HPC Help and Security

Best Practices and Common Sense for Security with HPC

Thursday, February 15, 2024

https://bit.ly/COMPLECS

SDSC
SAN DIEGO SUPERCOMPUTER CENTER

UC San Diego

About COMPLECS

- COMPLECS (COMPrehensive Learning for end-users to Effectively utilize CyberinfraStructure) is a new SDSC program where training will cover non-programming skills needed to effectively use supercomputers. Topics include parallel computing concepts, Linux tools and bash scripting, security, batch computing, how to get help, data management and interactive computing

COMPLECS is supported by NSF award 2320934.

Table of contents

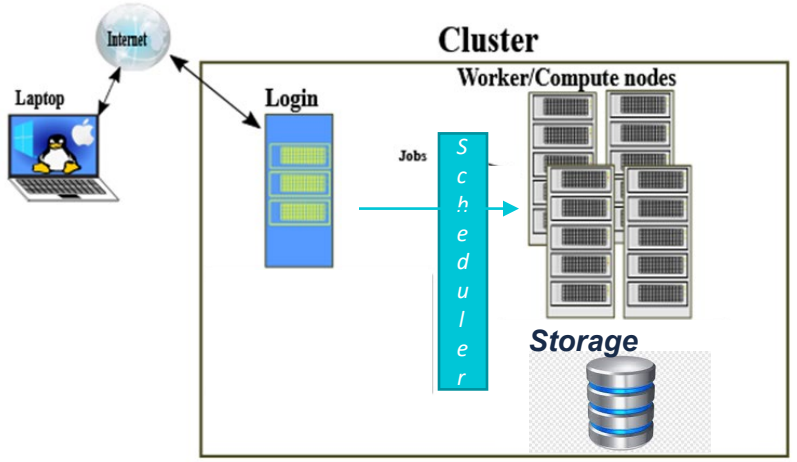
Help: Agenda

- Reduce Stress working with HPC
- Efficiently Identify and Effectively Utilize available resources to improve HPC experience.
  - Identify available resources
  - Identify the most appropriate resource
  - Create a effective Help ticket

HPC ecosystem: Shared Resource

- ACCESS serves
  - 1718 active users
  - 629 Active Allocations
  - 415 institutions
  - 73 fields of science
  - 14 resources
  - ~10 FTE Supporting staff per resource
- HPC/HTC resources includes the compute resources, memory, networking, storage, file systems and support staff
- HPC/HTC resources serve a ever expanding community

What all is shared



- CI Resources
  - Compute, Storage
    - Login node is shared with all users logged in
    - Compute nodes
      - may be different architecture from login nodes (there may be several different type of compute nodes that can be requested)
        - » sinfo command to see differences
      - Compute nodes are limited to users running batch jobs (shared, non-shared nodes)
      - Implications for performance
      - memory
      - Implications for charging
    - Storage & Filesystems
  - Security
  - Help/Support

Help: Workflow

- Understand you problem
  - Identify the issue
    - my code won't run, I can't log in, I can't access my files
  - Identify the nature of the issue
    - Is the specific error message
    - Is it a running job, has it run before, by you or someone else, on this system, other system, time frame
    - Is it an account/allocation setup
    - Is something missing or modified unexpectedly?
    - Performance
  - Identify the cause of the issue (escalation)
    - Is it code dependent
    - Is it system dependent
    - Is it account/allocation dependent
  - Identify the correct support option
  - Identify the solution

What are the available Help options?

- What are the available resources
  - User
  - Project/Colleagues
  - Web (Community forums, User guides, Git repositories)
  - Helpdesk

Help: Available Resources
Identify the most appropriate resource

- Internal
  - Yourself, Colleagues
    - Sanity check, Legacy Lab Code
  - The Web, User guide*****
    - Generic errors
- External
  - The Software Provider
    - Software specific error or bug
      - Github documentation
  - The Resource Provider
    - Resource specific question (file systems, nodes)
    - Performance
    - At the end of your rope ☺

Help Desk: Useful Information

- Clear Description of the problem
- Resource
- UserID/Username, Account/Project/Allocation
- Relevant issue information
  - JobId
  - List of Node(s) that job ran on
  - Working directory (submit script name)
  - Location of .err and .out files
  - Start time
  - End time
  - Resources requested
  - Application Used

Schedulers

- Scheduler are used for Cluster management and job scheduling
- Multiple schedulers available(Maui, torque, PBS, SLURM)
  - Running/monitoring/controlling jobs
  - Syntax varies, but the concepts the same
  - Slurm is widely used on ACCESS systems
- Schedulers can be configured to manage resources
  - System set Job limits
  - User Set Job limits
    - Number of jobs
    - Duration
    - Per User or Per Project
  - Set priority
  - Allocate resource
  - Allocate Licenses
  - Job dependency
  - Resource dependency

Batch scripts

- Job Scripts are used to request resources
  - Allocations
  - Compute resources (cpu, gpu)
    - Queues/Partitions/QOS
  - Time Limits
  - Memory
  - Set up environment and execute applications
  - Reservations
  - Licenses
  - Input files
  - Executables
  - Modules*
  - Location of output and error files
- Slurm tools used to monitor and manage resources
  - Squeue, scontol, sacct, sinfo

Case Scenario: Job Not Running

- Identify the Issue
  - Title Subject: Job not running
- Identify the Nature of the issue
  - Error message
    - Command line responses
    - .out and .err files
  - Use scheduler tools
    - squeue command: (ReqNodeNotAvail, UnavailableNodes:exp-4-50,exp-5-38,….)

```
login01 ~]$ squeue | grep PD
  28599461   compute pulsar_5  xqalaxy PD       0:00      1 (Resources)
  28598817   compute    Mn_BS  xqalaxy PD       0:00      2 (ReqNodeNotAvail, UnavailableNodes:exp-2-49,exp-9-[23,30],exp-12-12,exp-14-[30,50,53],exp-16-[5
```

- Identify the cause of the issue
  - Looking at Batch Script
  - Use Slurm commands
    - Scontrol
    - Sacct

squeue: Job Status

- View information about queued or running jobs
  - squeue [OPTIONS...] (PBS,MAUI: qstat,showq)
    - Queued Jobs

```
   JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
13574113   compute 80dgree_   yweng3 PD       0:00      2 (MaxMemPerLimit)
12668967   compute 0-xtensi  kavousi PD       0:00      1 (MaxMemPerLimit)
14756880   compute job001_p  amytsai PD       0:00     10 (Reservation)
14800161   compute namd-com sasadian PD       0:00      6 (QOSMaxCpuPerUserLimit)
14800218   compute namd-com sasadian PD       0:00      6 (QOSMaxCpuPerUserLimit)
14789098   compute bl_8JHNp    uscms PD       0:00      1 (MaxJobsPerAccount)
```

    - Running jobs

```
14813206   compute post0110 lpegolot  R   16:30:28      1 exp-9-35
14800090   compute namd-com sasadian  R   16:13:01      6 exp-2-29,exp-3-23,exp-4-33,exp-7-20,exp-9-[03,26]
14764467   compute V1WTReRU  aminkvh  R   16:08:56      1 exp-2-54
14773832   compute V4R1639Q  aminkvh  R   15:55:58      1 exp-8-14
14800092   compute namd-com sasadian  R   15:29:28      6 exp-4-29,exp-7-[07,39-40],exp-9-[28,41]
14812166   compute  scratch mlaskow2  R   15:53:59      1 exp-10-20
14812167   compute  scratch mlaskow2  R   15:39:34      1 exp-8-48
14800158   compute namd-com sasadian  R   15:17:18      6 exp-2-[26,50],exp-4-[52-53],exp-7-[42-43]
14812168   compute  scratch mlaskow2  R   15:20:01      1 exp-10-37
```

squeue: Job Status

- Queue states
  - PD –PENDING, R – RUNNING, S – SUSPENDED, CA – Canceled, CG – COMPLETING, CD – COMPLETED
- squeue – Common “reasons” for pending jobs
  - MaxMemPerLimit
  - QOSMaxNodePerUserLimit
  - Priority
  - ReqNodeNotAvail, Unavailable nodes: exp-x-xx
  - ReqNodeNotAvail, Reserved for maintenance
  - Licenses
  - Dependency (file system, license, another job)
  - Priority
- System Maintenance
  - https://support.access-ci.org/outages

sacct & scontrol

- scontrol [OPTIONS...] [COMMAND...]
  - View or modify Slurm configuration for queued or running jobs

```
[nickel@login01 ~]$ scontrol show job 28570649
JobId=28570649 JobName=V1213_CTG_N36
   UserId=              (535314) GroupId=cit170(10115) MCS_label=N/A
   Priority=0 Nice=0 Account=cit170 QOS=normal
   JobState=PENDING Reason=BadConstraints Dependency=(null)
   Requeue=0 Restarts=0 BatchFlag=1 Reboot=0 ExitCode=0:0
   RunTime=00:00:00 TimeLimit=12:00:00 TimeMin=N/A
   SubmitTime=2024-02-08T16:47:24 EligibleTime=2024-02-08T16:47:24
   AccrueTime=2024-02-08T16:47:24
   StartTime=Unknown EndTime=Unknown Deadline=N/A
   SuspendTime=None SecsPreSuspend=0 LastSchedEval=2024-02-08T19:03:46 Scheduler=Main
   Partition=compute AllocNode:Sid=login02:2522184
   ReqNodeList=exp-15-[04-20] ExcNodeList=exp-4-04,exp-7-31,exp-9-[23,30],exp-14-[30,50],exp-16-[53-54]
   NodeList=
   NumNodes=2-2 NumCPUs=256 NumTasks=64 CPUs/Task=4 ReqB:S:C:T=0:0:*:*
   ReqTRES=cpu=256,mem=498416M,node=2,billing=921600
   AllocTRES=(null)
   Socks/Node=* NtasksPerN:B:S:C=32:0:*:* CoreSpec=*
   MinCPUsNode=128 MinMemoryNode=249208M MinTmpDiskNode=0
   Features=(null) DelayBoot=00:00:00
   OverSubscribe=NO Contiguous=0 Licenses=(null) Network=(null)
   Command=/expanse/lustre/scratch/dgarzonarmendariz/temp_project/simulations/V1213_CTG_N36/output-0039/SIMFACTORY/SubmitScript
   WorkDir=/home/dgarzonarmendariz/GW150914
   StdErr=/expanse/lustre/scratch/dgarzonarmendariz/temp_project/simulations/V1213_CTG_N36/output-0039/V1213_CTG_N36.err
   StdIn=/dev/null
   StdOut=/expanse/lustre/scratch/dgarzonarmendariz/temp_project/simulations/V1213_CTG_N36/output-0039/V1213_CTG_N36.out
   Power=
   MailUser=rhaas MailType=INVALID_DEPEND,BEGIN,END,FAIL,REQUEUE,STAGE_OUT
```

- sacct [OPTIONS...]
  - View accounting data for completed jobs and job steps

```
[nickel@login01 ~]$ sacct -j 28582088 -l
JobID        JobIDRaw        JobName  Partition  MaxVMSize  MaxVMSizeNode  MaxVMSizeTask  AveVMSize     MaxRSS MaxRSSNode MaxRSSTask
    AveRSS MaxPages MaxPagesNode   MaxPagesTask   AvePages     MinCPU MinCPUNode MinCPUTask     AveCPU   NTasks  AllocCPUS    Elapsed
      State ExitCode AveCPUFreq ReqCPUFreqMin ReqCPUFreqMax ReqCPUFreqGov     ReqMem ConsumedEnergy  MaxDiskRead MaxDiskReadNode MaxD
iskReadTask    AveDiskRead MaxDiskWrite MaxDiskWriteNode MaxDiskWriteTask   AveDiskWrite    ReqTRES  AllocTRES TRESUsageInAve TRESUsa
geInMax TRESUsageInMaxNode TRESUsageInMaxTask TRESUsageInMin TRESUsageInMinNode TRESUsageInMinTask TRESUsageInTot TRESUsageOutMax TRE
SUsageOutMaxNode TRESUsageOutMaxTask TRESUsageOutAve TRESUsageOutTot
```

Why is my job not running? Queue, wait times, Time Limits

- Sacctmrg – used to view and modify slurm account information
- Sinfo –used to view information about nodes and partitions
- User Guide

Partition Name	Max Walltime	Max Nodes/Job	Max Running Jobs	Max Running + Queued Jobs	Charge Factor	Notes
compute	48 hrs	32	32	64	1	Exclusive access to regular compute nodes; *limit applies per group*
ind-compute	48 hrs	32	32	64	1	Exclusive access to Industry compute nodes; *limit applies per group*
shared	48 hrs	1	4096	4096	1	Single-node jobs using fewer than 128 cores
ind-shared	48 hrs	1	32	64	1	Single-node Industry jobs using fewer than 128 cores

```
[nickel@login01 ~]$ sinfo
PARTITION      AVAIL  TIMELIMIT  NODES  STATE NODELIST
compute           up 2-00:00:00      3 drain$ exp-14-30,exp-16-[53-54]
compute           up 2-00:00:00      1  maint exp-9-23
compute           up 2-00:00:00      2  down* exp-12-12,exp-14-53
compute           up 2-00:00:00      3   comp exp-8-29,exp-9-25,exp-13-19
compute           up 2-00:00:00      1   drng exp-8-26
compute           up 2-00:00:00      2  drain exp-9-30,exp-14-50
compute           up 2-00:00:00      8   resv exp-13-[55-56],exp-16-[55-56],exp-17-[53-
compute           up 2-00:00:00    185    mix exp-1-[22,25-26,48-49,51-55],exp-2-[01-02
30,39-40,42,44,46-47],exp-13-[01-07,12,16,20,22,27,42,46,48,50,52,54],exp-14-[01-02,04,
compute           up 2-00:00:00    375  alloc exp-1-[01-21,23-24,27-47,50,56],exp-2-[03
-[21-28,34,38-54],exp-12-[01-11,13-16,29,31-38,41,43,45,48-56],exp-13-[08-11,13-15,17-1
compute           up 2-00:00:00    154   idle exp-4-[15-56],exp-5-[01-56],exp-6-[01-56]
```

```
[nickel@login01 ~]$ sacctmgr show qos format=name%20,MaxWall,MaxTRESPU%20,MaxJobsPU,MaxSubmitPU,MaxTRESPA%20,MaxJobsPA,MaxSubmitPA
                Name     MaxWall              MaxTRESPU MaxJobsPU MaxSubmitPU              MaxTRESPA MaxJobsPA MaxSubmitPA
-------------------- ----------- -------------------- --------- ----------- -------------------- --------- -----------
              normal  2-00:00:00     cpu=8192,node=64        32          64   cpu=16384,node=128        32          64
       shared-normal  2-00:00:00     cpu=8192,node=64      4096        4096   cpu=16384,node=128      4096        4096
 large-shared-normal  2-00:00:00                              2           4                            4           4
      preempt-normal  7-00:00:00                                        128     cpu=4096,node=32                 128
         gpu-normal  2-00:00:00 cpu=160,gres/gpu=16+           4           8   gres/gpu=32,node=8        8          16
  gpu-shared-normal  2-00:00:00 cpu=240,gres/gpu=24+          24          24 cpu=320,gres/gpu=32+       24          24
 gpu-preempt-normal  7-00:00:00   gres/gpu=24,node=6          12          16  gres/gpu=48,node=12       16          20
```

Case Scenario: My job ran for only 10 minutes on 1 node why did I get charged so much?

- All systems charge differently
- ACCESS allocates in ACCESS Credits which can be converted to SUs (service Unit)
  - Each resource has a unique definition of an SU
  - Some are in core hours, some are in node hours, etc
  - Some charge for other components such as memory
- Visit the user guide for specifics
  - Check for opportunities to save
- Allocations are shared
- Charging is generally based on what is requested, not how resources are used
- Do test jobs to evaluate
  - Slurm commands to collect information
    - Sacct –u $USER
    - Sacct –j $JOBID

Accounting

- Most HPC system use a scheduling tool (ACCESS slurm)
- Different resources use different home grown tools to evaluate their usage.

```
[nickel@login01 ~]$ expanse-client user nickel -r expanse_gpu -p

 Resource  expanse_gpu

 NAME    STATE  PROJECT  TG PROJECT    USED  AVAILABLE  USED BY PROJECT
-------------------------------------------------------------------------
 nickel  allow  ddp324                   0       5000               21
 nickel  allow  ddp386                   0       2500               74
 nickel  allow  sds154   TG-ASC150024    0        100              517
 nickel  allow  sds166   TG-STA160003    0       2500                3
 nickel  allow  use300                   9     269000            63638
[nickel@login01 ~]$ ^C
[nickel@login01 ~]$ expanse-client user nickel -p

 Resource  expanse

 NAME    STATE  PROJECT  TG PROJECT    USED  AVAILABLE  USED BY PROJECT
-------------------------------------------------------------------------
 nickel  allow  ddp386                   2     110000             9163
 nickel  allow  sds154   TG-ASC150024    0       1000            16572
 nickel  allow  sds166   TG-STA160003    0     100000            56528
 nickel  allow  use300                5856    5050000          3457273
```

- Expanse-client tool(SDSC)
- Projects (PSC)
- taccinfo (TACC)
- ACCESS Portal updated at various intervals

```
[userid@login018 ~]$ projects
Your default charging project charge id  is abcd1234.  If you would like
the default charging project  use the command change_primary_group ~charg
Use the charge id listed below for the project you would like to make the
in place of ~charge_id~

Project: ABCD1234
     PI: Cy Entist
  Title: World Renowned Research

      Resource: Bridges 2 GPU
    Allocation: 10000.00
       Balance: 8872.00
      End Date: 2021-07-15
  Award Active: Yes
   User Active: Yes
     Charge ID: abcd1234
   Directories:
       HOME /jet/home/userid
      Resource: Bridges 2 Regular Memory
    Allocation: 23000.00
       Balance: 197450.00
      End Date: 2021-07-15
  Award Active: Yes
```

```
login1$ /usr/local/etc/taccinfo        # Generally more current than balances displayed on the portals.
```

Best Practices: jobs and job charging

- Check user guide for accounting policies
- Use system tools for most up to date accounting information
  - Slurm for individual job details
    - Sacctmgr
    - Sacct
    - Squeue
    - scontrol
    - Sinfo
  - Home grown tools for accounting information

Case Scenario:
Do you have application X v1.x?

- Software is made available
  - Modules
  - Compile
  - Containers
    - Ask help desk
- Most HPC system use module to manage their software stacks
  - Check for available software with the module commands
    - module [options] sub-command [args ...]
- Login nodes, compute nodes are different
  - Compiling codes need to happen on the nodes on which they will run
  - Modules need to be loaded on the nodes on which applications will run
- Software managed by the resource provider has usually been tested and fine tuned for the specific resources

Environment Modules

- Environment Modules help manage software incompatibilities, versioning, and dependencies
- Environment Modules provide for dynamic modification of your shell environment
- Module commands set, change, and/or delete environment variables
- Modules manage software versions
- Module manage dependencies by loading or unloading other modules
  - Check for dependencies with module spider <module_name>
- Module list - lists all the currently loaded modules

Navigating with Modules

- Usage: module [options] sub-command [args ...]
  - module list –currently loaded modules
  - module avail – list of available software modules based on your current module path
  - module spider –lists all available software modules and the versions on the system
  - module load [modulefile] –loads modules or specifies that there are unresolved dependencies
  - module show [modulefile] displays information about loaded modules including changes, dependencies , versions and path
  - module unload [modulefile] Unloads a specified module from the environment
  - Module purge – unloads all loaded modules
  - Module reset – will resent modules to default settings.

```
[nickel@login01 ~]$ module avail

--------------------------- /cm/shared/apps/spack/0.17.3/cpu/b/share/spack/lmod
   anaconda3/2021.05/q4munrg             gh/2.0.0/mkz3ux1             mercurial/
   aocc/3.2.0/io3s466                    git-lfs/2.11.0/kmruniy       parallel/2
   aria2/1.35.0/q32jtg2                  git/2.31.1/ldetm5y           pigz/2.6/b
   entrezdirect/10.7.20190114/6pkkpx2    intel/19.1.3.304/6pv46so     rclone/1.5
   gcc/10.2.0/npcyll4                    matlab/2022b/lefe4oq         sratoolkit

-------------------------------------------------------- /cm/local/modulefiles -
   cmjob               docker/20.10.21    openmpi/mlnx/gcc/64/4.1.5a1        s
   cuda-dcgm/3.1.3.1   lua/5.4.4          shared                      (L)    s

------------------------------------------------- /cm/shared/apps/access/modulef
   accessusage/0.5-1    cue-login-env

----------------------------------------------------- /usr/share/modulefiles -
   DefaultModules (L)    cpu/0.17.3b (c,L,D)    gpu/0.17.3b    (g,D)    nostack
   cpu/0.15.4     (c)    gpu/0.15.4  (g)        nostack/0.15.4 (e)

------------------------------------------------------- /cm/shared/modulefiles -
   AMDuProf/3.4.475    cm-pmix3/3.1.7    default-environment    sdsc/1.0 (L)

  Where:
```

```
[nickel@login01 ~]$ module spider amber

----------------------------------------------------
  amber:
----------------------------------------------------
     Versions:
        amber/18.18
        amber/20-patch15
        amber/20
        amber/20.21
     Other possible modules matches:
        amber/22
```

Best Practices: Software

- Review user guide for tools available
- Use system installed applications when available
- Use containers to manage out of date software

Best Practices: Getting help

- Understand your problem
- Engage with appropriate support tools
- Help Desk
  - While help desk staff are exceptional, they should be considered general practitioners
  - Provide relevant and adequate information for helpdesk to reduce iterations
    - Username, Account, System, Jobid, specific error message if available, etc.
    - The user with the problem should submit the help ticket
- Always be nice to the support desk! ☺

Table of contents

- Getting Help
  - Available Resources
  - Identify the most appropriate resource
  - Use Case: Job wont start(schedulers)
  - Use Case: Job charging (sacct, scontrol, sinfo, smanager)
  - User Case: Software(modules)
- **Securing your environment**
  - Re-enforce the importance for Data security
  - What do we need to protect
  - When do we need to protect ourselves
  - Who do we need to protect ourselves from
  - Use Case: Sharing files (passwords)
  - Use Case: I can't access the resource(file structure, chmod, chown)
  - Use Case: Missing Files(backups)
- Summary: Quick and easy tips/reminders for getting help and reducing risk

Security is a Shared Responsibility

- Resource provider is responsible to:
  - Provide Authorized Users Access to Computational Resources
  - Protect user accounts/data from unauthorized users
  - Enforce the user set permissions on data
- End user will: (review ACCESS AUP)
  - Protect personal account credentials
  - Protect personal data with appropriate permission controls
  - Use resources only for the purpose for which it has been authorized to use

Security: Agenda

- Revisit your security practices to manage risk
- Data Breach
  - Any incident in which confidential or sensitive information has been accessed without permission, including unauthorized access to a computer system or network. The offending party then steals the private, sensitive, or confidential personal and financial data of the customers or users.

- Data Security
  - The safeguarding digital information throughout its life cycle to protect it from loss, corruption, theft, or unauthorized access”. Including hardware, software, storage devices, and user devices.

Why do we need data security?

- Protect our data and resources from unauthorized access
  - To Avoid
    - Data corruption
    - Loss of data
    - Loss of access to resources, data
  - To maintain
    - Direct Access to resources, data
    - Indirect access to resources/data
      - A compromised personal computer can compromise external resources
      - An attacker on your computer can do anything you can

When do we need to protect ourselves

- Always
  - Even if you don't….
    - have anything interesting
    - have sensitive data, your research is public
  - But……..
    - Attackers are opportunistic
    - Attackers are not aware of what you have
    - Attackers are interested in information you are not aware they are interested in
    - "Attack" may not be deliberate
      - Causes of Data loss include: Theft, Computer virus, Software corruption, Hardware failure, Natural disaster, Power failure, Human Error

Who do we need to protect from

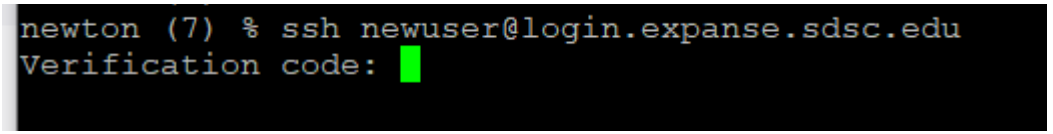
- Nefarious Character (deliberate, intentional)
- Friendly Character (inadvertent, unintentional)
  - Deleting personal files
    - > rm –rf *
    - > rm –rf / directory/file (notice the space after the '/')
- System issues

What do we need to protect!

- Client, Resource
  - Personal Devices, credentials
  - Remote Devices
- Data
  - Files, directories (Data corruption/modification/deletion)
- Code/Project
  - Research Integrity
  - Project dependencies
  - Project repository

Case scenario: I cant log in!

- Title: User is unable to log into a system
- Description to include
  - Username
    - Check username on system (may be different)
  - Resource Name: Where are you logging in
  - When did you last access the system, have any of your credentials changed
- Checks
  - Have you checked the system availability page
  - Have you checked the user guide
  - Check specific error message
  - Check User News
    - Stay Subscribed to be notified
    - Check on User Portal -- https://support.access-ci.org/announcements



Resource Access

- Different resources have different access protocols, or mechanisms for access, even if its at the same site
- Password and username issues
  - Indicator message: Enter verification code
    - https://allocations.access-ci.org/profile (if username is not available for the resource then the account has not been created yet) Generally it can take 1 business day for accounts to be fully functional.
  - ssh keys
    - Indicator message: Enter password

Passwords/passphrases

- Passwords
  - Don’t reuse passwords
  - Longer is better
  - Don’t keep digital plaintext copies of passwords
  - Don’t Hard code password in files
  - Don’t share passwords

ssh keys

- Ssh –secure socket shell
- SSH keys: key pair of public and private keys that are used to authenticate and establish an encrypted communication channel between a client and a remote machine over the internet.
  - Generate an SSH key pair on local host (RSA, DSA, ECDSA, Ed25519).
    - ssh-keygen (linux)
    - Puttygen (windows)
  - Copy Public Key to remote host
    - Authorized_keys file
  - Best practices for Key Mangement
    - regular generation, re-keying, and rotation of SSH keys
    - Do not reuse Passcodes
    - Avoid hard-coding keys

Agent Forwarding

- Manage ssh connections using ssh-agent forwarding
- ssh agent forwarding allows users to use local SSH keys for remote authentication
- Keys will be removed when systems is rebooted
- Keys should be removed from an agent when not in use
- Steps to set up agent forwarding
  - Activate ssh-agent(linux), pagent(window)
  - Add key to ssh-agent
    - Verify that the ssh-agent is running and if you have any keys already added
      - ssh-add -l
    - Add the private key to the ssh-agent
      - ssh-add sdsc_id_rsa
    - Save public key to remote host
      - Ssh-copy-id username@login.remote.host

  (https://github.com/sdsc-hpc-training-org/hpc-security/blob/master/ssh_methods/connect-using-ssh-agent.md)

Best Practices: Secure your credentials

- Passwords
  - Don’t reuse passwords
  - Longer is better
  - Don’t keep digital plaintext copies of passwords
  - Don’t share passwords
- Use password-manager program
- Use SSH keys, ssh-agent
- Multi-factor Authentication(MFA)

Case Scenario: My files have vanished!

- A user logs in to find all of their work has vanished. What happened?
  - It started with a request. I need to make a directory publicly available to other people in my group on expanse. I was told chmod would do the trick

Can I have access the the input data?

Sure, let me change the file permissions

  - File systems
    - Where was the data stored
    - When did you last access data
    - Have backups
  - Who had access
    - Don’t share credentials
    - What about sharing file permissions
    - Back door attack

File management

- Review file permissions ls –l
- Review user group
  - Groups username, id username
- Controls: Permissions granularity levels
  - (Attribute), User, Group, Other
  - Read(4), Write(2), Execute(1)
  - Default 755 (User(read, write, execute):Group(read, execute): Other(read, execute)
  - Use chmod, chown commands to modify ownership and permissions

What is in a file system

- Common File systems and their utilization
  - Home – Usually limited in space
  - Scratch - Large space, limited persistence, no backup
  - Node local scratch(SSD)- good performance, only available during job
  - Archival- Slow, backup
- Types of file systems
  - Lustre
  - Nsf
  - Gpfs
  - ceph

Have a backup plan

- Copy critical data of system regularly
- Version Control: Git
- Checksum data transfers to ensure no corruption
- ACCESS to HPC system and file retention is usually limited. Transferring data takes time
- Data on the system is the users responsibility
- Plan ahead for data transfers
- Convert many small files into a single archive file before transfer

Best Practices: Backups

- Backups should be done at regular intervals that make sense to your project
  - Frequency
  - Ensure backups are made on “good” versions
  - Perhaps retain a few versions just in case
- Don’t back up everything
  - Clean up unnecessary files
  - Backup files not easily reproduced or replaced
    - source code, scripts, config files and large output files
- Backups should be on a different resource
- Ensure credentials would not allow hackers to get onto the external resource
- Test the backup plan with the restore process

Best Practices: Risk Management

- Manage Credentials
- Manage directory and file Access
  - Use least privileges
  - Use chmod, chown commands to modify ownership and permissions
- Data Resiliency
  - Clean up unnecessary files
  - Back up Data
  - Use integrity checking
    - Data transfers, bad hardware
- Have a contingency plan(Data recovery plan)
  - Off site backup

Best Practices: Client security

- Protect your resources
- Install and run anti-malware software
- Keep personal machine and software updated

Best Practices: Project security

- Reduce dependencies with in projects
  - Large software project depend on third party libraries and modules
  - Therefore the project is relying on the best practices of others to maintain the security and integrity of the project
- Protect web based application on shared compute resources (Jupyter Notebooks, Globus connect personal)
- Backup our data!

In Conclusion

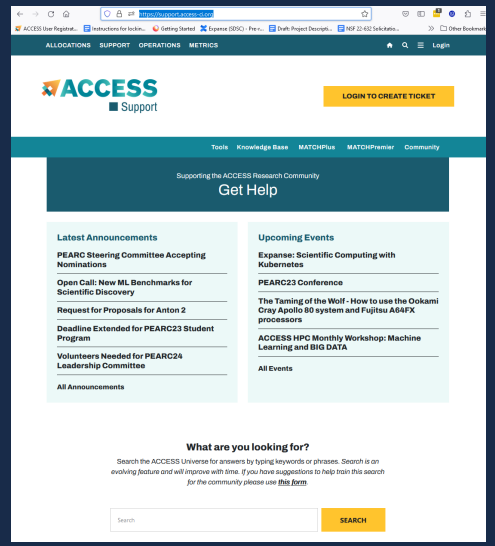
- Manage your risks by securing your accounts, research and data
- Be a good citizen
- There is help available if you know where to look

Review and helpful links

- ACCESS AUP
  - https://identity.access-ci.org/aup.html
- SSH Key setup
  - https://github.com/sdsc-hpc-training-org/hpc-security
- Comet Webinar- Indispensable Security: Tips to Use SDSC's HPC Resources Securely
  - *https://www.sdsc.edu/event_items/202007_CometWebinar.html*
- Expanse Webinar: Enduring Security: The Journey Continues
  - *https://education.sdsc.edu/training/interactive/202204_expanse_enduring_security/index.html*
- Training Catalog
  - *https://www.sdsc.edu/education_and_training/training_hpc.html#catalog*

Questions

- consult@sdsc.edu
- https://support.access-ci.org/




NSF