HPC–101
Onboarding

Director: Gwen Jacobs
Sean Cleveland, Adriana Comerford, Jennifer Geis,
Ron Merrill, David Schanzenbach

Information Technology Services
Cyberinfrastructure
University of Hawai'i

https://www.hawaii.edu/its/ci/
uh-hpc-help@lists.hawaii.edu

June 21, 2019

Overview

Part I – Introduction:

1. Terminology
2. University of Hawai'i High Performance Compute Cluster

Part II – Using the UH-HPC:

1. Connecting via SSH
2. Directories & Centralized Software
3. Job Scheduler

Part III – High Level View of Policies:

1. UH Policies
2. UH-HPC Usage Policies

Part I

Introduction

Introduction – Overview

1. Terminology

2. University of Hawai'i High Performance Compute Cluster
   - Condo Program & Leasing
   - Storage
   - Networking
   - User Support

Terminology

- **Node** – Another name for a server or computer
- **Login node** – A specialized node that users connect to in order to submit work to a computer cluster
- **Computer cluster** – A set of loosely or tightly connected nodes that work together so that, in many respects, they can be viewed as a single system[1]
- **Data transfer node (DTN)** – Specialized nodes that minimize the impedance on the network to access the full capability of the network
- **Science DMZ (SciDMZ)** – A portion of the network configured with equipment and security policies in order to optimize for high-performance scientific applications rather than for general-purpose business systems or "enterprise" computing
- **Multi-factor Authentication (MFA)** – An authentication method in which a computer user is granted access only after successfully presenting two or more pieces of evidence or factors to an authentication mechanism, e.g., DUO

[1] https://en.wikipedia.org/wiki/Computer_cluster

Terminology

- **Symbolic Link (symlink)** – A file that contains a reference to another file or directory
- **Command-line interface/interpreter (CLI)** – A text-based user interface used to view and manage computer files
- **Message Passing Interface (MPI)** – A standard that is used by programs to pass messages between nodes
- **High Performance Compute (HPC)** – A computing paradigm in which applications are typically a tightly coupled parallel job that benefit from a low-latency interconnect
- **High Throughput Compute (HTC)** – A computing paradigm that focuses on the efficient execution of a large number of loosely-coupled tasks
- **Modified time** – The last time the file was modified (content has been modified)[2]
- **Shell script (script)** – A computer program designed to be run by a CLI[3]

[2] https://unix.stackexchange.com/a/2465
[3] https://en.wikipedia.org/wiki/Shell_script

1. Terminology

2. University of Hawai'i High Performance Compute Cluster
   - Condo Program & Leasing
   - Storage
   - Networking
   - User Support

University of Hawai'i High Performance Compute Cluster

- The University of Hawai'i High Performance Compute Cluster (UH-HPC) is **free** to use for all active faculty, staff, and students affiliated with the University of Hawai'i
- Community acquired nodes are equally accessible to all users
- Nodes purchased through the **condo program** are shared with the community, but priority is given to the node owner and their agents
- Nodes may be **leased** from the community pool
- Additional permanent storage can be leased with up to a five year contract by faculty & staff

UH-HPC Resource Summary

	Nodes	CPU Cores	Memory	GPUs	Home/Group Space	Scratch Space	Storage for lease
Total	297	6,308	50 TB	56	80 TB	700 TB	1 PB

What is the Condo Program & leasing?

Condo Program

- The condo program allows faculty & staff to purchase nodes and have them integrated with the UH-HPC
- Condo nodes can take advantage of networking and storage infrastructure that one may not typically have access to
- Condo nodes are managed and maintained by ITS staff
- No maintenance fee will be assessed until the node is off warranty (typically five years)

Leasing

- Nodes leased from the community can have a contracts period from one month, up to one year
- Node lessees are provided priority access to their leased hardware
- Leases are not considered an equipment purchase

Storage

The UH-HPC has two classes and three types of storage that users can potentially access. Each type of storage has their own attributes and restrictions

Free Storage

1. Permanent Storage
   - Home Storage
   - Group/Lab Storage
2. Scratch Storage
   - Lustre®(Aug 2019)
   - Network File System (NFS)

For Fee Storage

1. ValueStorage
2. Long Term Storage (LTS)

Permanent Storage

1. Home Storage
   - **Purpose**: Personal storage for applications and active data that needs to persist on the UH-HPC
2. Group/Lab Storage
   - **Purpose**: Group/Lab storage allows users to share data & applications with a need for persistence on the UH-HPC

All permanent storage options on the UH-HPC have the following attributes:

- 50 GB default quota with a max quota of 300 GB
- Quota increases are re-evaluated annually
- Available on all nodes
- Freely available to users

Scratch Storage

1. Lustre®(Aug 2019)
   - High performance parallel file system
   - 50 TB quota
2. NFS
   - Not a parallel file system
   - Potentially better performance for random I/O patterns
   - 5 TB quota

All scratch file systems on the UH-HPC have the following attributes:

- **Purge policy** – 10 days based on file modify time
- Available on all nodes
- Freely available to users

Networking

Internal Networks

- Quad Data Rate (QDR) InfiniBand® (IB)
  - 40 Gbit
  - Older compute nodes
  - low latency ($\approx 1.3\mu$s)
  - non-blocking
- 25/100 Gbit Ethernet
  - Newer compute nodes
  - Nodes connected @ 25 Gbit
  - non-blocking

External Networks

- 100 Gbit SciDMZ
  - Login & compute nodes connected via a firewall
  - DTNs are directly connected

User Support

Online documents & FAQ

Users are encouraged to look through the online documentation & FAQ prior to contacting ITS-CI directly. Many questions we receive are repeat questions and we try to capture them in our FAQ

xCAT cluster information, policies & FAQ

Contact Information

If your question is not answered in our online documentation, please contact us at: UH-HPC-Help@lists.hawaii.edu

- For batch jobs . . .
  - Job ID, path to submission script, submission command, error file location, output files
- For other problems . . .
  - State the problem, command issued, host, directory, remote host, error messages

Part II

Using the UH-HPC

Connecting via SSH

Requirements

- Valid UH credentials
- Registered for MFA/DUO
- Familiarity with a SSH client & a file-transfer method
- Comfortable with the CLI

Connection Information

- Login node:
  - **uhhpc.its.hawaii.edu**
- DTNs:
  - **hpc-dtn1.its.hawaii.edu**
  - **hpc-dtn2.its.hawaii.edu**

Valid Credentials

- Your UH user name
- Accepted forms of authentication
  - UH Password + MFA
  - SSH key + MFA

Try and connect to the UH-HPC Login node now using your SSH client

User Directories

Filesystem

```
[testuser@login001 ~]$ df -h
Filesystem                                  Size  Used Avail Use% Mounted on
fs01:/mnt/datastore/hpc/home/testuser        50G  128K   50G   1% /home/testuser
fs02:/mnt/datastore/hpc/scratch/testuser    5.0T  256K  5.0T   1% /mnt/scratch/nfs_fs02/testuser
```

Home

```
[testuser@login001 ~]$ ls -l
total 1
drwxr-xr-x 3 testuser testuser 23 Jan 15 20:38 examples
lrwxrwxrwx 1 testuser testuser 19 Jan 15 20:38 nfs_fs02 -> /mnt/scratch/nfs_fs02/testuser
```

- ~/examples contains example scripts to use as templates
- ~/nfs_fs02 is a symlink to the NFS scratch file system

Modules

A tool to help users manage their Unix or Linux shell environment, by allowing groups of related environment-variable settings to be made or removed dynamically.[4]

Commands

- 'module avail' – list installed modules
- 'module show <module name>' – Show what actions a module performs
- 'module load <module name>' – Loads the named module
- 'module spider <search string>' – search the modules list for a match
- 'module list' – Show what modules are loaded
- 'module purge' – Unload all loaded modules

- Hidden modules can be shown using the --show_hidden flag, e.g., 'module --show_hidden avail'
- We create modules for frequently requested software packages for all users to access
- Compilers, libraries, interpreters, applications are all added as modules
- Users are encouraged to install software in their home/group directories
- Modules can be listed on the login nodes, but loaded applications will only work on the compute nodes
- The UH-HPC currently uses lmod

[4] https://en.wikipedia.org/wiki/Environment_Modules_(software)

Terminology

- **Job Scheduler** – A tool/application to control and prioritize the execution order of unrelated jobs
- **Job** – Another name for a script or application that is to be executed
- **Job ID** – A number assigned to each job submitted to the job scheduler
- **CPU/Socket** – A processing unit in the node which may contain one or more cores
- **Core** – A processing element on a CPU (Multi-threading)
- **Task** – An instance of a running program or process (MPI)
- **Partition** – A group of nodes divided into possibly overlapping sets, which also contains constraints for the given set of nodes

---

http://slurm.schedmd.com/quickstart.html

SLURM – **S**imple **L**inux **U**tility for **R**esource **M**anagement

The UH-HPC uses the SLURM job scheduler to allocate nodes and assign jobs to them

How it works

Jobs are not executed in a **first in first out** manner. Instead, jobs are assigned a priority, which is continuously being re-evaluated for pending jobs.

Depending on load, some resources may go idle while waiting for sufficient free resources for a higher priority job. In these cases, the scheduler will use what is known as **backfilling** to fill in the idle machines with jobs that will not affect the start time of higher priority jobs.

---

https://en.wikipedia.org/wiki/Slurm_Workload_Manager

http://slurm.schedmd.com/slurm.html

Commands

Basic

- ***sbatch*** – Used to submit a job script for later execution
- ***srun*** – Used to submit a job for execution or initiate job steps in real time
- ***scancel*** – Used to cancel a pending or running job or job step

Informational

- ***squeue*** – Reports the state of jobs or job steps
- ***sinfo*** – Reports the state of partitions and nodes managed by Slurm
- ***sacct*** – Reports job accounting information about active or completed jobs

– Examples usage of the SLURM commands can be seen on schedmd's quickstart

http://slurm.schedmd.com/quickstart.html

Interactive jobs using srun

Command

```
[login ~]$ srun -I30 -p sandbox -N 1 -c 1 --mem=6G -t 0-01:00:00 --pty /bin/bash
```

Options

- **-I30** – exit if resources are not available within the time period specified (30 seconds)
- **-p sandbox** – Submit my interactive job to the sandbox partition
- **-N 1** – Number of nodes requested (If omitted, default is 1)
- **-c 1** – Number of cores per task requested (If omitted, default is 1)
- **--mem=6G** –Memory allocated per node (See partition details for defaults)
- **-t 0-01:00:00** – How much time you are requesting (DD-HH:MM:SS)
- **--pty** – Execute initial task in pseudo terminal mode
- **/bin/bash** – Task to execute

Interactive jobs terminate when the specified time has elapsed or if you give the **exit** command.
Interactive jobs are good for testing, compiling and relatively short jobs.
Longer jobs should use a shell script and **sbatch**.

Batch job using sbatch

Command

```
[login ~]$ sbatch <path to shell script>
```

Info

- Where sbatch is executed, becomes the jobs working directory
- Submission scripts are shell scripts that begin with special comments that are parameters for the scheduler
- Parameters are evaluated with the command-line taking precedent over what the shell script contains
- Jobs submitted with sbatch are assigned a job ID by SLURM

Please navigate to **~/examples/slurm/non_mpi** and try to submit the example batch submission script uising sbatch.

Example Batch Job Script

```
[login001 nfs_fs02]$ cat example.slurm

#!/bin/bash
# Comments (#) and empty lines are fine between #SBATCH
#SBATCH --job-name=example
#SBATCH --partition=sandbox
#SBATCH --time=0-04:00:00 ## time format is DD-HH:MM:SS
# task-per-node x cpus-per-task should not exceed core count on an individual node
#SBATCH --nodes=1
#SBATCH --tasks-per-node=1
#SBATCH --cpus-per-task=20
#SBATCH --cpu-specs=0 # Allow access to all cores on a node
#SBATCH --mem=64G # Memory per node my job requires
#SBATCH --distribution="*:*:*" # set the task and core distribution to the defaults
#SBATCH --constraint="x86"
##SBATCH --constraint="x86&ib_qdr" # Used for MPI jobs that requires inter-node communication via IB
##SBATCH --gres=gpu:NV-K40:2 # commented out
#SBATCH --error=example-%A.err # %A - filled with jobid, where to write the stderr
#SBATCH --output=example-%A.out # %A - filled with jobid, wher to write the stdout
## Useful for remote notification
#SBATCH --mail-type=BEGIN,END,FAIL,REQUEUE,TIME_LIMIT_80
#SBATCH --mail-user=user@test.org
# All options and environment variables found on schedMD site: http://slurm.schedmd.com/sbatch.html
# =============== Start of commands to execute ===============
# source ~/.bash_profile # Not required unless you need something from your environment
export OMP_NUM_THREADS=${SLURM_CPUS_PER_TASK}
module load lang/R  # load the default R software module
Rscript hello.r
```

Partitions

Details

Partition	Max walltime	Jobs - total(running)	Max nodes per job	Default memory	Shared	Preemption
sandbox	0-04:00:00	∞	2	512 MB	YES	NO
shared	3-00:00:00	∞	1	512 MB	YES	NO
shared-long	7-00:00:00	5(2)	1	512 MB	YES	NO
exclusive	3-00:00:00	∞	20	∞	NO	NO
exclusive-long	7-00:00:00	5(2)	20	∞	NO	NO
kill-shared	3-00:00:00	∞	1	512 MB	YES	YES
kill-exclusive	3-00:00:00	∞	20	∞	NO	YES

Node Breakdown

Partition	Intel x86	GPU	IB	Ethernet	Min:Max Cores per node	Min:Max Memory per node
sandbox	4	0	4	0	20:20	128:128 GB
shared	116	1	116	0	20:40	128:1024 GB
shared-long	116	1	116	0	20:40	128:1024 GB
exclusive	116	1	116	0	20:40	128:1024 GB
exclusive-long	116	1	116	0	20:40	128:1024 GB
kill	24	8	17	7	20:20	96:128 GB
kill-exclusive	24	8	17	7	20:20	96:128 GB

Constraints & General Resources

--constraint

Nodes have features assigned to them by the administrators. Users can specify which of these features are required by their job using the constraint option. Only nodes having features matching the job constraints will be used to satisfy the request. Multiple constraints may be specified with "&" (AND), "|" (OR), etc.

--gres

Specifies a comma delimited list of generic consumable resources which a job should be granted access to.

Constraints & General resources to Node ID

Node range	Constraint
lmem-[0001-0005], node-[0001-0067,0081-0143]	x86, intel, ivy-bridge, ib_qdr
gpu-[0001-0002]	x86, intel, haswell, nvidia, tesla, kepler, ib_qdr
gpu-[0003-0009]	x86, intel, skylake, nvidia, turing, geforce, eth, eth_25

Node range	Gres [type:desc:count]
gpu-[0001-0002]	gpu:NV-K40:2
gpu-[0003]	gpu:NV-RTX2080Ti:4
gpu-[0004-0008]	gpu:NV-RTX2080Ti:8
gpu-[0009]	gpu:NV-RTX2070:8

Reserved Resources

On the UH-HPC, the scheduler by default withholds 1 core per node from use by users through a feature in SLURM known as **core specialization**.

Definition

Core specialization is a feature designed to isolate system overhead (system interrupts, etc.) to designated cores on a compute node. This can reduce applications interrupts ranks to improve completion time. [1] [2]

Override

--core-spec=0: In some cases, users may find through testing that using all the cores on a node show no degradation in performance. The user is able to override the 1 core reservation and utilize all cores on a node. Please note, that when this options is used, the node is placed into **exclusive mode not allowing other jobs to be scheduled along side it**.

Only use this option if you are allocating all cores on a node!

[1] https://slurm.schedmd.com/core_spec.html

[2] https://slurm.schedmd.com/SUG14/process_isolation.pdf

Part III

High Level View of Policies

High Level View of Policies – Overview

1. UH Policies

2. UH-HPC Usage Policies
   - Login nodes & DTN nodes
   - Storage Policy
   - User Account Life Cycle

UH Policies

Chapter 708, Hawaii Revised Statutes

- Access only by authorized people
- Should not be used in the act of committing a crime

University of Hawaii Executive Policy E2.210

- Protect your password (we will never ask or require your password)
- Computer resources should not be used to test or compromise systems without prior authorization
- University resources are intended to be used for institutional purposes and may not be used for private gain

High Level View of Policies – Overview

1. UH Policies

2. UH-HPC Usage Policies
   - Login nodes & DTN nodes
   - Storage Policy
   - User Account Life Cycle

Login nodes & DTN nodes

The following actions are acceptable on the shared systems in the UH-HPC:

1. File/Directory management [Login & DTN nodes]
2. Text editing with a text editor: vi/vim, emacs, nano [Login & DTN nodes]
3. Transferring files to and from the cluster (scp, rsync, SFTP, globus, aspera, lftp, etc.) [Login & DTN nodes]
4. Shall be used to submit batch and interactive jobs [Login nodes]
5. SSH shell access [Login & DTN nodes]

All other action shall take place on a compute node using an interactive session. If any actions outside of the sanctioned activities are detected, the following escalation will take place:

1. The process will be killed without notice
2. If the user continues this action, we will notify the user of violating the policy
3. If the user continues to ignore our warnings, the user may be banned from the cluster for a duration of time

Storage Policy

1. ITS is not responsible for any data that is deleted or loss due to user error, hardware failure, administrator error. Users are responsible for their own data and are highly encouraged to backup data that is important to them at other storage locations
2. Files located on the scratch file systems are subject to a **10 day** purge policy that is based on the file/directory modified timestamp. **Files that are purge cannot be recovered**. Users are encouraged to copy their results off the scratch file system as soon as possible upon completion of their job
3. Home storage currently snapshots once a day per user in case of accidental file deletion

User Account Life Cycle

1. Accounts that are idle for 120 days will be locked
2. Accounts that have been locked can be unlocked upon request with no further action then emailing us a request to unlock their account
3. Accounts that are idle for a total of 180 days will be purged from the cluster
4. Users that have been purged may request a reactivation of their account, but may be required to take a quiz to confirm retention of how to use the UH-HPC
5. Users that become **Ohana** (graduated or no longer affiliated with UH) will be allowed up to one year on the UH-HPC before their accounts will be purged
6. Ohana accounts can be purged sooner based on:
   - Standard idle lockout/purge policy
   - A faculty or staff requesting a graduated student's account be removed sooner than the one year grace period
7. Potential users whom are UH Ohana at the time of account request/creation may not have an account created on the UH-HPC without confirmation from an active faculty member or staff

Where to find all UH-HPC specific policies

1. Chapter 708, Hawaii Revised Statutes
2. University of Hawaii Executive Policy E2.210
3. Common systems & Storage
4. User Account life cycle
5. Security
6. For lease storage
7. Condo Program & Off Warranty condo nodes

Questions?