SDSC Summer Institute 2021

Running Secure Jupyter Notebooks on Expanse

Preparation Day – Wednesday, July 28

Mary Thomas
SDSC

EXPANSE
COMPUTING WITHOUT BOUNDARIES

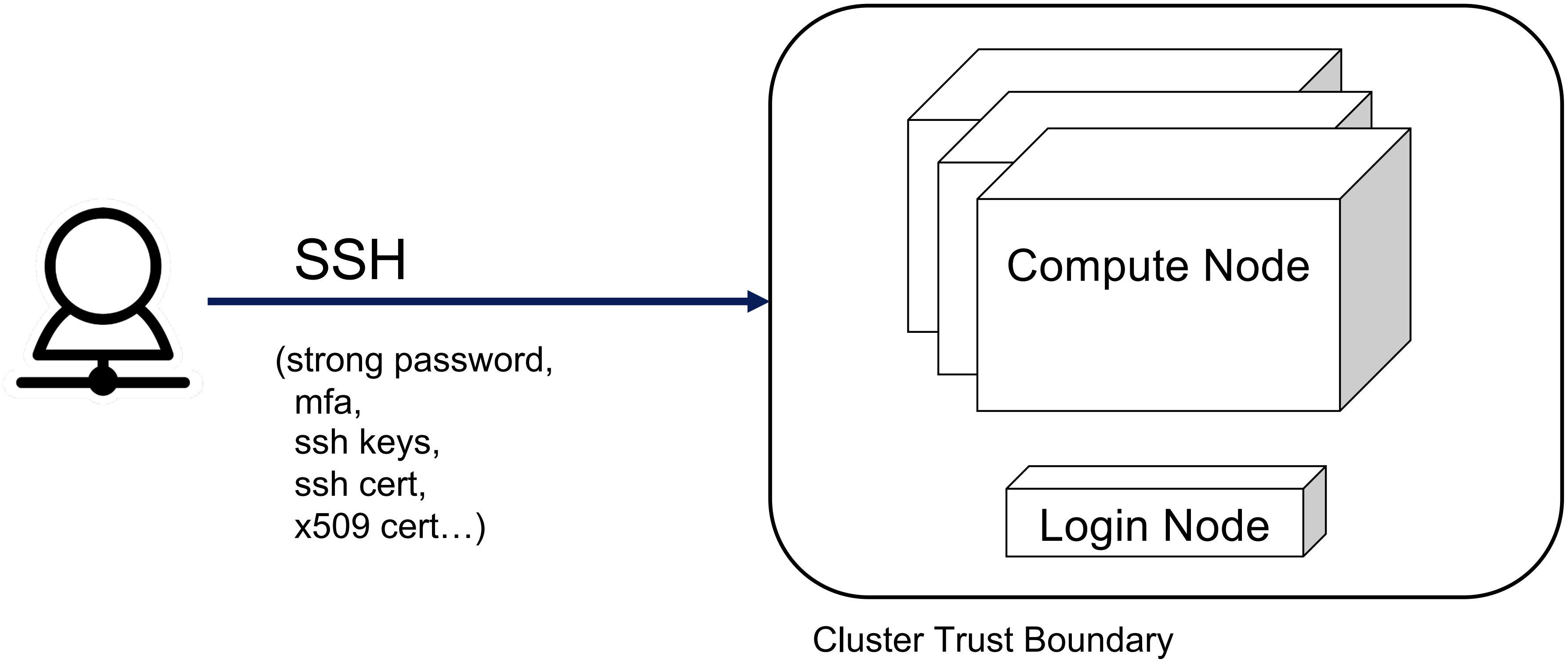
SAN DIEGO SUPERCOMPUTER CENTER

NSF Award 1928224

Secure Connections to HPC Systems are Important

External Side of the HPC System

Internal Side of the HPC System



SSH

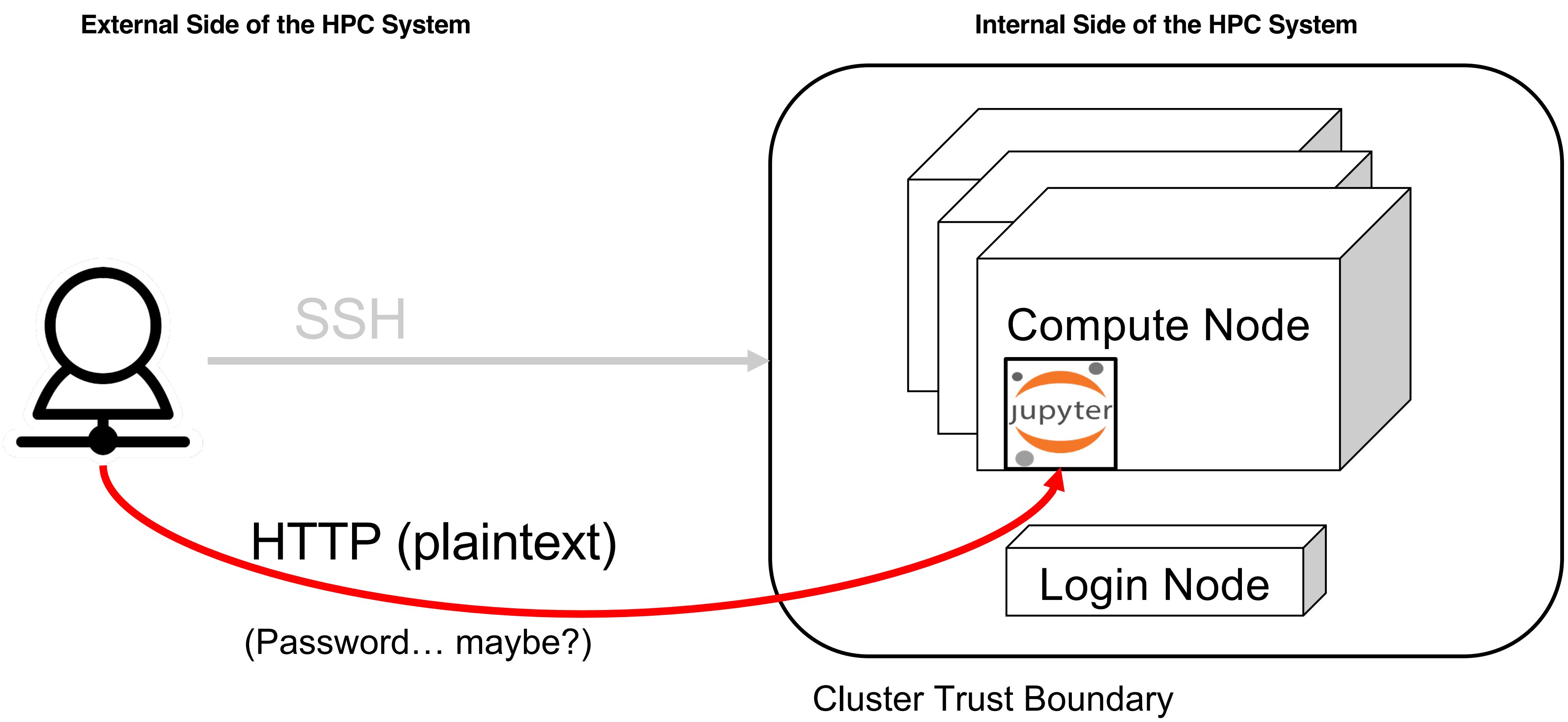
(strong password,
mfa,
ssh keys,
ssh cert,
x509 cert…)

Compute Node

Login Node

Cluster Trust Boundary

Jupyter Notebooks are Popular



But provide a plaintext back-door to the system

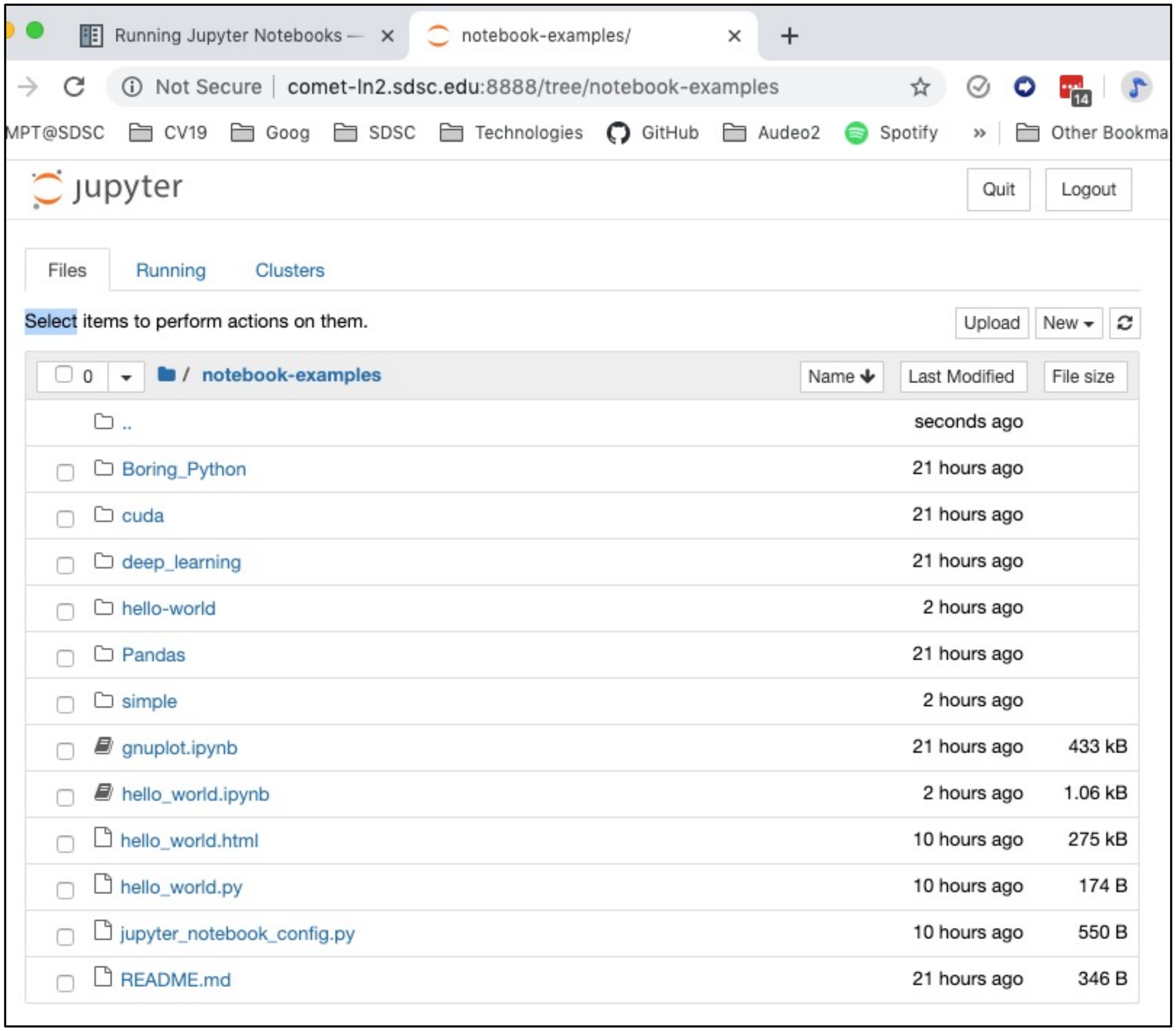
Jupyter Has a Key Vulnerability:
Jupyter Servers Provide Access to HPC File Systems

SDSC Jupyter Services Policy:

- Portals, JupyterHub, and other services cannot be mounted directly to disk (must be on VM or external)
  - Many use root in vulnerable ways
  - If a user launches Jupyter Lab or Notebooks, the jobs will be killed.
- No applications can run on login nodes

SDSC recommendation:

- use secure connections: when you choose insecure connections your account is vulnerable to hacking



Select items to perform actions on them.

/ notebook-examples	Last Modified	File size
..	seconds ago	
□ Boring_Python	21 hours ago	
□ cuda	21 hours ago	
□ deep_learning	21 hours ago	
□ hello-world	2 hours ago	
□ Pandas	21 hours ago	
□ simple	2 hours ago	
□ gnuplot.ipynb	21 hours ago	433 kB
□ hello_world.ipynb	2 hours ago	1.06 kB
□ hello_world.html	10 hours ago	275 kB
□ hello_world.py	10 hours ago	174 B
□ jupyter_notebook_config.py	10 hours ago	550 B
□ README.md	21 hours ago	346 B

Motivation: Make Doing the Right Thing Easier than the Wrong Things

A Wrong Thing: Plaintext to Compute Node

- Submit batch job.
- Wait till job runs.
- Figure out what node it’s on.
- Point web browser at node.

A Right Thing: Improve secure access:

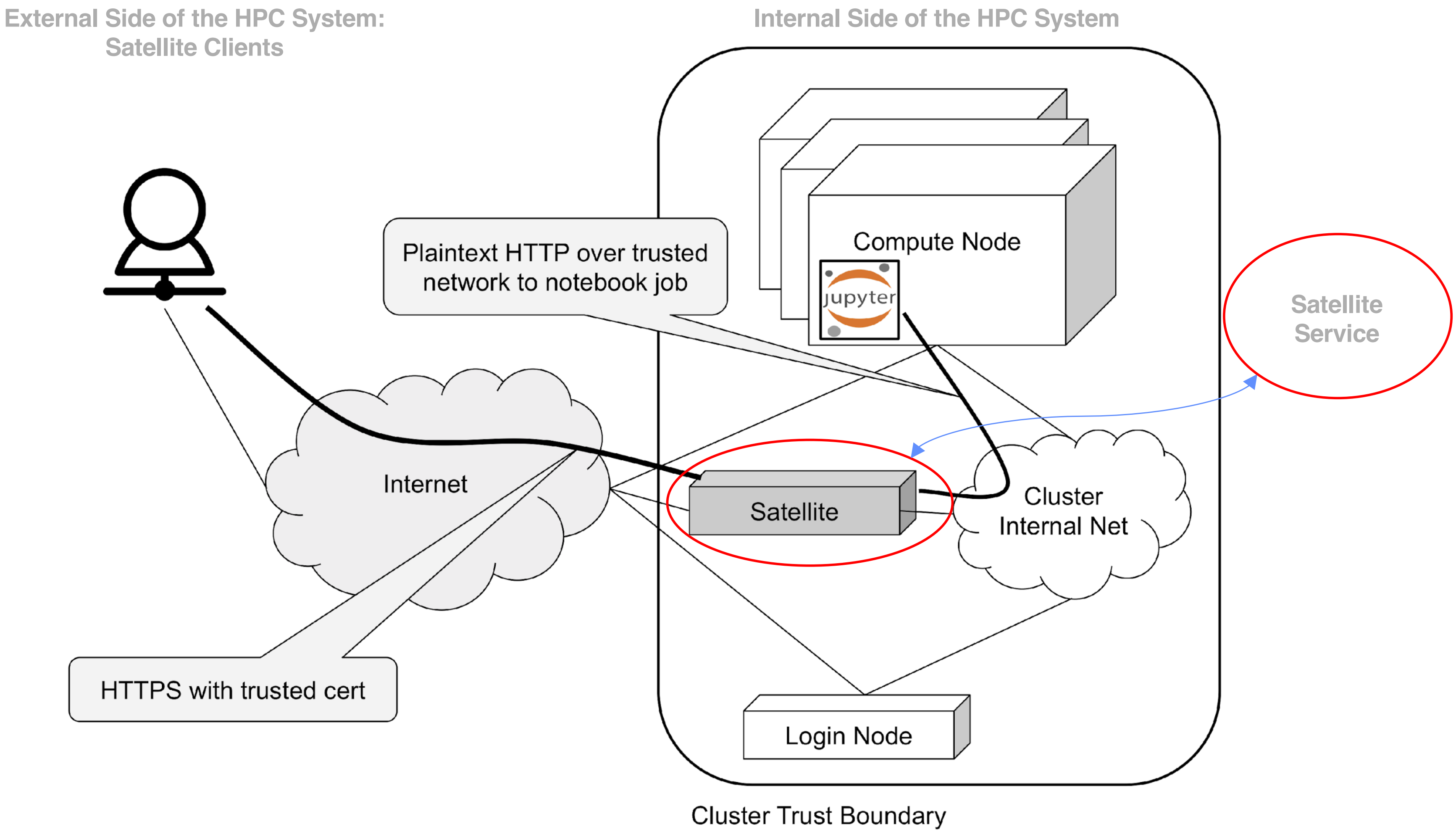
- Invoke **the Satellite Reverse Proxy Service**
- Point browser at secure, encrypted URL (HTTPS).
- (Wait until Jupyter Notebook shows up.)

A Simple Solution:
SDSC Satellite Reverse Proxy Service

Just Two Components!

- Satellite: a self-service HTTP(s) reverse-proxy.
- Satellite Client: a shell-based utility to orchestrate a user's interaction with both Satellite and Slurm to start a Jupyter session within a batch job.

Jupyter Notebooks With Satellite



Satellite Clients

start-jupyter

- 1st generation shell utility developed to orchestrate a user's interaction with both Satellite and Slurm to start a Jupyter session within a batch job.

- Key features in design:
  - User calls start-jupyter launch script, which requests token from Satellite, passes token to batch job script and submits the job to Slurm; token redeemed from batch job once it runs
  - Provided user with a prefabricated set of batch job scripts to choose from for certain popular applications on each system; user could modify/make their own custom batch job script
  - Small custom shell function library to make code more reusable

- Currently runs on: Expanse, Comet, TSCC, TSCC Stratus

https://github.com/sdsc-hpc-training-org/reverse-proxy

galyleo

- 2nd generation shell utility developed to orchestrate a user's interaction with both Satellite and Slurm to start a Jupyter session within a batch job.

- Developed while reviewing start-jupyter codebase to sort out how best to support Expanse (OOD) Portal and HPC User Services Group long-term; effectively recycled existing an SSH tunneling orchestration utility to use Satellite proxy service instead.

- Key features in design:
    - Recreate same interactions with Satellite service.
    - Increase flexibility for users to configure software environment; but also try to make it simpler for them to do themselves
    - Batch job script is generated completely on-the-fly.
    - Command-line argument driven.
    - Quiet mode for OOD portal

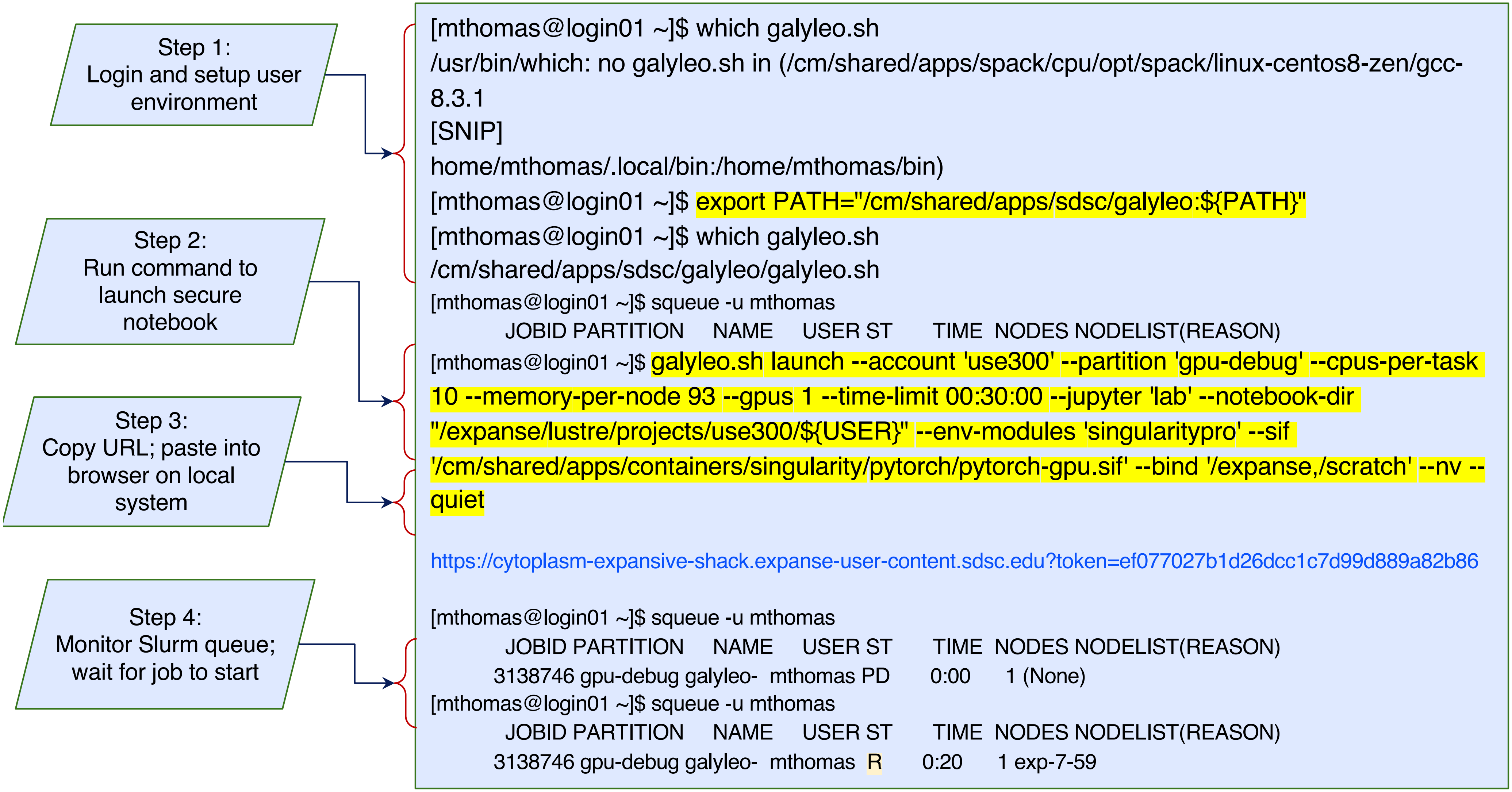
https://github.com/mkandes/galyleo

galyleo demo examples on Expanse

```
# Set location of galyleo directory on Expanse

export PATH="/cm/shared/apps/sdsc/galyleo:${PATH}"

# Example 1: Launch a Jupyter Notebook session on a single CPU core in the 'debug'
# partition on Expanse using the 'base' Anaconda3 software environment provided as part
# of Expanse's standard software modules.

/cm/shared/apps/sdsc/galyleo/galyleo.sh launch --account 'use300' --partition 'debug'  --cpus-per-task 1 --memory-per-node 1 --time-limit 00:30:00 --jupyter 'notebook' --notebook-dir "/expanse/lustre/projects/use300/${USER}" --env-modules 'cpu,gcc,anaconda3' --conda-env 'base' --quiet
```

```
# Example 2: Launch a JupyterLab session on a single GPU in the 'gpu-debug' partition
# on Expanse using the latest PyTorch Singularity container available.

galyleo.sh launch --account 'use300' --partition 'gpu-debug' --cpus-per-task 10 --memory-per-node 93 --gpus 1 --time-limit 00:30:00 --jupyter 'lab' --notebook-dir "/expanse/lustre/projects/use300/${USER}" --env-modules 'singularitypro' --sif '/cm/shared/apps/containers/singularity/pytorch/pytorch-gpu.sif' --bind '/expanse,/scratch' --nv --quiet
```

Running GPU notebooks using galyleo.sh



Step 1:
Login and setup user environment

Step 2:
Run command to launch secure notebook

Step 3:
Copy URL; paste into browser on local system

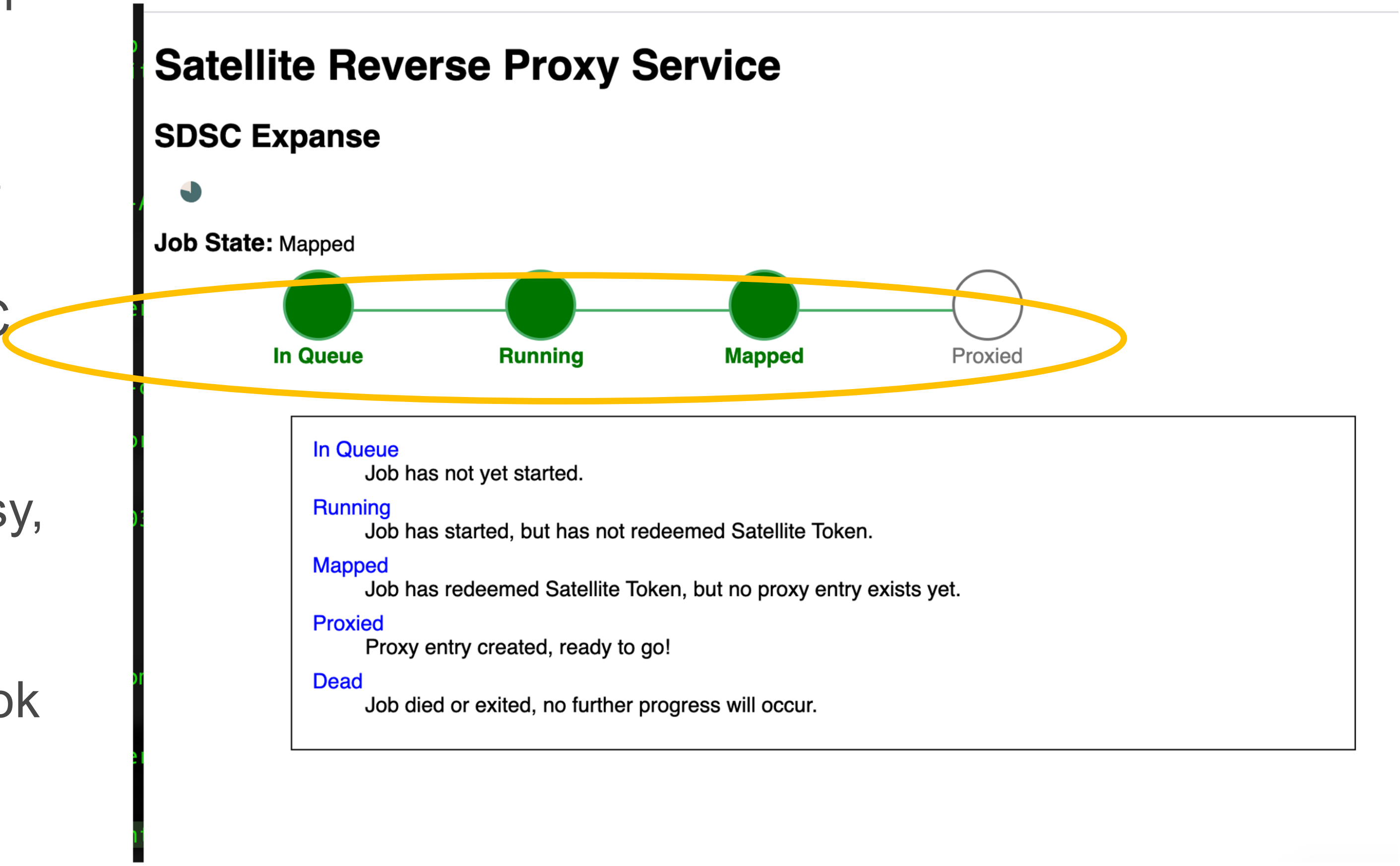
Step 4:
Monitor Slurm queue; wait for job to start

```
[mthomas@login01 ~]$ which galyleo.sh
/usr/bin/which: no galyleo.sh in (/cm/shared/apps/spack/cpu/opt/spack/linux-centos8-zen/gcc-8.3.1
[SNIP]
home/mthomas/.local/bin:/home/mthomas/bin)
[mthomas@login01 ~]$ export PATH="/cm/shared/apps/sdsc/galyleo:${PATH}"
[mthomas@login01 ~]$ which galyleo.sh
/cm/shared/apps/sdsc/galyleo/galyleo.sh
[mthomas@login01 ~]$ squeue -u mthomas
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
[mthomas@login01 ~]$ galyleo.sh launch --account 'use300' --partition 'gpu-debug' --cpus-per-task 10 --memory-per-node 93 --gpus 1 --time-limit 00:30:00 --jupyter 'lab' --notebook-dir "/expanse/lustre/projects/use300/${USER}" --env-modules 'singularitypro' --sif '/cm/shared/apps/containers/singularity/pytorch/pytorch-gpu.sif' --bind '/expanse,/scratch' --nv --quiet

https://cytoplasm-expansive-shack.expanse-user-content.sdsc.edu?token=ef077027b1d26dcc1c7d99d889a82b86

[mthomas@login01 ~]$ squeue -u mthomas
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
           3138746 gpu-debug galyleo-  mthomas PD       0:00      1 (None)
[mthomas@login01 ~]$ squeue -u mthomas
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
           3138746 gpu-debug galyleo-  mthomas  R       0:20      1 exp-7-59
```

- Paste the HTTPS URL into a web browser

Satellite Service: Monitor Notebook Status

- Load notebook URL in browser; wait for it to launch
- Monitor pending page
- Run the “squeue” command on the HPC system to check job status
- If the job queue is busy, it may take a while to launch the notebook
- Treat Jupyter Notebook URL as a password



Satellite Reverse Proxy Service

SDSC Expanse

Job State: Mapped

In Queue
Job has not yet started.

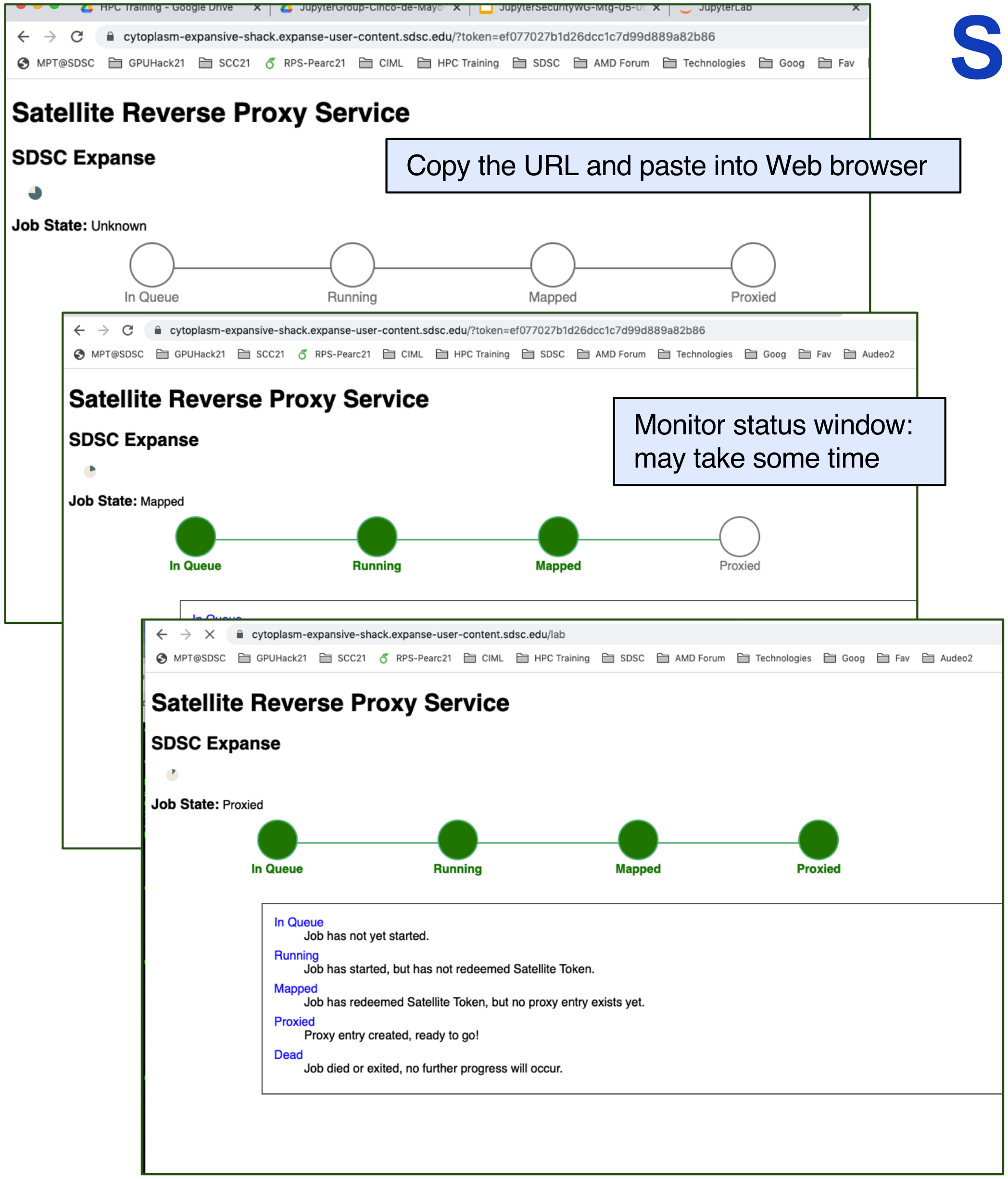
Running
Job has started, but has not redeemed Satellite Token.

Mapped
Job has redeemed Satellite Token, but no proxy entry exists yet.

Proxied
Proxy entry created, ready to go!

Dead
Job died or exited, no further progress will occur.

Satellite Pending Page



Copy the URL and paste into Web browser

Monitor status window: may take some time

cytoplasm-expansive-shack.expanse-user-content.sdsc.edu/?token=ef077027b1d26dcc1c7d99d889a82b86

Satellite Reverse Proxy Service

SDSC Expanse

Job State: Unknown

cytoplasm-expansive-shack.expanse-user-content.sdsc.edu/?token=ef077027b1d26dcc1c7d99d889a82b86

Satellite Reverse Proxy Service

SDSC Expanse

Job State: Mapped

cytoplasm-expansive-shack.expanse-user-content.sdsc.edu/lab

Satellite Reverse Proxy Service

SDSC Expanse

Job State: Proxied

In Queue
: Job has not yet started.

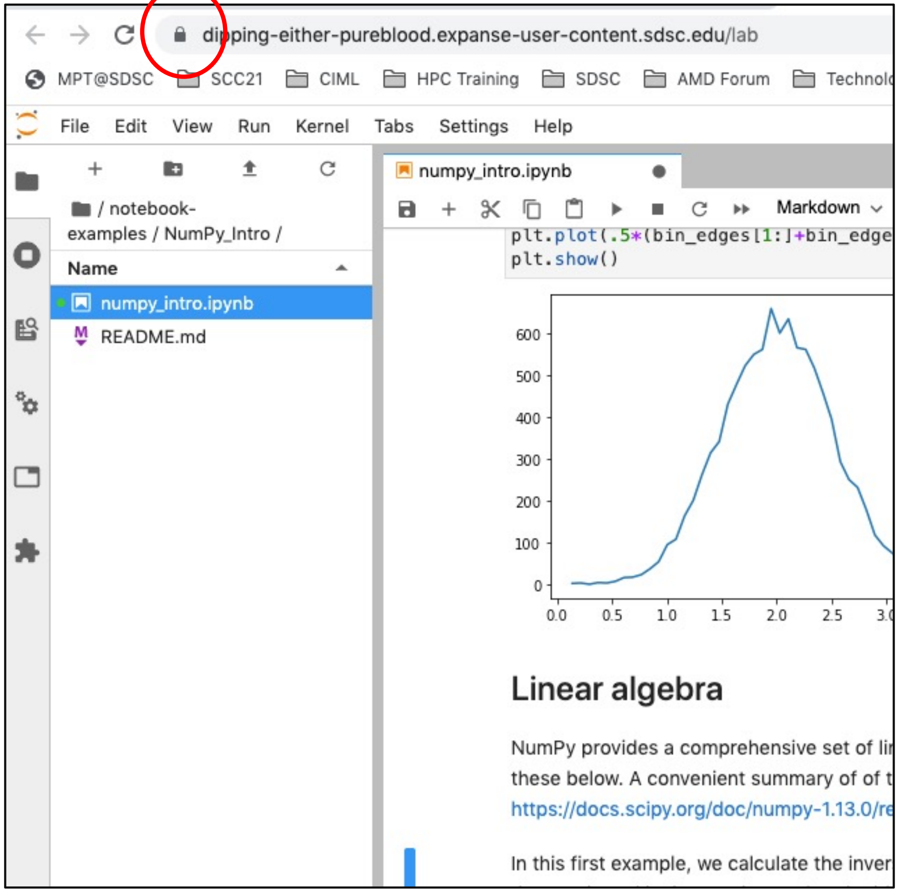
Running
: Job has started, but has not redeemed Satellite Token.

Mapped
: Job has redeemed Satellite Token, but no proxy entry exists yet.

Proxied
: Proxy entry created, ready to go!

Dead
: Job died or exited, no further progress will occur.

Access your secure Jupyter Service when compute node is allocated



dipping-either-pureblood.expanse-user-content.sdsc.edu/lab

```
plt.plot(.5*(bin_edges[1:]+bin_edge
plt.show()
```

Linear algebra

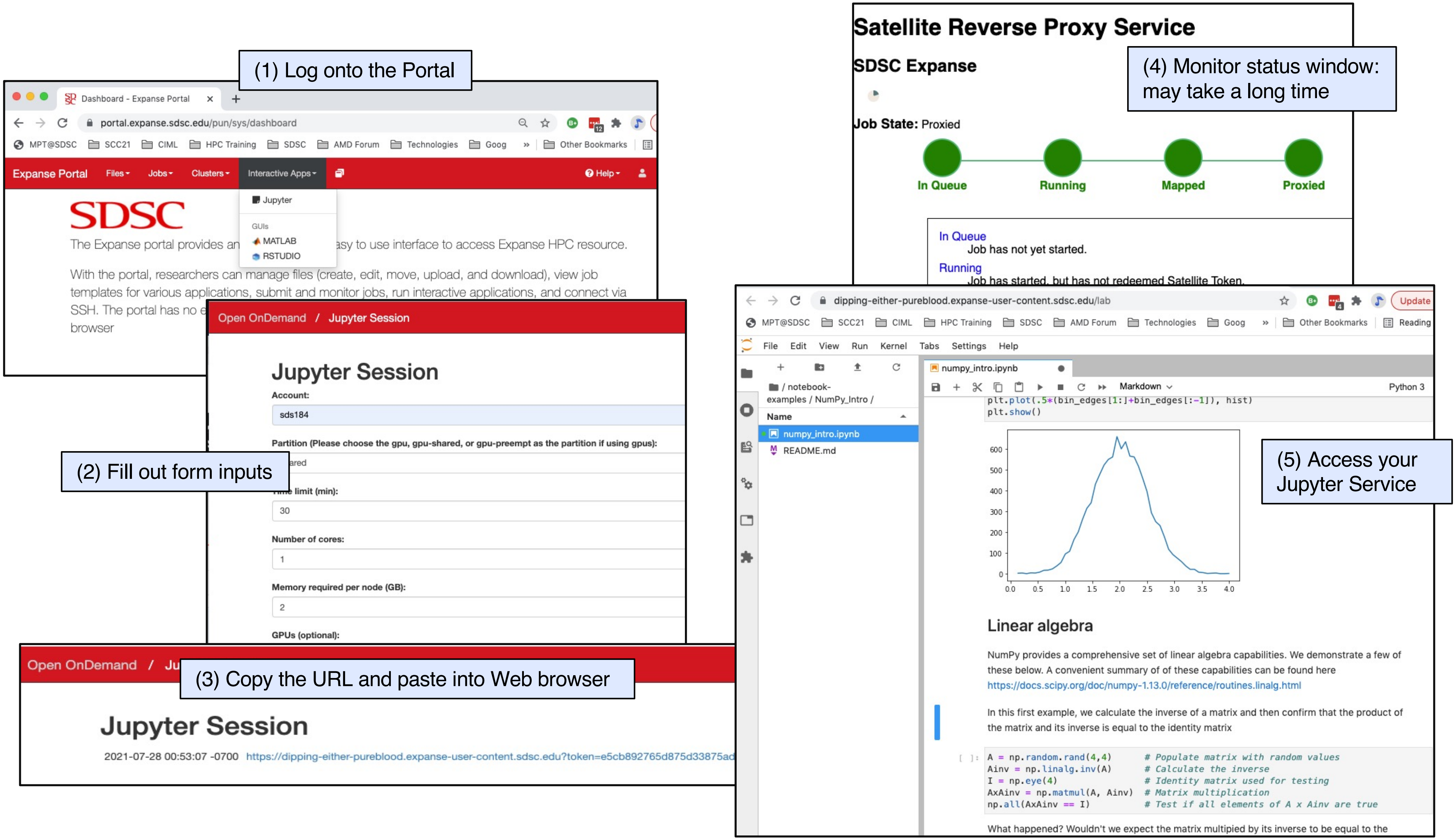
NumPy provides a comprehensive set of li

these below. A convenient summary of of t

https://docs.scipy.org/doc/numpy-1.13.0/re

In this first example, we calculate the inver

Running Notebooks from Expanse Portal



(1) Log onto the Portal

The Expanse portal provides an easy to use interface to access Expanse HPC resource.

With the portal, researchers can manage files (create, edit, move, upload, and download), view job templates for various applications, submit and monitor jobs, run interactive applications, and connect via SSH. The portal has no e browser

(2) Fill out form inputs

Jupyter Session

Account:	sds184
Partition (Please choose the gpu, gpu-shared, or gpu-preempt as the partition if using gpus):	ared
Time limit (min):	30
Number of cores:	1
Memory required per node (GB):	2
GPUs (optional):	

(3) Copy the URL and paste into Web browser

Jupyter Session

2021-07-28 00:53:07 -0700 https://dipping-either-pureblood.expanse-user-content.sdsc.edu?token=e5cb892765d875d33875ad

(4) Monitor status window: may take a long time

Satellite Reverse Proxy Service

SDSC Expanse

Job State: Proxied

In Queue
Job has not yet started.

Running
Job has started, but has not redeemed Satellite Token.

(5) Access your Jupyter Service

```
plt.plot(.5*(bin_edges[1:]+bin_edges[:-1]), hist)
plt.show()
```

Linear algebra

NumPy provides a comprehensive set of linear algebra capabilities. We demonstrate a few of these below. A convenient summary of of these capabilities can be found here https://docs.scipy.org/doc/numpy-1.13.0/reference/routines.linalg.html

In this first example, we calculate the inverse of a matrix and then confirm that the product of the matrix and its inverse is equal to the identity matrix

```
[ ]: A = np.random.rand(4,4)       # Populate matrix with random values
     Ainv = np.linalg.inv(A)       # Calculate the inverse
     I = np.eye(4)                 # Identity matrix used for testing
     AxAinv = np.matmul(A, Ainv)   # Matrix multiplication
     np.all(AxAinv == I)           # Test if all elements of A x Ainv are true
```

What happened? Wouldn't we expect the matrix multipied by its inverse to be equal to the

Satellite Client Example: start-jupyter

Running start-notebook

Make sure you have activated your conda environment

```
(base) [mthomas@login02 reverse-proxy]$ ./start-jupyter -A use300
Your notebook is here:
https://tightness-pulverize-starting.expanse-user-content.sdsc.edu?token=68a79db47927f40bc3831ce0c37f87a3
If you encounter any issues, please email help@xsede.org and mention the Reverse Proxy Service.
If you're not on Xsede, please email consult@sdsc.edu instead.
Your job id is 4543288
You may occasionally run the command 'squeue -j 4543288' to check the status of your job
(base) [mthomas@login02 reverse-proxy]$ squeue -j 4543288
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
           4543288   compute notebook  mthomas  R       0:50      1 exp-6-32
```

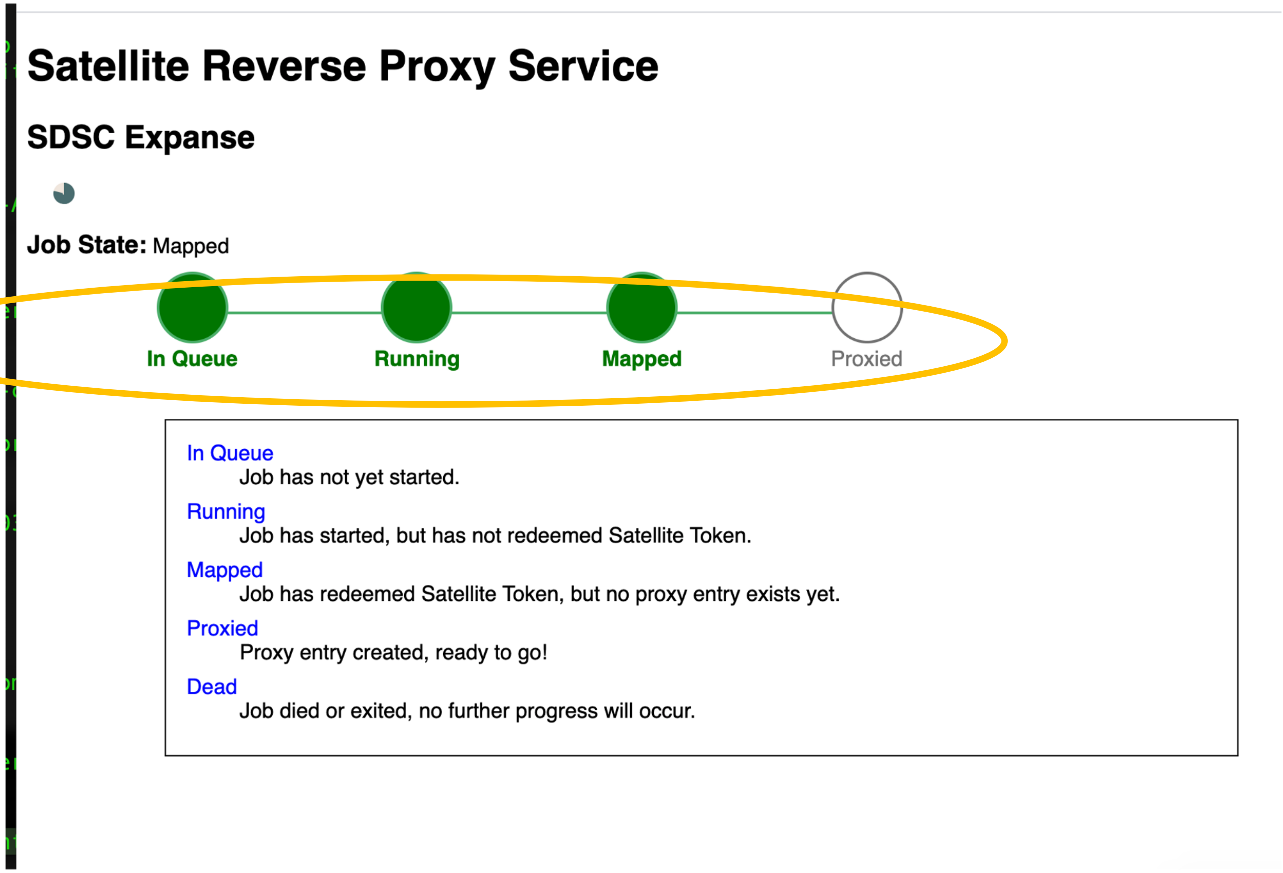
Once notebook is running, you can check the slurm output file

```
(base) [mthomas@login02 reverse-proxy]$ cat slurm-4543288.out
Image:
[I 01:52:43.924 NotebookApp] JupyterLab extension loaded from /cm/shared/apps/spack/gpu/opt/spack/linux-centos8-skylake_avx512/gcc-8.3.1/anaconda3-2020.11-bsn4npoxyw7jzz7fajncek3bvdoaa5wv/lib/python3.8/site-packages/jupyterlab
[I 01:52:43.925 NotebookApp] JupyterLab application directory is /cm/shared/apps/spack/gpu/opt/spack/linux-centos8-skylake_avx512/gcc-8.3.1/anaconda3-2020.11-bsn4npoxyw7jzz7fajncek3bvdoaa5wv/share/jupyter/lab
[I 01:52:43.926 NotebookApp] Serving notebooks from local directory: /home/mthomas
[I 01:52:43.926 NotebookApp] Jupyter Notebook 6.1.4 is running at:
[I 01:52:43.926 NotebookApp] http://exp-6-32.eth.cluster:8888/?token=...
[I 01:52:43.926 NotebookApp]  or http://127.0.0.1:8888/?token=...
[I 01:52:43.926 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100     9  100     9    0     0     53      0 --:--:-- --:--:-- --:--:--    52
Success!
[I 01:53:21.097 NotebookApp] 302 GET /?token=68a79db47927f40bc3831ce0c37f87a3 (10.21.0.30) 0.53ms
```

Paste the HTTPS URL into a web browser

Satellite Server Pending Page

- Load notebook URL in browser; wait for it to launch
- Monitor pending page
- Run the “squeue” command on the HPC system to check job status
- If the job queue is busy, it may take a while to launch the notebook
- Treat Jupyter Notebook URL as a password



Satellite Reverse Proxy Service

SDSC Expanse

Job State: Mapped

In Queue
    Job has not yet started.

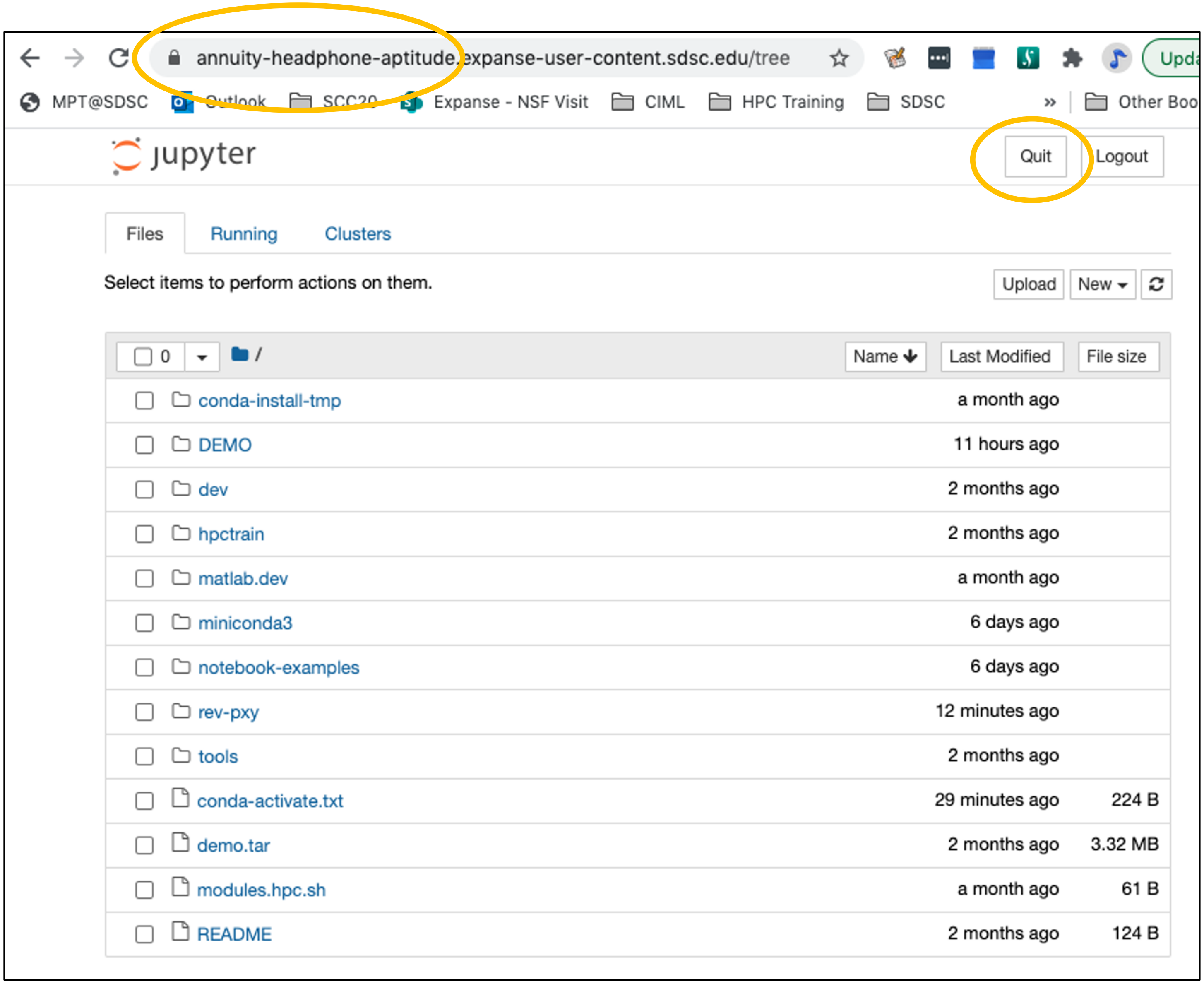
Running
    Job has started, but has not redeemed Satellite Token.

Mapped
    Job has redeemed Satellite Token, but no proxy entry exists yet.

Proxied
    Proxy entry created, ready to go!

Dead
    Job died or exited, no further progress will occur.

Your notebook is launched



Select items to perform actions on them.

Name	Last Modified	File size
□ conda-install-tmp	a month ago	
□ DEMO	11 hours ago	
□ dev	2 months ago	
□ hpctrain	2 months ago	
□ matlab.dev	a month ago	
□ miniconda3	6 days ago	
□ notebook-examples	6 days ago	
□ rev-pxy	12 minutes ago	
□ tools	2 months ago	
□ conda-activate.txt	29 minutes ago	224 B
□ demo.tar	2 months ago	3.32 MB
□ modules.hpc.sh	a month ago	61 B
□ README	2 months ago	124 B

When done with the notebook be sure to shut it down by quitting the notebook

Thank You

Resources

- Training Material for this presentation:
  - https://hpc-training.sdsc.edu/
  - https://github.com/sdsc-hpc-training-org/expanse-101
  - https://github.com/sdsc-hpc-training-org/notebooks-101
  - Interactive tutorial: https://education.sdsc.edu/training/interactive/202012_running_jupyter_notebooks_on_expanse/index.html
- Expanse User Guide
  - https://www.sdsc.edu/support/user_guides/expanse.html
- SDSC Training Resources
  - https://www.sdsc.edu/education_and_training/training
  - https://github.com/sdsc-hpc-training/webinars
- XSEDE Training Resources
  - https://www.xsede.org/for-users/training
  - https://cvw.cac.cornell.edu/expanse/