

Introduction to Minerva

Minerva Scientific Computing Environment

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

Jielin Yu, PhD
The Minerva HPC Team

February 13 2025



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:

- 4x **login nodes** - Intel Emerald Rapids 8568Y+, 2.3GHz – 96 cores with 512 GB memory per node.
- Compute nodes -
 - 146 **high memory nodes** - Intel Emerald Rapids 8568Y+, 2.3GHz - 96 cores with 1.5 TB memory per node.)
 - 37 **high memory nodes** - Intel 8168/8268, 2.7/2.9GHz - **1.5 TB** mem/node
 - **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
 - 2 - Intel 8358,2.6GHz - 2 TB memory - 4x A100- **80 GB** GPU
 - 2 - Intel 8358 2.6 GHz- 500 GB memory - 4x H100- **80 GB** GPU
 - 47 - Intel ER 8568Y+, 2.3GHz - 1.5 TB memory - 4x H100- **80 GB** GPU
 - 4 - AMD Genoa 9334 2.7GHz - 1.5 TB memory - 8x L40S- **48 GB** GPU

NIH FUNDED NODES

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: 32 PB 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**



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 Rocky 9

- 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](#)
 - Hospital user will need to request a “School Network Account” via [SailPoint](#).
- Register your token at the Self Service Portal **school site** (<https://register4vip.mssm.edu/vipssp/>)
- Complete forms at <https://forms.hpc.mssm.edu/> to activate your account after it is created.
- SSH client: terminal (Mac), MobaXterm/Putty (Windows)
- Send a request to [ASCIT](#) to add your school account to **MSSM2FA_Users** group.
- 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

HIPAA and NIH Acknowledgement Forms

- 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 the NIH acknowledgement, then sign both agreement forms **annually** (every Jan.) at
<https://forms.hpc.mssm.edu/>
- Users who have not signed both agreements will have their accounts locked until all the agreements are signed.

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`

```
[yuj25@li04e03 ~]$ minerva_help
```

Minerva HPC website: <https://labs.icahn.mssm.edu/minervalab/>
Minerva Documentation at <https://labs.icahn.mssm.edu/minervalab/documentation/>
Minerva training slides at <https://labs.icahn.mssm.edu/minervalab/resources/the-minerva-user-group-and-training-classes/>

Publish R/shiny to Rstudio Connect server: <https://labs.icahn.mssm.edu/minervalab/rstudio-connect-server/>
Data transfer: <https://labs.icahn.mssm.edu/minervalab/documentation/file-transfer-globus/>
Web service: <https://labs.icahn.mssm.edu/minervalab/documentation/web-services/>
TSM data archive: <https://labs.icahn.mssm.edu/minervalab/documentation/access-tsm-with-command-line/>

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-module.sh (with details at <https://labs.icahn.mssm.edu/minervalab/documentation/r/>)

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 current job 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

Logging in - login nodes

4 login nodes: minerva[11-14], which points to the login node **li04e[01-04]**

- only available within campus-network

Users	Login method	Login servers	Password Components
Sinai users	userID	@minerva.hpc.mssm.edu or specific nodes: @minerva11.hpc.mssm.edu @minerva12.hpc.mssm.edu @minerva13.hpc.mssm.edu @minerva14.hpc.mssm.edu	Sinai Password followed by 6 Digit Symantec VIP token code
External users			

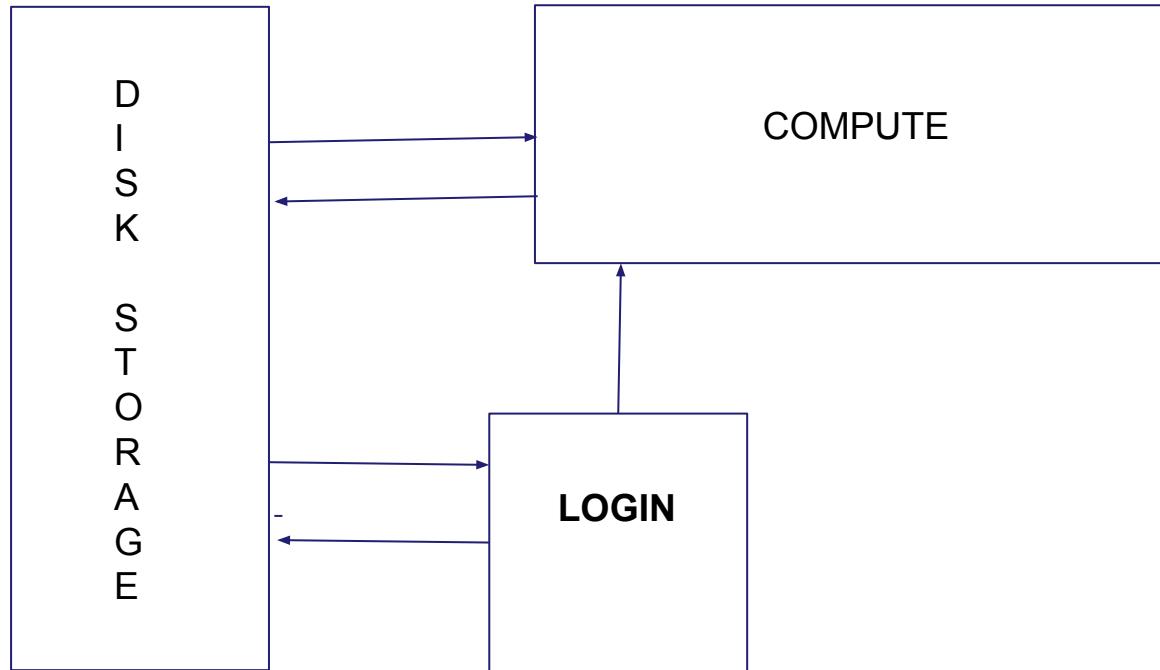
Note: Load balancer **Round-robin** is configured for **minerva.hpc.mssm.edu**. It will distribute client connections to the nearest across a group of login nodes.

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	<code>/hpc/users/<userid></code> <code>\$ quota -s</code>	<ul style="list-style-type: none">• 20GB quota.• Slow. Use for “config” files, executables...NOT DATA• NOT purged and is backed up
Work	<code>/sc/arion/work/<userid></code> <code>\$ df -h /sc/arion/work<userid></code>	<ul style="list-style-type: none">• 100GB quota• Fast, keep your personal data here• NOT purged but is NOT backed up
Scratch	<code>/sc/arion/scratch/<userid></code> <code>\$ df -h /sc/arion/scratch</code>	<ul style="list-style-type: none">• Free for all, shared by all; For temporary data• Current size is about 100TB• <u>Purge every 14 days and limit per user is 10TB</u>
Project	<code>/sc/arion/projects/<projectid></code> <code>\$ df -h /sc/arion/projects/<projectid></code>	<ul style="list-style-type: none">• 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 <u>\$119/TiB/yr</u>

Minerva Cluster from 30,000 ft.



Minerva Ticket Submission

- Send an email to: hpchelp@hpc.mssm.edu to start a ticket.
- Information to include:
 - The error message or the location of the log file on Minerva.
 - use command “pwd” to check the current working directory.
 - All the commands you used to get the error message.
 - The location of the scripts used and how did you run the script.
 - Which node.
 - `yuj25@li04e04`
 - The job ID and location of the job script.
 - Job <`123456789`> is submitted to queue <premium>.
- Please do not include any “<” in the email.

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/3.12.5` (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:
 - [Minerva Lmod guide](#)
 - [Lmod user guide](#)

User Software Environment - Major packages

OS: Rocky 9 with glibc-2.34(GNU C library) available

GCC: system default /usr/bin/gcc is gcc 11.4

`$ module load gcc` (default is 14.2.0) or `$ ml gcc`

`$ ml python` Python: default version 3.12.5 (it will load python and all available python packages)

`$ ml R` R: default version 4.2.0 (it will load R and all available R packages), latest version 4.4.1

`$ml CPAN` Collection of system Perl software: default system version 5.40.0

`$ml anaconda3` Anaconda3: default version 2024-06

`$ml java` java: latest version 21.0.4

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

- <https://labs.icahn.mssm.edu/minervalab/documentation/conda/>
- To avoid incompatibilities with other python, clear your environment with module purge before loading Anaconda

\$ml purge

\$ml anaconda3/2024.06

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

- 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/3.6.4`

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/specifyed/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 sessions [here](#)

User Software - How to Run Jupyter Notebook

Option 1 : On-the-fly Jupyter Notebook in a Minerva job

<https://labs.icahn.mssm.edu/minervalab/documentation/python-and-jupyter-notebook/#jupyter>

- Scripts in /usr/local/bin on head nodes only:
 - (Not recommended) `minerva-jupyter-web.sh`
 - Runs jupyter installed inside a singularity container
 - No module system setup
 - `minerva-jupyter-module-web.sh`
 - Runs jupyter installed in module python/3.12.5
 - Can access modules on minerva
 - Can load users' conda environments (jupyter must be installed in conda env)
 - `minerva-jupyter-module-web.sh -mm anaconda3/2024.06 -env ENV_NAME`
 - Install jupyter in your conda env: `conda install jupyter`
 - Do not have any conda env activated before running the script.
- Add --help to the script to get help message/usage

User Software - How to Run Jupyter Notebook

Option 2 : Jupyter Notebook in a batch job

- Primarily, the nbconvert tool allows you to convert a Jupyter .ipynb notebook document file into another static format including HTML, LaTeX, PDF, Markdown, reStructuredText, and more. nbconvert can also add productivity to your workflow when used to execute notebooks programmatically.

<https://nbconvert.readthedocs.io/en/latest/usage.html>

- Use the commands below in your job script:

```
ml python/3.12.5
```

```
jupyter nbconvert --to notebook --execute myfile.ipynb
```

- The result is saved in a file named **myfile.nbconvert.ipynb**

Option 3 : Via Open Ondemand

User Software - Rstudio

Option 1 : On-the-fly Rstudio over Web in a Minerva job

<https://labs.icahn.mssm.edu/minervalab/documentation/r/#rstudio>

- Scripts in /usr/local/bin on login nodes only:
 - (Not recommended) Scripts run RStudio Server installed inside singularity containers, no module system setup:
 - `minerva-rstudio-web-r4.sh`
 - `minerva-rstudio-web-r4.2.3.sh`
 - Scripts with module system setup:
 - `minerva-rstudio-web-module.sh`
- Add --help to the script to get help message/usage

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

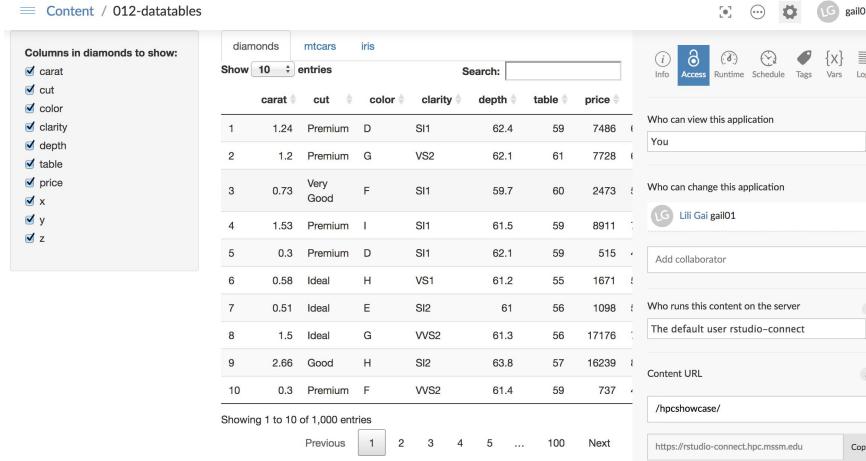
- Enable X11 forwarding
- Run “`ml rstudio proxies; rstudio`”

Option 3 : Via Open Ondemand

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>
- To login to the web: rstudio-connect.hpc.mssm.edu, please use your minerva username and password joined with <6 digit vip token>, like ssh in to the login node.



The screenshot shows the RStudio Connect interface for a data visualization application. The left sidebar lists 'Content' and the current page '012-databatables'. The main area displays a table with columns: diamonds, mtcars, and iris. The 'diamonds' column is selected. The table shows 10 entries of diamond data. The right sidebar includes sections for 'Info', 'Access' (selected), 'Runtime', 'Schedule', 'Tags', 'Vars', and 'Logs'. It also shows who can view and change the application, and an 'Add collaborator' field. The 'Content URL' is set to '/hpcshowcase/' and the full URL is https://rstudio-connect.hpc.mssm.edu.

	carat	cut	color	clarity	depth	table	price
1	1.24	Premium	D	SI1	62.4	59	7486
2	1.2	Premium	G	VS2	62.1	61	7728
3	0.73	Very Good	F	SI1	59.7	60	2473
4	1.53	Premium	I	SI1	61.5	59	8911
5	0.3	Premium	D	SI1	62.1	59	515
6	0.58	Ideal	H	VS1	61.2	55	1671
7	0.51	Ideal	E	SI2	61	56	1098
8	1.5	Ideal	G	VVS2	61.3	56	17176
9	2.66	Good	H	SI2	63.8	57	16239
10	0.3	Premium	F	VVS2	61.4	59	737

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**
 - minerva11/12/13/14 (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>

Archiving Data: IBM Tivoli Storage Management (TSM)

- We have updated our Minerva Tivoli Storage Manager (TSM) archival policy:
https://labs.icahn.mssm.edu/minervalab/wp-content/uploads/sites/342/2024/08/Minerva-TSM-PolicyV1_Aug2024.pdf
- 1 GB is the minimum file size to be transferred to TSM. Files <1 GB must be aggregated with tar/zip, resulting in a file size of 1 GB or larger.
- The maximum amount of data archived or retrieved from TSM is limited to 40 terabytes per week.
- TSM is for long-term file storage only, with each file stored once.
- Data archived in TSM has a retention time of six years. We will notify you by email three months in advance before your file will be deleted from TSM.
- Failure to adhere to this policy may result in your archival or retrieval processes being killed, and may ultimately result in removal of your access to TSM.

Web server

- As of October 1st 2022, there are two different Domain Name System for user website's landing point with different network access:
 - <https://userid.u.hpc.mssm.edu> for internal websites
 - <https://userid.dmz.hpc.mssm.edu> for public websites
- By default, each user's default web services landing point is <https://userid.u.hpc.mssm.edu>, with only internal access (campus network or VPN are needed for access).
- If you need public websites for your research, please fill out [the form](#) or at <https://redcap.link/g08ytzki>. Once the request is received, we will send your PI the policy form to sign:
https://sc.u.hpc.mssm.edu/HPC/Comprehensive_Webserver_Security_Policy_for_DMZ_Server.pdf
Then we will forward the request to the IT security team, who will scan the web application. If no critical/high vulnerabilities reported, we will move the webpage to userid.dmz.hpc.mssm.edu for public access. The time to complete this request will be depending on the vulnerability status of the website. A rough estimate is 1 week.

Web server

- 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 strongly 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 see an 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@li04e03 ~]# cat /hpc/users/gail01/www/.htaccess  
Options +Indexes
```

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://yuj25.u.hpc.mssm.edu/code/>

Acknowledgements

- ▶ Supported by the Clinical and Translational Science Awards (CTSA) grant UL1TR004419 from the National Center for Advancing Translational Sciences, National Institutes of Health.



Last but not Least

Got a problem? Send an email to:

hpchelp@hpc.mssm.edu