reboot=0  ExitCode=0:0
RunTime=00:00:28  TimeLimit=00:10:00  TimeMin=N/A
SubmitTime=2018-08-08T14:28:44  EligibleTime=2018-08-08T14:28:44
StartTime=2018-08-08T14:28:44  EndTime=2018-08-08T16:28:44  Deadline=N/A
...  
Partition=fsr8602  AllocNode:Sid=teach:30986
NodeList=c1-38
NumNodes=1  NumCPUs=1  NumTasks=1  CPUs/Task=1  ReqB:S:C:T=0:0:*:*:*
...  
Command=/home/zhuofei/workDir/phys8602_sub.sh
WorkDir=/home/zhuofei/workDir
StdErr=/home/zhuofei/workDir/log.174
StdOut=/home/zhuofei/workDir/log.174
Step8 (Cont.): Check node info using sinfo

```
$ sinfo

<table>
<thead>
<tr>
<th>PARTITION</th>
<th>AVAIL</th>
<th>TIMELIMIT</th>
<th>NODES</th>
<th>STATE</th>
<th>NODELIST</th>
</tr>
</thead>
<tbody>
<tr>
<td>highmem</td>
<td>up</td>
<td>7-00:00:00</td>
<td>5</td>
<td>idle</td>
<td>c1-[36-37,40],c2-[9-10]</td>
</tr>
<tr>
<td>gpu</td>
<td>up</td>
<td>1-00:00:00</td>
<td>1</td>
<td>down*</td>
<td>c2-2</td>
</tr>
<tr>
<td>interq</td>
<td>up</td>
<td>1-00:00:00</td>
<td>3</td>
<td>idle</td>
<td>c2-[4-6]</td>
</tr>
<tr>
<td>batch</td>
<td>up</td>
<td>7-00:00:00</td>
<td>39</td>
<td>idle</td>
<td>c1-[1-35,38-39],c2-[11-12]</td>
</tr>
<tr>
<td>fsr8602</td>
<td>up</td>
<td>10:00</td>
<td>39</td>
<td>idle</td>
<td>c1-[1-35,38-39],c2-[11-12]</td>
</tr>
<tr>
<td>fsr4601</td>
<td>up</td>
<td>1:00</td>
<td>39</td>
<td>idle</td>
<td>c1-[1-35,38-39],c2-[11-12]</td>
</tr>
</tbody>
</table>
```

idle = no cores in use; mix = some cores are still free; alloc = all cores are allocated
GACRC Wiki [http://wiki.gacrc.uga.edu](http://wiki.gacrc.uga.edu)

Running Jobs: [https://wiki.gacrc.uga.edu/wiki/Running_Jobs_on_the_teaching_cluster](https://wiki.gacrc.uga.edu/wiki/Running_Jobs_on_the_teaching_cluster)

Software: [https://wiki.gacrc.uga.edu/wiki/Software](https://wiki.gacrc.uga.edu/wiki/Software)

Transfer File:

[https://wiki.gacrc.uga.edu/wiki/Transferring_Files#The_File_Transfer_node_for_the_teaching_cluster](https://wiki.gacrc.uga.edu/wiki/Transferring_Files#The_File_Transfer_node_for_the_teaching_cluster)

luster_28txfer.gacrc.uga.edu.29

Linux Command: [https://wiki.gacrc.uga.edu/wiki/Command_List](https://wiki.gacrc.uga.edu/wiki/Command_List)

Training: [https://wiki.gacrc.uga.edu/wiki/Training](https://wiki.gacrc.uga.edu/wiki/Training)
GACRC Support
https://uga.teamdynamix.com/TDClient/Requests/ServiceCatalog?CategoryID=11593

- **Job Troubleshooting:**
  
  Please tell us details of your question or problem, including but not limited to:
  
  - Your user name
  - Your job ID
  - Your working directory
  - The queue name and command you used to submit the job

- **Software Installation:**
  
  - Specific name and version of the software
  - Download website
  - Supporting package information if have

Please note to make sure the correctness of datasets being used by your jobs!
GACRC Service Catalog

Services (11)

**Account Creation**
For a research group’s PI to request user accounts for group members on the GACRC computing systems.

**Class Account Creation**
For an instructor to request user accounts for students attending a course that will need to use GACRC computing systems.

**Class Account Modification**
For instructors to request changes to be made in previously requested class account.

**Computing Lab Modification/Deletion**

**General Internal**

**General Support**
Report issues and request help with GACRC systems, except for software installation requests and account/lab creation requests.

**Lab Creation**
For a research group’s PI to register a computing lab on the GACRC computing systems.

**Modify/Delete Account**
For PIs to request changes in or deletion of user accounts on GACRC computing systems.

**Software Installation/Update**
Request software and common application database (e.g. NCBI blast databases) installation and upgrade.
Thank You!