Introduction to Minerva

Minerva Scientific Computing Environment

https://labs.icahn.mssm.edu/minervalab

Patricia Kovatch
Lili Gai, PhD

Hyung Min Cho, PhD
Eugene Fluder, PhD
Jielin Yu, PhD
Wei Guo, PhD
Kali McLennan
Yiyuan Liu, PhD

September 27 2023

Icahn
School of
Medicine at
Mount
Sinai

Outlines

- Compute and storage resources
- Account and logging in
- User software environment
- Other services on file transfer, data archive, and web server

Minerva cluster @ Mount Sinai

Chimera Partition:

- 3x **login nodes** - Intel 8168 24C, 2.7GHz - **384 GB** memory
- Compute nodes -
  - 275 **regular memory nodes** - Intel 8168 24C, 2.7GHz - 48 cores per node - **192 GB**/node)
  - 37 **high memory nodes** - Intel 8168/8268, 2.7/2.9GHz - **1.5 TB** mem
  - **GPU nodes:**
    - 12 -Intel 6142, 2.6GHz - 384 GB memory - 4x V100-**16 GB** GPU
    - 8 - Intel 8268, 2.9 GHz - 384 GB memory - 4x A100- **40 GB** GPU
      - 1.8 TB SSD per node
    - 2 - Intel 8358,2.6GHz - 2 TB memory - 4x A100- **80 GB** GPU
      - 7 TB SSD per node, NVlink

NIH FUNDED NODES

BODE2 Partition:

- $2M S10 BODE2 awarded by NIH (Kovatch PI)
- 78 compute nodes - Intel 8268, 2.9 GHz -48 cores per node - **192 GB**/node

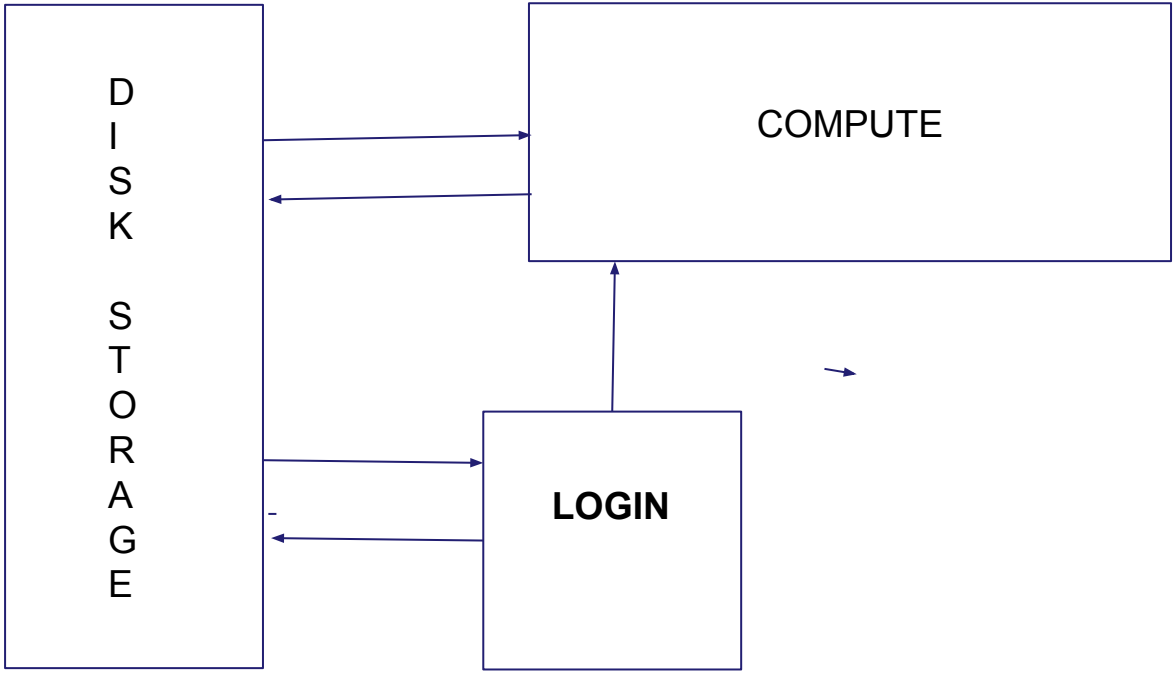
CATS Partition:

- $2M CATS awarded by NIH (Kovatch PI)
- 55 compute nodes - Intel 8358, 2.6 GHz- 64 cores per node -**1.5 TB** / node

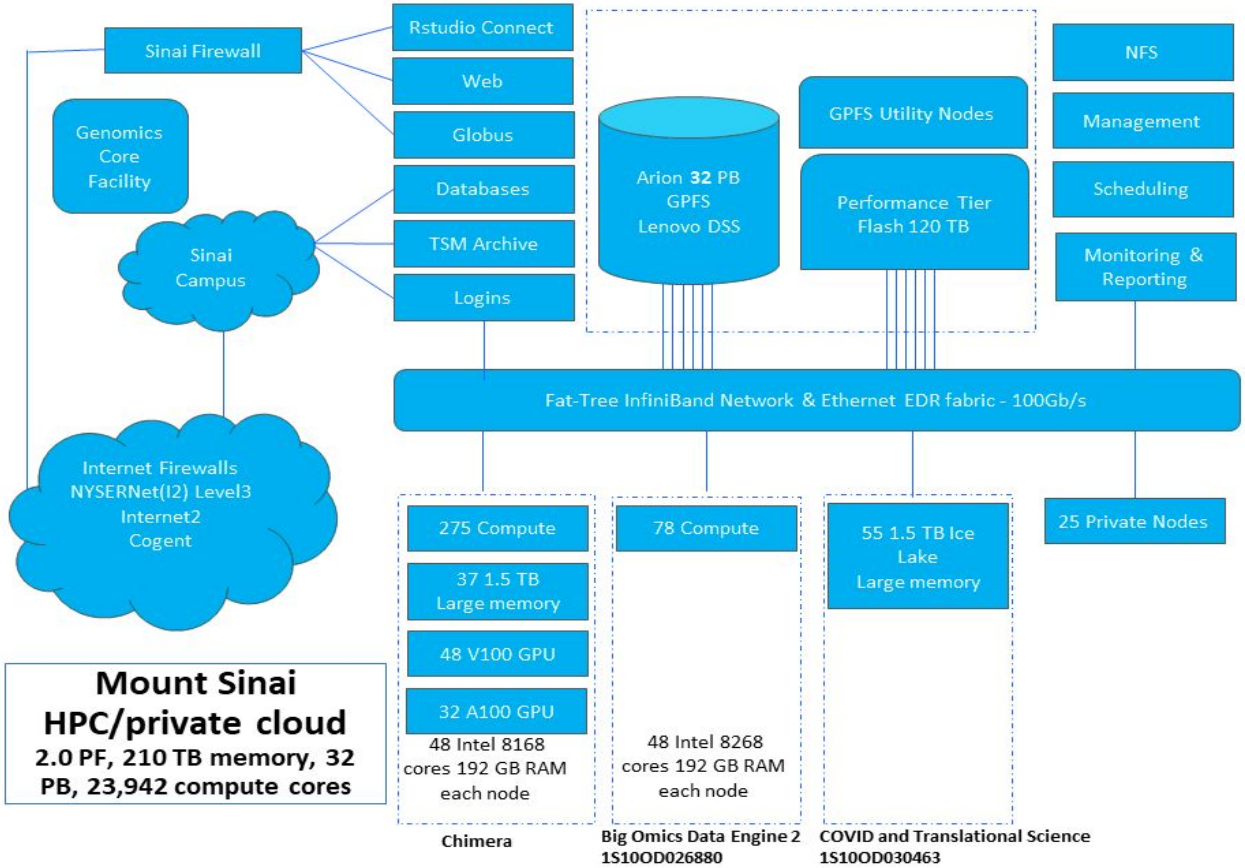
Storage: 32PB of high-speed online storage as an IBM General Parallel File System (**GPFS**)

- **Path /sc/arion** : Use the system path environment variable in scripts **$GPFS**

Minerva Cluster from 30,000 ft.



Minerva Cluster@ Mount Sinai



HIPAA

- Minerva is HIPAA compliant as of October 1st, 2020, i.e., Protected Health Information (PHI) data is allowed to be stored and processed on Minerva.

- All users have to read the HIPAA policy and complete Minerva HIPAA Agreement Form **annually** (every Jan.) at https://labs.icahn.mssm.edu/minervalab/hipaa/

- Users who have not signed the agreement will have their accounts locked until the agreement is signed.

General Minerva Information

- The Minerva website is:

  https://labs.icahn.mssm.edu/minervalab/

  or

  https:www.hpc.mssm.edu

  - Contains the documentation for the features and access to the forms
- Access to Minerva requires a Minerva userid
  - Not automatic. You need to apply for one.
  - Username will be the same as your Mount Sinai Login id.
  - Link to form off of Scientific Computing home page or https://labs.icahn.mssm.edu/minervalab/request-an-account/

Logging in

Minerva is a Linux machine with Centos 7.6

- Linux is command line based, not GUI (But we have a gui wrapper: OnDemand)
- Logging in requires campus network, **SSH client** installed on your machine, **username, memorized password,** and **one-time code** obtained from a Symantec VIP token

Detailed procedures:

- Campus network (School VPN needed if off-campus)
- Apply for an account at https://acctreq.hpc.mssm.edu/
  - Apply account for external users following here
- Complete HIPAA form at https://labs.icahn.mssm.edu/minervalab/hipaa/ to activate your account
- Register your token at the Self Service Portal **school site** (https://register4vip.**mssm**.edu/vipssp/)
- SSH client: terminal (Mac), MobaXterm/Putty (Windows)
- Logging info at https://labs.icahn.mssm.edu/minervalab/logging-in/

*Note: Minerva is school resource, so use your **school password and school portal** for register*

Migration from Symantec VIP to Azure MFA

- DTP is in the process of migrating to Azure MFA
- Planned cutover from Symantec to Azure for Minerva is late 2024 (when VPN is fully with Azure MFA)
  - All users of Minerva need to register for MFA in preparation https://itsecurity.mssm.edu/mobile-device-security/ms-authenticator/
- In the meantime, for **new users**, to use VIP token to access Minerva
  - Need to first get access to VPN or school email; then register VIP token
  - If no VPN or school email account, then you need to request a role via Sailpoint "**Azure MFA and Symantec VIP Token Registration"**

Logging in - Linux / Mac

Connect to Minerva via ssh

- Open a terminal window on your workstation
- ssh *your_userID*@minerva.hpc.mssm.edu
- To display graphics remotely on your screen, pass the “-X” or “-Y” flag:
  - ssh -X *your_userID*@minerva.hpc.mssm.edu
  - Mac: Install XQuartz on your mac first
  - Test by running the command: xclock
    - Should see a clock
- Landed on one of the login nodes, and at your home directory
  - Never run jobs on login nodes
  - For file management, coding, compilation, check/manage jobs etc., purposes only
  - Basic linux command**: cd, ls and more**

```
imac:~ gail01$ ssh -X gail01@minerva.hpc.mssm.edu
Please input your password and two factor token:
Password:
Last login: Mon Sep 13 16:24:06 2021 from 10.254.167.11
================================================================
Run "Minerva_help" for useful Minerva commands and websites

   === Upcoming Minerva Training Sessions ===
Session 1: 15 Sep 2021, 11:00AM-12:00PM – Introduction to Minerva
Session 2: 22 Sep 2021, 11:00AM-12:00PM – LSF Job Scheduler
Session 3: 29 Sep 2021, 11:00AM-12:00PM – Globus: Data Transfer
Zoom link for all sessions:
https://mssm.zoom.us/j/5420563013


   === Send ticket to hpchelp@hpc.mssm.edu ===
WE DO NOT BACKUP USER FILES
        PLEASE ARCHIVE/BACKUP YOUR IMPORTANT FILES
   === Send ticket to hpchelp@hpc.mssm.edu ===
================================================
gail01@li03c04: ~ $ pwd
/hpc/users/gail01
gail01@li03c04: ~ $ xclock
```

```
gail01@li03c02: ~ $ minerva_help

Our Minerva HPC website:          https://labs.icahn.mssm.edu/minervalab/
Our lastest Slides:
Basic Minerva Enviorment https://labs.icahn.mssm.edu/minervalab/wp-content/uploads/sites/342/2020/10/Minerva_Intro_-2020-09-16.pdf
Advanced LSF job scheduler https://labs.icahn.mssm.edu/minervalab/wp-content/uploads/sites/342/2020/09/Minerva_LSF_2020-09-23.pdf

Publish R/shiny to Rstudio Connect server: https://labs.icahn.mssm.edu/minervalab/rstudio-connect-server/
Data transfer:                    https://labs.icahn.mssm.edu/minervalab/data-transfer/
Web service:                      https://labs.icahn.mssm.edu/minervalab/web-services/
TSM data archive:                 https://labs.icahn.mssm.edu/minervalab/archiving-data/

Useful Commands

Check limits on login nodes:      ulimit -a
Modules:
Show all available modules:       ml avail
Load module:                      ml <package name>
List loaded modules:              ml list
Unload a module:                  ml -<package name>
Remove all loaded modules:        ml purge

Run rstudio over GUI:             ml rstudio; rstudio
Run rstudio over web:             minerva-rstudio-web.sh (with details at https://labs.icahn.mssm.edu/minervalab/rstudio-web/)

Check available accounts for computes: mybalance
Submit job to LSF job scheduler: bsub < my_lsf_script.sh
see a list of your jobs:          bjobs
check currentjob in long format: bjobs -l jobID
List all compute nodes:           bhosts
List high memory nodes:           bhosts himem
List gpu nodes:                   bhosts gpu
List interactive nodes:           bhosts interactive
List all available LSF queues:    bqueues
```

Logging in - Windows

- **Install MobaXterm from** https://mobaxterm.mobatek.net/
  - Enhanced terminal for Windows with **X11 server**, tabbed SSH client, network tools and much more

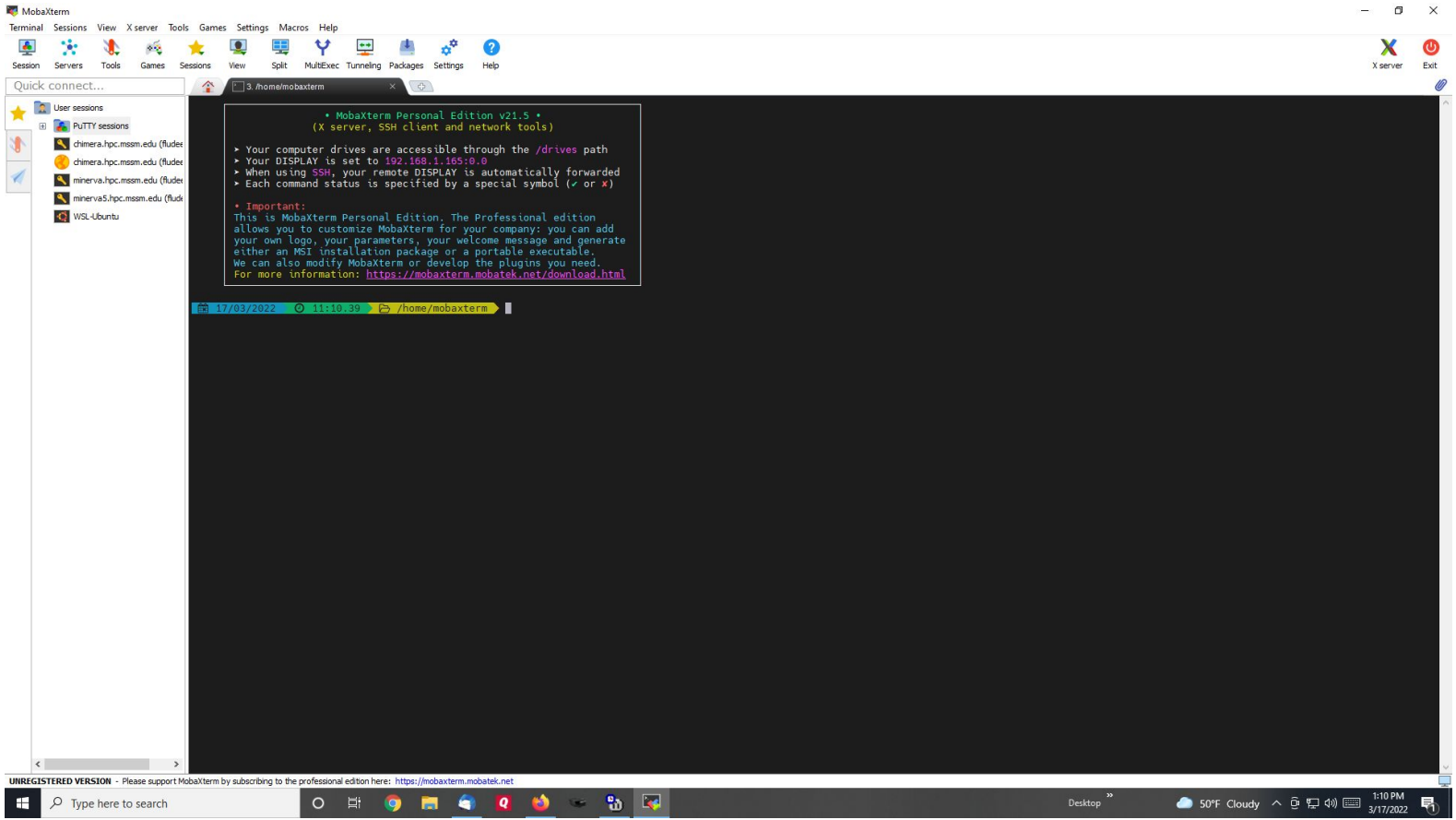
OR

- **Install PuTTY from www.putty.org**
  - Google it. It will be the first hit https://www.youtube.com/watch?v=ma6Ln30iP08
  - If you are going to be using GUI's, in Putty: Connection > SSH > X11
    - Ensure "Enable X11 forwarding" is selected
  - On Windows box install Xming
    - Google; Download; Follow bouncing ball
  - Test by logging into Minerva and run the command: xclock
    - Should see a clock

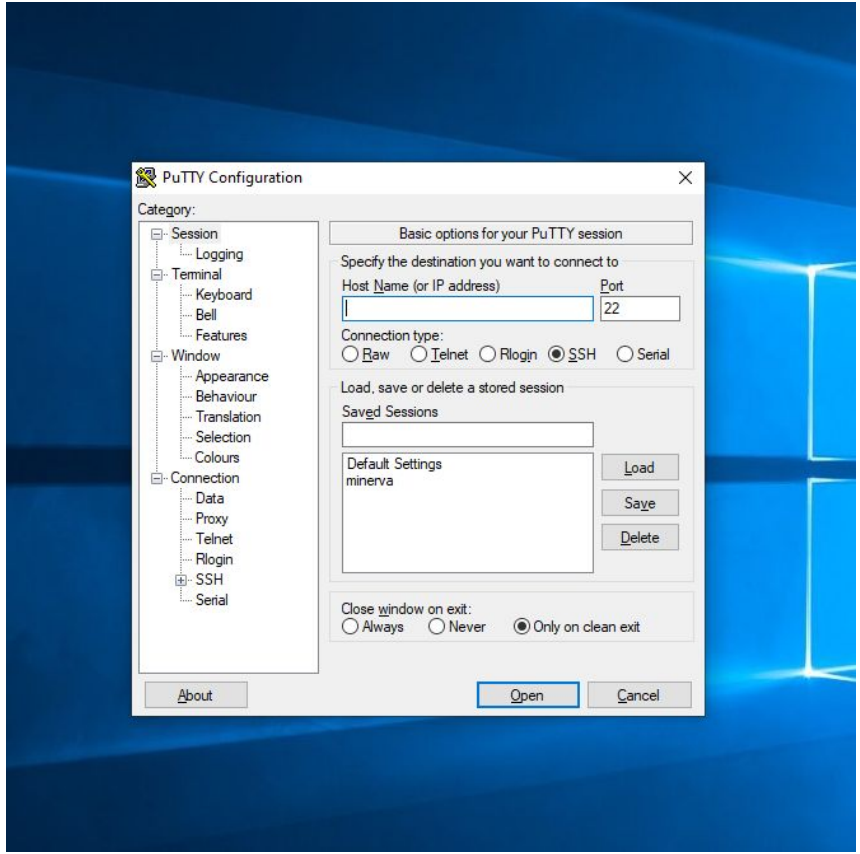
OR

- **Install Windows Subsystem for Linux (WSL) here**
  - Run a Linux environment - including most command-line tools, utilities, and applications -- directly on Windows, unmodified, without the overhead of a traditional virtual machine or dualboot setup

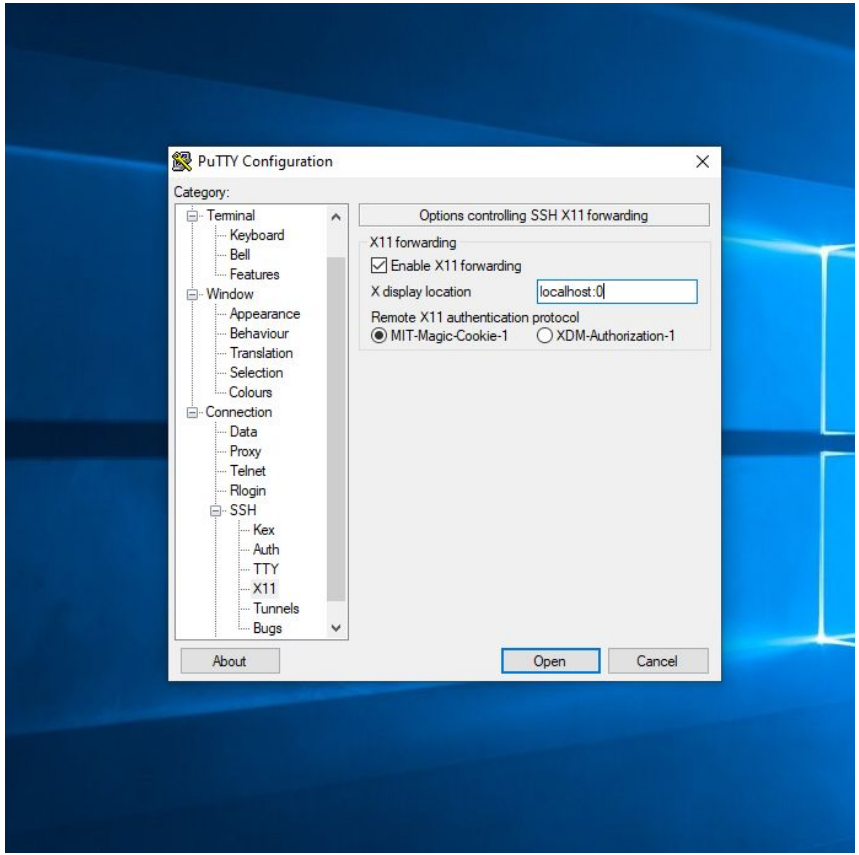
MobaXterm


MobaXterm
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
X server Exit
Quick connect...
User sessions
PuTTY sessions
chimera.hpc.mssm.edu (fluder
chimera.hpc.mssm.edu (fluder
minerva.hpc.mssm.edu (fluder
minerva5.hpc.mssm.edu (fluder
WSL-Ubuntu
3. /home/mobaxterm
• MobaXterm Personal Edition v21.5 •
(X server, SSH client and network tools)
➤ Your computer drives are accessible through the /drives path
➤ Your DISPLAY is set to 192.168.1.165:0.0
➤ When using SSH, your remote DISPLAY is automatically forwarded
➤ Each command status is specified by a special symbol (✔ or ✘)
• Important:
This is MobaXterm Personal Edition. The Professional edition
allows you to customize MobaXterm for your company: you can add
your own logo, your parameters, your welcome message and generate
either an MSI installation package or a portable executable.
We can also modify MobaXterm or develop the plugins you need.
For more information: https://mobaxterm.mobatek.net/download.html
17/03/2022 11:10.39 /home/mobaxterm
UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net
Type here to search
Desktop
50°F Cloudy
1:10 PM
3/17/2022

Putty- Initial Screen



Putty= X11 screen



Logging in - login nodes

3 login nodes: minerva[12-14], which points to the login node **li03c[02-04]**

- only available within campus-network

Users	Login method	Login servers	Password Components
Sinai users	userID	**@minerva.hpc.mssm.edu** **or specific nodes:** **@minerva12.hpc.mssm.edu** **@minerva13.hpc.mssm.edu** **@minerva14.hpc.mssm.edu**	Sinai Password followed by 6 Digit Symantec VIP token code
External users			

Note: Load balancer **Round-robin** is configured for **minerva.hpc.mssm.edu.** It will distribute client connections to the nearest across a group of login nodes.

Logging in - Config file

- **~/.ssh/config at your local workstation**
- **%USERPROFILE%\.ssh\config on Windows**
  - Set ControlMaster to reuse ssh connection for all hosts
  - Enable X11 forwarding
  - Set alias for hostname, so just type **ssh minerva** for login

```
$ cat ~/.ssh/config
Host *
        ControlMaster auto
        ControlPath /tmp/ssh_mux_%h_%p_%r
        ControlPersist 24h
        PreferredAuthentications        keyboard-interactive
        ServerAliveInterval 240
        ServerAliveCountMax 2
        ForwardX11 yes
        ForwardX11Timeout   12h

Host minerva
        Hostname minerva.hpc.mssm.edu
        User gail01
```

Minerva Storage

- Storage is in folders and subfolders. In linux, subfolders are separated by “/”
- 4-ish folders you can have (Possibly multiple project folders)
- Use **showquota** to show /sc/arion usage by user or project
  - *$showquota -u gail01 arion* or *$showquota -p projectname arion*

Home	/hpc/users/<userid> $ **quota -s**	• 20GB quota. • Slow. Use for “config” files, executables…NOT DATA • NOT purged and is backed up
Work	/sc/arion/work/<userid> $ **df -h** /sc/arion/work<userid>	• 100GB quota • Fast, keep your personal data here • NOT purged but is NOT backed up
Scratch	/sc/arion/scratch/<userid> $ **df -h** /sc/arion/scratch	• Free for all, shared by all; For temporary data • Current size is about 100TB • Purge every 14 days and limit per user is 10TB
Project	/sc/arion/projects/<projectid> $ **df -h** /sc/arion/projects/<projectid>	• PI’s can request project storage by submitting an allocation request at here,and get approval from allocation committee; Fee schedule and policy here. • Not backed up • Incurs charges **$100**/TiB/yr

User Software Environment: Lmod

> 1000 modules, and different versions are supported on Minerva

Lmod Software Environment Module system implemented:

- Search for module: **$module avail** or **$ module spider**
  Check all available R versions *$ ml spider R*

  …….R/3.3.1, R/3.4.0-beta, R/3.4.0, R/3.4.1, R/3.4.3_p, R/3.4.3, R/3.5.0, R/3.5.1_p, R/3.5.1, R/3.5.2, R/3.5.3
- To check the detailed PATH setting in module files: **$ml show R**
- Load module: **$ml python or $module load python** or **$ml python/2.7.16** (for a specific version)
- Unload module **$ml -gcc or $module unload gcc**
- List loaded modules: **$ml or $module list**
- Purge ALL loaded modules **$ ml purge**
- Autocompletion with tab
- More at Lmod user guide

```
gail01@li03c03: ~ $ ml python
gail01@li03c03: ~ $ ml

Currently Loaded Modules:
  1) gcc/8.3.0   2) python/3.7.3



gail01@li03c03: ~ $ ml python/2.7.16

The following have been reloaded with a version change:
  1) python/3.7.3 => python/2.7.16

gail01@li03c03: ~ $ ml -gcc
```

User Software Environment - Major packages

OS: **Centos 7.6** with **glibc-2.17(GNU C library) available**

GCC: system default /usr/bin/gcc is gcc 4.8.5

$ module load gcc (default is 8.3.0) or *$ ml gcc*

$ ml python Python: default version 3.7.3 (it will load python and all available python packages)

Note: python2 or python3 *$ml python/2.7.16*

$ ml R R: default version 4.2.0 it will load R and all available R packages)

$ml CPAN Collection of system Perl software: default system version 5.16.3

$ml anaconda3 Anaconda3: default version 2018-12

$ml java java: default version 1.8.0_211

SAS access: *$ml sas*

- The cost for the license is **$150.00** per activation, and request form at here

Matlab access: *$ml matlab*

  - The cost for the license is **$100.00** per activation, and request form at here.

User Software Environment - Anaconda Distribution

- Anaconda3/Anaconda2: Support minimal conda environments (such as tensorflow, pytorch, qiime) e.g., tensorflow (both in CPU and GPU)
- **To avoid incompatibilities with other python, clear your environment with module purge before loading Anaconda**

  $ml purge

  $ml anaconda3/2020.11

  $conda env list ***# get a list of the env available (Or*** *$conda info --envs)*

  source activate tfGPU2.4.1
- User should install their own envs locally, (see more guide here)
  - ➔ Use option -p PATH, --prefix PATH Full path to environment location (i.e. prefix).

    $conda create python=3.x -p /sc/arion/work/gail01/conda/envs/myenv

    $conda env create -p myenv -f environment.yml
  - ➔ Set envs_dirs and pkgs_dirs in .condarc file, specify directories in which environments and packages are located

    $conda create -n myenv python=3.x
- Set conda base auto-activation false

  conda config --set auto_activate_base false

  More at Conda config guide

```
$ cat ~/.condarc file
envs_dirs:
- /sc/arion/work/gail01/conda/envs
pkgs_dirs:
- /sc/arion/work/gail01/conda/pkgs
conda config --set auto_activate_base false
```

User Software - Singularity Container Platform

Singularity tool is supported, instead of docker (Security concern)

- Docker gives superuser privilege, thus is better at applications on VM or cloud infrastructure
- It allows you to create and run containers that package up pieces of software in a way that is portable and reproducible. Your container is a single file and can be ran on different systems

To load singularity module: *$ module load singularity/3.11.0*

To pull a singularity image: *$ singularity pull --name hello.simg shub://vsoch/hello-world*

To create a container within a writable directory (called a sandbox):

$singularity build --sandbox lolcow/ shub://GodloveD/lolcow (create container within a writable directory)

To pull a docker image: *$singularity pull **docker://**ubuntu:latest*

To shell into a singularity image: *$ singularity shell hello.simg*

To run a singularity image: *$ singularity run hello.simg*

To get a shell with a specified dir mounted in the image

$ singularity run -B /user/specified/dir hello.simg

Note: /tmp, user home directory, and /sc/arion/is automatically mounted into the singularity image.

User Software - Singularity Container

To build a new image from recipe file/definition file:

Use Singularity **Remote Builder** or your local workstation

- Singularity build is not fully supported due to the sudo privileges for users
- Using the Remote Builder, you can easily and securely create containers for your applications without special privileges or set up in your local environment
- Write your recipe file/definition file https://sylabs.io/guides/3.6/user-guide/definition_files.html
- Convert docker recipe files to singularity recipe files:

  $ml python

  $spython recipe Dockerfile Singularity

For more information about Singularity on Minerva, please check our training slide here

User Software - How to Run Jupyter Notebook

One simple command to get interactive **web** sessions in a HPC LSF job (Available on login nodes only) with details here

Option1: $minerva-jupyter-module-web.sh (--help to get help message/usage)

[INFO] This script is to submit a Python Jupyter Notebook web instance inside an
[INFO] LSF job on ***one single host*** for users.
[INFO] By default, this script uses Jupyter from **python/3.7.3**
[INFO] You can load other ***python version and other modules needed*** for your Jupter Notebook by -mm option

You can load Minerva modules needed for your Jupyter Notebook

Option 2: $minerva-jupyter-web.sh (--help to get help message/usage)

[INFO] This script is to submit a ***Singularity containerized*** Jupyter Notebook web instance inside an
[INFO] LSF job on ***one single host*** for users.
[INFO] By default, this script uses this **Singularity image** (shub://ISU-HPC/jupyter)

For users who want an isolated/clean env working with container image. You need to install/maintain your own python related package. No module system setup

User Software - How to Run Jupyter Notebook

Option 1 (con't): Access Jupyter notebook running on Minerva compute node via port forwarding

1) You can use one simple command wrapper mentioned above:$minerva-jupyter-module-web.sh

OR

2) Issue commands step by step with more control by yourself:

start an interactive session for example

$bsub -P acc_xxx -q interactive -n 2 -R "span[hosts=1]" -R rusage[mem=4000] -W 3:00 -Is /bin/bash

#Then on the allocated nodes lc01c30, start Jupyter Notebook

lc01c30 $ml python

lc01c30$jupyter notebook --no-browser --port=8889

#On your local workstation, forward port XXXX(8889) to YYYY(8888) and listen to it

$ssh -t -t -L localhost:8888:localhost:8889 gail01@minerva.hpc.mssm.edu ssh -X lc01c30 -L localhost:8889:localhost:8889

#Open firefox on local: http://localhost:8888

* note: you can change the portal number 8888/8889 to others

User Software - How to Run Jupyter Notebook

Option 2 (con't): On-the-fly Jupyter Notebook in a Minerva job $minerva-jupyter-web.sh

- Containerized application for workflow reproducibility, packages installed in $HOME/.local
- See usage: `minerva-jupyter-web.sh -h`
- No module system setup.

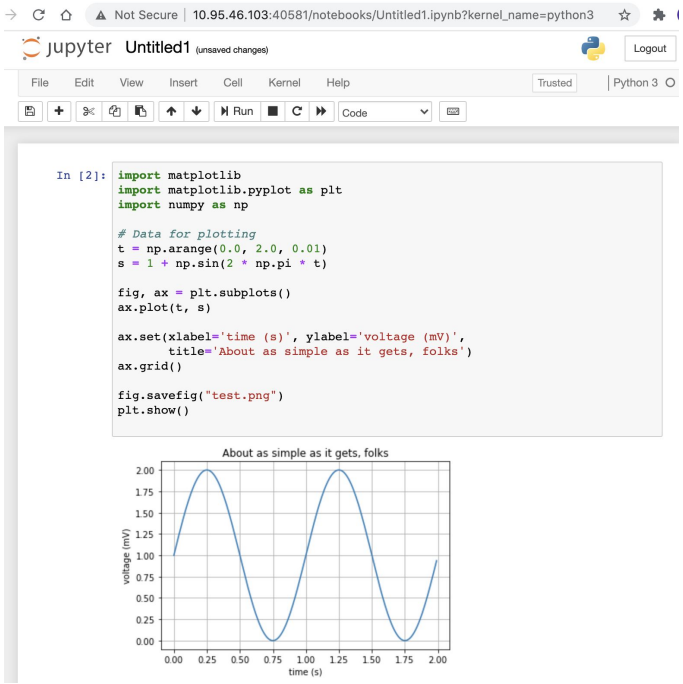
To install your own python packages:

- Open the terminal in web the jupyter web, type

```
pip install packages
--user
```

This will be in your home directory $HOME/.local.

restart the jupyter notebook



User Software - How to Run Jupyter Notebook

Summary

	minerva-jupyter-module-web.sh	minerva-jupyter-web.sh
Access modules on Minerva	Yes	No
Using singularity image	No	Yes
Support GPU node	Yes	Yes
Python versions	By default, python/3.7.3; You can load other *python version and other modules needed* for your Jupter Notebook by -mm option	This script uses the python within this Singularity image (shub://ISU-HPC/jupyter)
Others	For users who want to access Minerva modules.	For users who want an isolated/clean env working with a container image. You need to install/maintain your own python related package. No module system setup

User Software - Rstudio

Option 1: On-the-fly Rstudio over Web in a Minerva job $minerva-rstudio-web-r4.sh

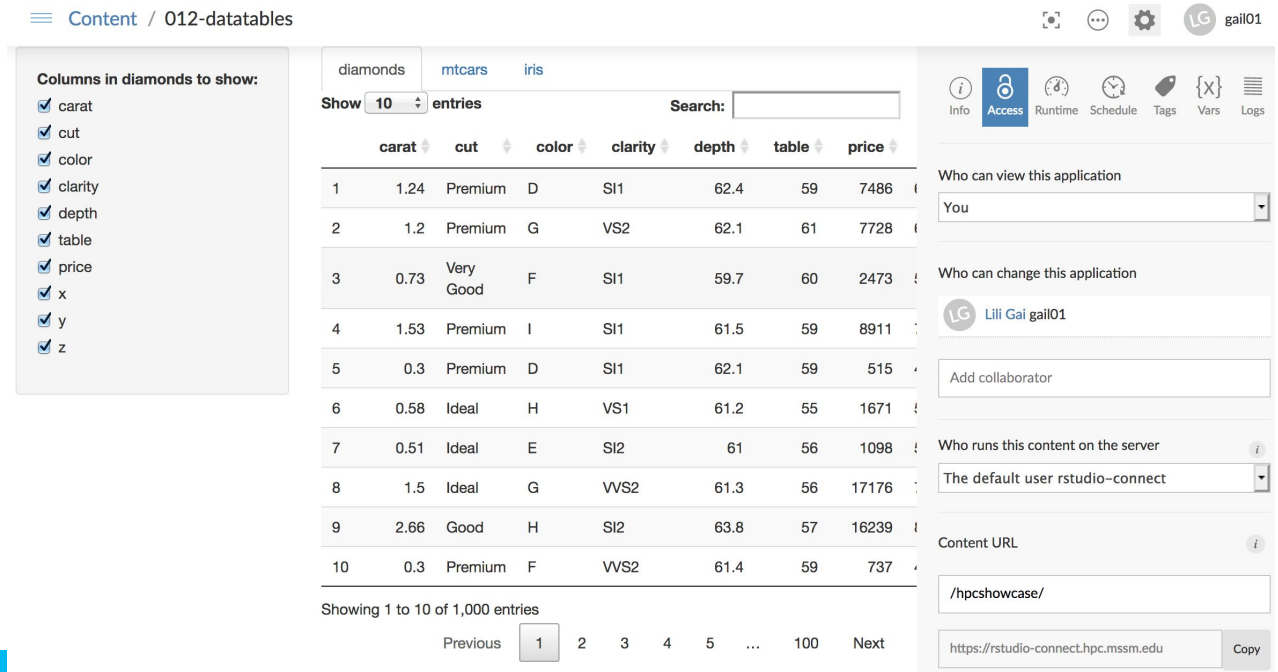
- One simple command to get interactive **web** sessions in a HPC LSF job
- Available on login nodes only
- Containerized application for workflow reproducibility, packages installed in $HOME
  - Since this is a container env, you need to install/maintain your own R related package. No module system setup.
- See usage with details:
  - `minerva-rstudio-web-r4.sh -h`

Option 2: Run rstudio over GUI (graphical user interface)

- Enable X11 forwarding (see P.7 & P.9)
- ml rstudio; rstudio

Posit Connect (formerly Rstudio Connect) server https://rstudio-connect.hpc.mssm.edu

- You can publish Shiny, R Markdown for collaborators or others
- If interested in publishing on Rstudio-connect, please check instruction at https://labs.icahn.mssm.edu/minervalab/documentation/r/#rstudioconnect



User Software Environment - some config

- You can load modules in your **.bashrc** script to load them on startup

  or check out User Collections:

https://lmod.readthedocs.io/en/latest/010_user.html#user-collections-label

- You can create your own modules and modify MODULEPATH so they can be found by

  module use /hpc/users/fludee01/mymodules

  or

  export MODULEPATH=/hpc/users/fludee01/mymodules:$MODULEPATH
- You can set PATH or PYTHONPATH by

  export PATH=/hpc/users/gail01/.local/bin:$PATH

  export PYTHONPATH=/hpc/users/gail01/.local/lib/python3.7/site-packages:$PYTHONPATH

File Transfer - Globus (Preferred)

- Globus is developed/maintained at the University of Chicago and used extensively at HPC centers
- Globus makes it easy to move/sync/share **large amounts of data**.
- Globus will **retry failures, recover from faults automatically when possible, and report the status of your data transfer**. Globus website

Globus on Minerva under HIPAA+BAA subscription

- *Be able to share data with their identity/email address. No Minerva account needed*
- *Can upgrade your Globus account to Plus, enabling file transfer between two personal Globus endpoints and data share from a Globus Connect Personal endpoint*

Data transfer with Globus on Minerva (see instructions here)

- *Login to Globus with Mount Sinai school email (eg, first.last@mssm.edu)*
- *Minerva collections: MSSM Minerva User Home Directories and MSSM Minerva Arion FileSystem*
- ***Use HTTPS for download/upload:** Now you can move data within your browser, without installing Globus Connect Personal; you'll see options for upload and download in the Globus web app.*
- *Users handling HIPAA/sensitive data on machines running Globus Connect Personal, please check High Assurance in the preference*

File Transfer - Con't

- **SCP, SFTP**
  - Good for relatively small files, not hundreds of TB's. Not recommended.
  - *Some scp apps for Windows/Mac use cached password. This feature must be turned off.*
  - *ftp is not supported on Minerva due to security risk*
  - *Note when you use VPN, data transfer between Minerva and your local computer may be pretty slow because the bandwidth is limited by school IT*

- **On Minerva**
  - After login to Minerva, ssh *li03c01* for data transfer, no time limit
  - minerva12/13/14 (33h) or interactive nodes (12h).
  - Please use a screen session so that you can return to your work after the drop of the connection.

Archiving Data: IBM Tivoli Storage Management (TSM)

- Keep for 6 years with two copies
- Can be accessed via either a GUI or the command line

```
$ module load java
$ dsmj -se=userid
```

or

```
$ dsmc -se= userid
```

- Large transfers can take a while. Use a ***screen*** session and disconnect to prevent time-outs
- Full more details here
- **Collaboration account:**
  - If your group needs a collaboration account for group related tasks like archiving a project directory or managing group website, please check https://labs.icahn.mssm.edu/minervalab/minerva-quick-start/collaboration-account

Web server

- Your website at **https://userid.u.hpc.mssm.edu**
- The document root for a user's site is within home folder in a folder called ~/www
- **NO PHI may be shared via the webserver.**

Step 1: Create ~/www. `$ mkdir ~/www`

Step 2: Place content (eg. index.html) put files or create symlink (from arion) in the www folder

```
$ cat > ~/www/index.html <<EOF
Hello World from my website.
EOF
```

Step 3: **Authentication (optional but recommended)**

If you use your website for file sharing, we strongly recommend you to set up password protection for your files. Please refer to the "Authentication" part of the instructions, located here:
https://labs.icahn.mssm.edu/minervalab/documentation/web-services/

Web Server

The ***indexes* option** is turned off by default for security reasons. You will an see error message "Forbidden, You don't have permission to access this resource." if you don't have an **index.html/ index.php** file under the folder.

You can enable this option in the htaccess file in order to list your files, for example:

```
[gail01@li03c03 ~]# cat /hpc/users/gail01/www/.htaccess
Options +Indexes
```

However, access to /sc and /hpc/users will be disabled around 1 Jan 2023. Alternate methods to publish data are being investigated.

Web server https://users.hpc.mssm.edu/

Some demos on setting up your first python flask and dash app

https://gail01.u.hpc.mssm.edu/flask_demo/

https://gail01.u.hpc.mssm.edu/dash_demo/

Code is at https://gail01.u.hpc.mssm.edu/code/

Load Sharing Facility(LSF)

A Distributed Resource Management System

bsub - submit a batch job to LSF

- command job submission: bsub [options] command

  $ bsub -P acc_hpcstaff -q premium-n 1 -W 00:10 echo "Hello Chimera"

- LSF script submission: bsub [options] < my_batch_job (Options on the command line override what is in the script)

```
gail01@li03c03: ~ $ cat myfirst.lsf
#!/bin/bash
#BSUB -J myfirstjob                    # Job name
#BSUB -P acc_hpcstaff                  # REQUIRED; To get allocation account, type "mybalance"
#BSUB -q premium                       # queue; default queue is premium
#BSUB -n 1                             # number of compute cores (job slots) needed, 1 by default
#BSUB -W  6:00                         # REQUIRED; walltime in HH:MM
#BSUB -R rusage[mem=4000]              # 4000 MB of memory request per "-n"; 3000 MB by default
#BSUB -oo %J.stdout                    # output log (%J : JobID)
#BSUB -eo %J.stderr                    # error log
#BSUB -L /bin/bash                     # Initialize the execution environment

echo "Hello Chimera"                   # Command that you need to run
gail01@li03c03: ~ $ bsub < myfirst.lsf
Job <2937044> is submitted to queue <premium>.
```

LSF: batch job submission examples with bsub

Interactive session:

 # interactive session

$ bsub -P acc_hpcstaff -q interactive -n 1 -W 00:10 -Is /bin/bash

 # interactive GPU nodes, flag "-R v100" is required

$ bsub -P acc_hpcstaff -q interactive -n 1 -R v100 -gpu num=1 -R span[hosts=1] -W 01:00 -Is /bin/bash

Batch jobs submission:

 # simple standard job submission

$ bsub -P acc_hpcstaff -q premium-n 1 -W 00:10 echo "Hello World"

 # GPU job submission if you don't mind the GPU card model

$ bsub -P acc_hpcstaff -q gpu -n 1 -gpu num=1 -R span[hosts=1] -W 00:10 echo "Hello World"

*# himem job submission, flag "**-R himem**" is required*

*$ bsub -P acc_hpcstaff -q **premium** -n 1 -R himem -W 00:10 echo "Hello World"*

Last but not Least

Got a problem? Need a program installed? Send an email to:

hpchelp@hpc.mssm.edu