SDSC Summer Institute 2020: 5.3 Jupyter Notebooks, Reverse Proxy Server

Mary Thomas, Computational Scientist, SDSC

Outline

- Getting Started; Comet Overview
- What are Jupyter Notebooks?
- Security concerns
    - HTTP vs HTTPS
    - SSH vs SSH tunneling (HTTP)
- Software Requirements for Running Notebooks on Comet
    - Install conda, conda environments
- Methods for Running Notebooks on Comet
    - Running notebooks on the Login node or interactive node
- SDSC Reverse Proxy Service (HTTPS)
- Live Demo

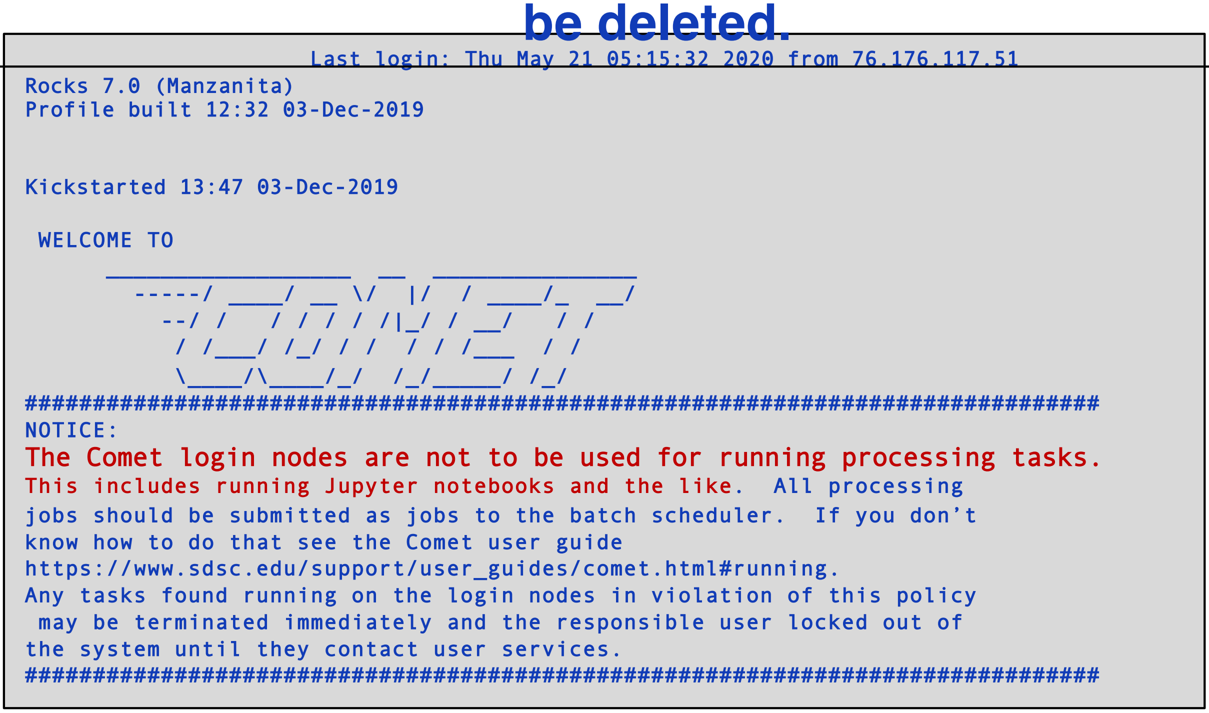
- Key Goal: Learn how to run Jupyter Notebooks securely.

Basic Information

- This webinar location:
  - https://github.com/sdsc-hpc-training-org/notebooks-101
- Online repo for companion tutorial/webinar information:
  - https://github.com/sdsc-hpc-training-org/notebook_examples
  - https://github.com/sdsc-training-org/webinars
  - Access to the Jupyter Reverse Proxy Server:
  - https://github.com/sdsc-hpc-training-org/reverse-proxy
- Other training events and links to past events listed at SDSC:
  - https://www.sdsc.edu/education_and_training/training.html
- You must be familiar with running basic Unix commands, connecting to Comet via SSH, running notebooks, and other basic skills. Check out our basic skills repo:
    - https://github.com/sdsc-hpc-training-org/basic_skills
  - You must have a comet account in order to access the system. To obtain a trial account:
  - http://www.sdsc.edu/support/user_guides/comet.html#trial_accounts
- Comet User Guide:
  - https://www.sdsc.edu/support/user_guides/comet.html

REMINDER!!!!

Jupyter Notebooks should not be run on the login nodes. Those jobs will be deleted.



```
                     Last login: Thu May 21 05:15:32 2020 from 76.176.117.51
Rocks 7.0 (Manzanita)
Profile built 12:32 03-Dec-2019


Kickstarted 13:47 03-Dec-2019

 WELCOME TO
      ___________________  __  ________________
        -----/ ____/ __ \/  |/  / ____/_  __/
           --/ /   / / / / /|_/ / __/   / /
            / /___/ /_/ / /  / / /___  / /
            \____/\____/_/  /_/_____/ /_/
####################################################################################
NOTICE:
The Comet login nodes are not to be used for running processing tasks.
This includes running Jupyter notebooks and the like.  All processing
jobs should be submitted as jobs to the batch scheduler.  If you don't
know how to do that see the Comet user guide
https://www.sdsc.edu/support/user_guides/comet.html#running.
Any tasks found running on the login nodes in violation of this policy
 may be terminated immediately and the responsible user locked out of
the system until they contact user services.
####################################################################################
```

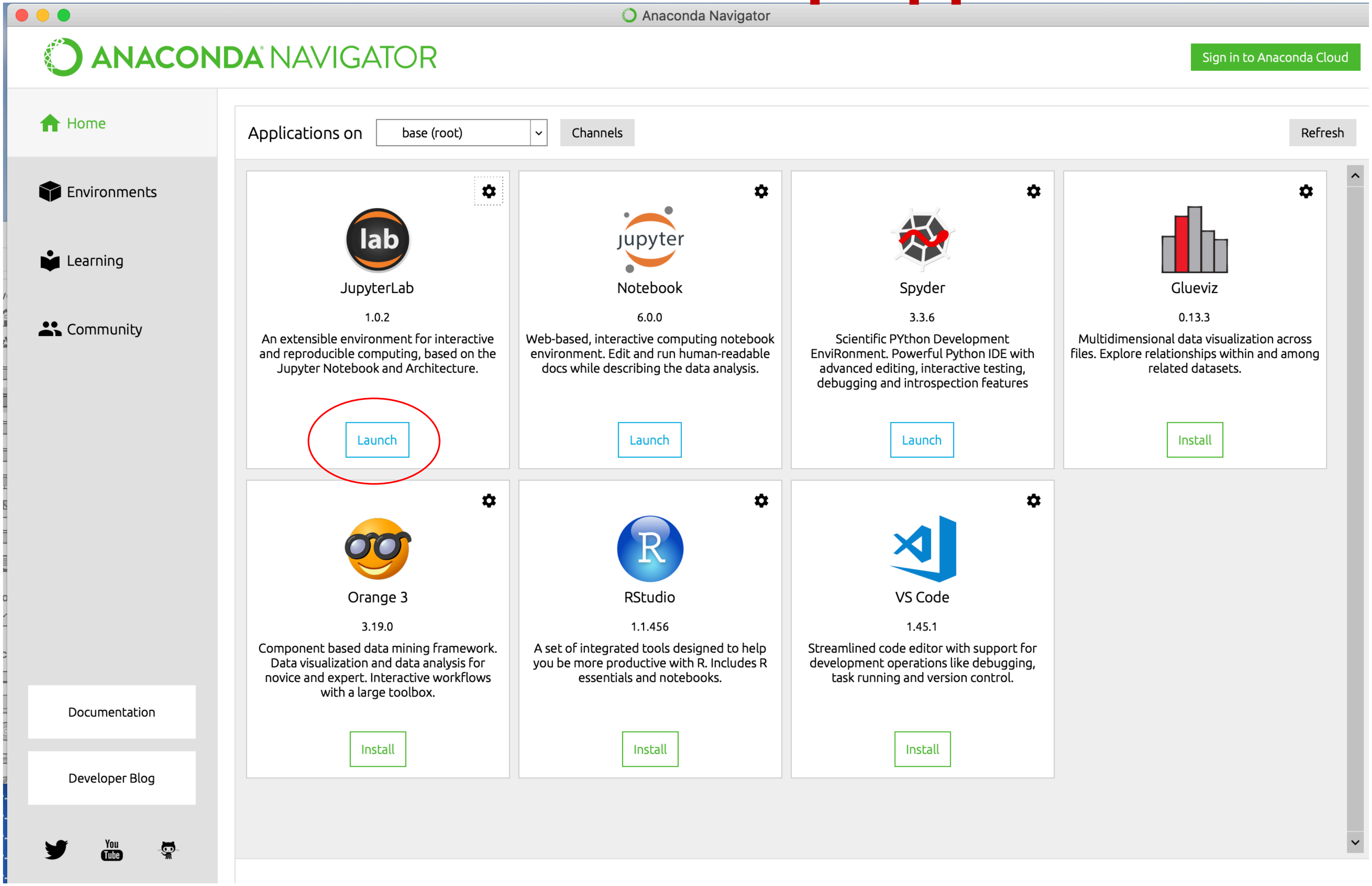
Obtaining Notebook Examples

```
(base) [username@comet-ln3:~] git clone https://github.com/sdsc-hpc-training-org/notebook_examples.git
Cloning into 'notebook_examples'...
remote: Enumerating objects: 55, done.
remote: Counting objects: 100% (55/55), done.
remote: Compressing objects: 100% (44/44), done.
remote: Total 55 (delta 6), reused 55 (delta 6), pack-reused 0
Unpacking objects: 100% (55/55), done.
(base) [username@comet-ln3:~] cd notebook_examples/
(base) [username@comet-ln3:~/notebook_examples] ll
total 609
drwxr-xr-x  7 username use300      9 May 20 12:38 .
drwxr-x--- 58 username use300     89 May 20 12:38 ..
drwxr-xr-x  3 username use300      8 May 20 12:38 Boring_Python
drwxr-xr-x  4 username use300      4 May 20 12:38 cuda
drwxr-xr-x  2 username use300      4 May 20 12:38 deep_learning
drwxr-xr-x  8 username use300     13 May 20 12:38 .git
-rw-r--r--  1 username use300 432678 May 20 12:38 gnuplot.ipynb
drwxr-xr-x  2 mthomas use300       6 May 21 07:34 hello-world
drwxr-xr-x  8 username use300   1060 May 20 12:45 hello_world.ipynb
drwxr-xr-x  2 username use300     10 May 20 12:38 Pandas
-rw-r--r--  1 username use300    322 May 20 12:38 README.md
(base) [username@comet-ln3:~/notebook_examples]
```

Software Requirements for Running Notebooks on Comet

https://comet-notebooks-101.readthedocs.io/en/latest/prerequisites.html

Anaconda: desktop application



Anaconda Navigator

ANACONDA NAVIGATOR

Sign in to Anaconda Cloud

- Home
- Environments
- Learning
- Community

Documentation

Developer Blog

Applications on base (root) Channels Refresh

JupyterLab
1.0.2
An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.
Launch

Notebook
6.0.0
Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.
Launch

Spyder
3.3.6
Scientific PYthon Development EnviRonment. Powerful Python IDE with advanced editing, interactive testing, debugging and introspection features
Launch

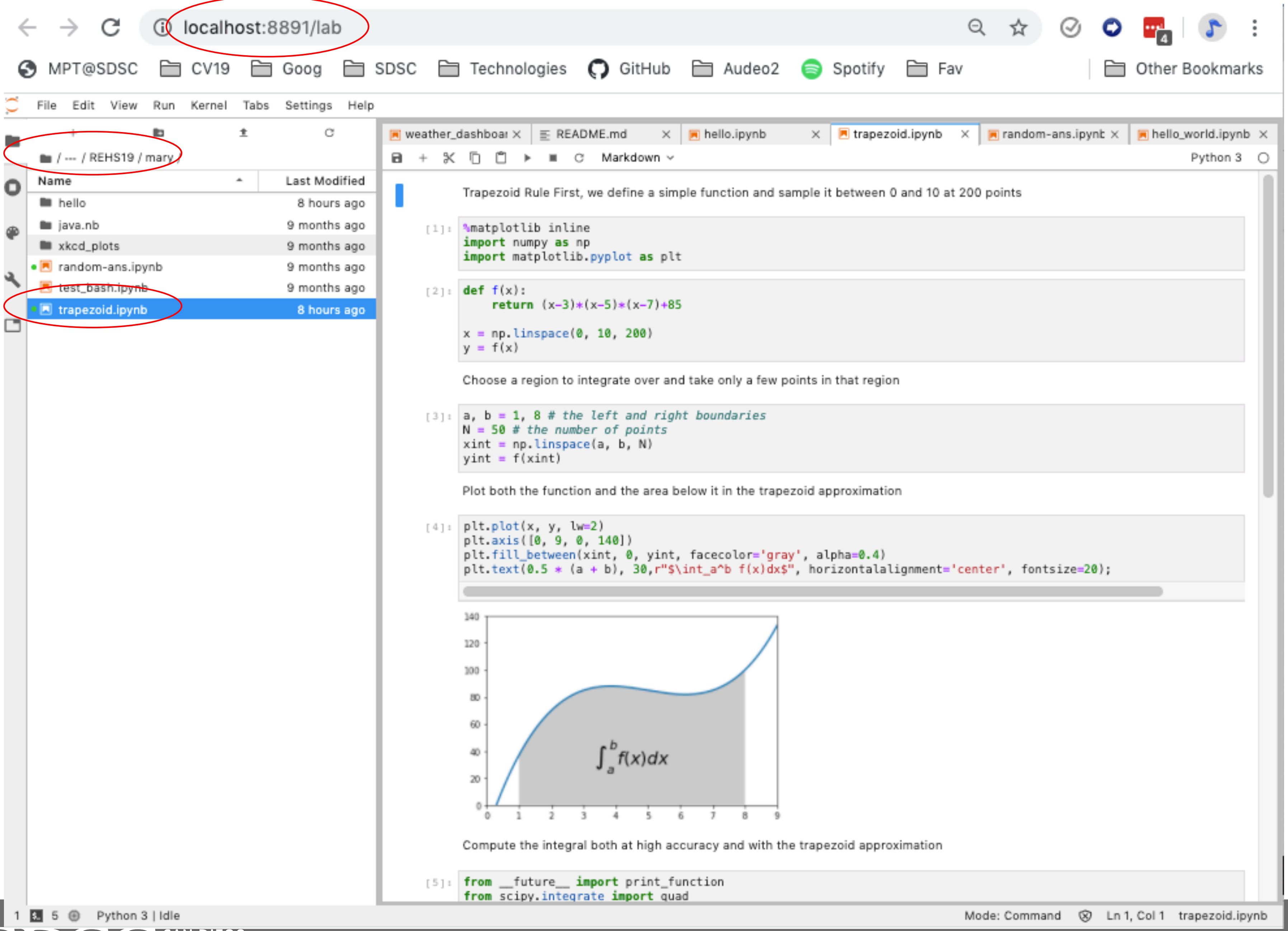
Glueviz
0.13.3
Multidimensional data visualization across files. Explore relationships within and among related datasets.
Install

Orange 3
3.19.0
Component based data mining framework. Data visualization and data analysis for novice and expert. Interactive workflows with a large toolbox.
Install

RStudio
1.1.456
A set of integrated tools designed to help you be more productive with R. Includes R essentials and notebooks.
Install

VS Code
1.45.1
Streamlined code editor with support for development operations like debugging, task running and version control.
Install

OS X – Launch Apps with click of a Button



/ ··· / REHS19 / mary /

Name	Last Modified
hello	8 hours ago
java.nb	9 months ago
xkcd_plots	9 months ago
random-ans.ipynb	9 months ago
test_bash.ipynb	9 months ago
trapezoid.ipynb	8 hours ago

Trapezoid Rule First, we define a simple function and sample it between 0 and 10 at 200 points

```
[1]: %matplotlib inline
     import numpy as np
     import matplotlib.pyplot as plt
```

```
[2]: def f(x):
         return (x-3)*(x-5)*(x-7)+85

     x = np.linspace(0, 10, 200)
     y = f(x)
```

Choose a region to integrate over and take only a few points in that region

```
[3]: a, b = 1, 8 # the left and right boundaries
     N = 50 # the number of points
     xint = np.linspace(a, b, N)
     yint = f(xint)
```

Plot both the function and the area below it in the trapezoid approximation

```
[4]: plt.plot(x, y, lw=2)
     plt.axis([0, 9, 0, 140])
     plt.fill_between(xint, 0, yint, facecolor='gray', alpha=0.4)
     plt.text(0.5 * (a + b), 30,r"$\int_a^b f(x)dx$", horizontalalignment='center', fontsize=20);
```

Compute the integral both at high accuracy and with the trapezoid approximation

```
[5]: from __future__ import print_function
     from scipy.integrate import quad
```

```
1  5  Python 3 | Idle                Mode: Command   Ln 1, Col 1   trapezoid.ipynb
```

Software Requirements on HPC Systems

- Not so easy to run notebooks on HPC system/Unix
- Important and convenient to have customized, virtual Python environments,
  - install packages that aren't installed with the system's Python installation
  - You need different sets of Python packages for different purposes.
- We recommend that you setup your own local environment:
  - This gives you control over libraries used by your notebooks
  - You can install either Anaconda or just conda
    - Anaconda includes the conda command (which can be used to create, use, and manage virtual Python environments).
    - Use system Python
- Optionally: use singularity
  - Install locally using anaconda/etc.
  - Advantage of using containers: everything is built for you to use
  - Disadvantage: not easy to modify

Conda

- https://docs.conda.io/projects/conda/en/latest/
- Conda is an open-source package management system and environment management system (like pip)
- Created for Python programs
  - can package and distribute software for any language.
- Conda Cheat Sheet:
  - https://kapeli.com/cheat_sheets/Conda.docset/Contents/Resources/Documents/index

Create a virtual environment

- Use conda to create a virtual environment
  - Choose whatever name you want
  - $ conda create --name example_env
- To see which virtual environments you’ve created:
  - $ conda env list
- To use a particular virtual environment (e.g., one named ‘example_env’):
- $ source activate example_env # Note: don’t use ‘conda activate’
- Install JupyterLab and JupyterNotebooks

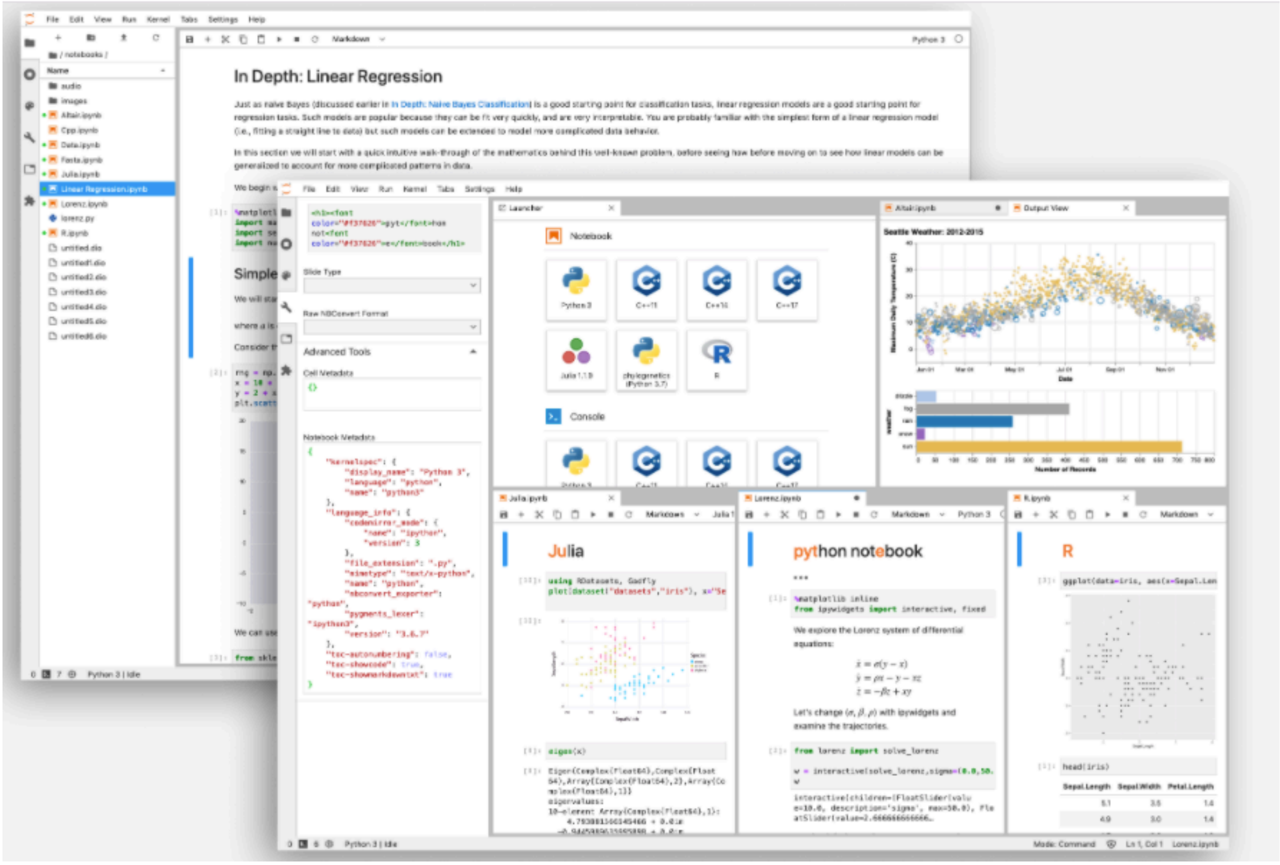
A caveat about file systems

- Be aware of where you launch your notebook service:
- Login nodes and the nodes that run batch scripts have access to the user’s home directory, but the compute nodes do not.
- The home directory is where the files that make up the virtual environment are stored by default.
- So if you want to use the virtual environment from a batch script, it either has to run on the batch node (e.g., don’t try to run it via a jsrun command) or you will have to figure out how to force conda to store virtual environments in your $MEMBERWORK directory.
- If you launch the notebook from your home dir, you will not be able to run notebooks from your projects directory

Overview of Jupyter Notebooks

What are Jupyter Notebooks?

- Why do we use them?



In Depth: Linear Regression

Just as naive Bayes (discussed earlier in In Depth: Naive Bayes Classification) is a good starting point for classification tasks, linear regression models are a good starting point for regression tasks. Such models are popular because they can be fit very quickly, and are very interpretable. You are probably familiar with the simplest form of a linear regression model (i.e., fitting a straight line to data) but such models can be extended to model more complicated data behavior.

In this section we will start with a quick intuitive walk-through of the mathematics behind this well-known problem, before seeing how before moving on to see how linear models can be generalized to account for more complicated patterns in data.

https://jupyter.org/

Overview of Jupyter Notebooks

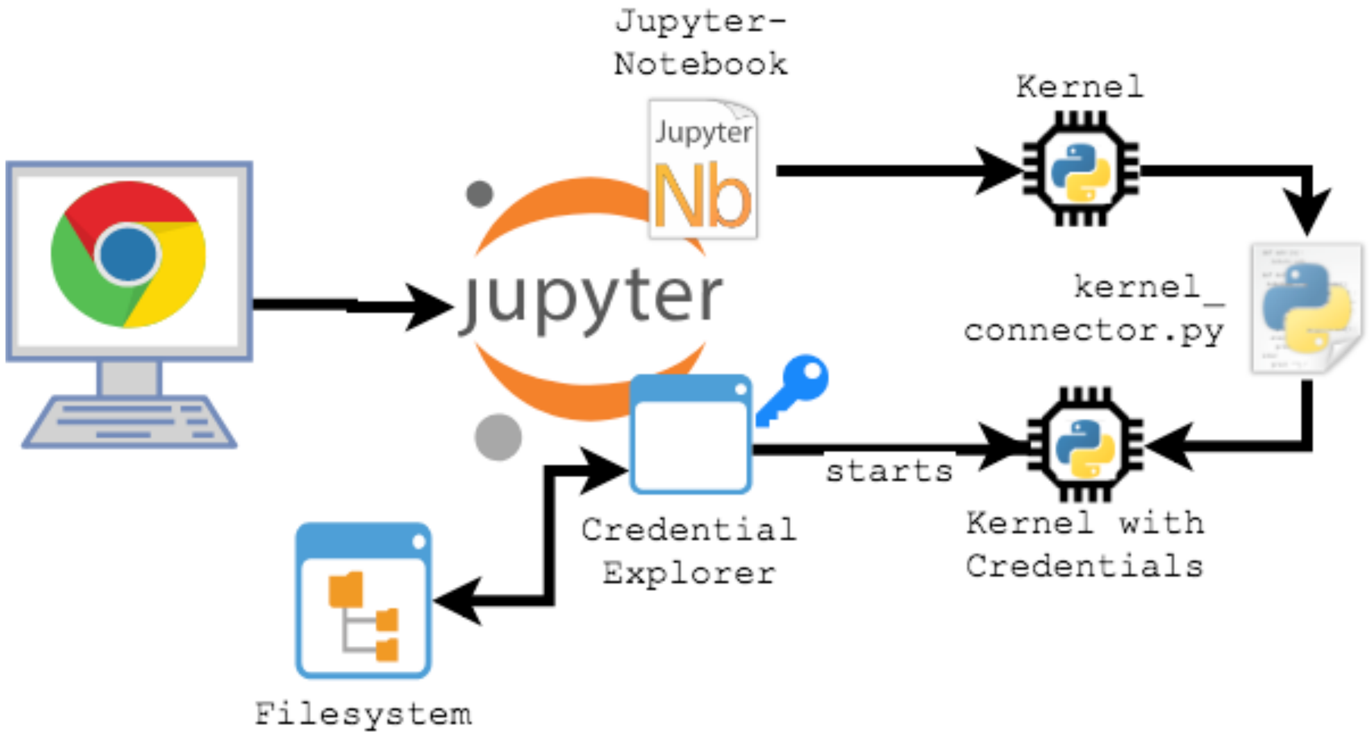
- Community of open-source developers, scientists, educators, and data scientists.
- Goal: build open-source tools and create community that facilitates scientific research, reproducible and open workflows, education, computational narratives, and data analytics.
- Jupyter supports over 100 programming languages, and connects data analytics tools across a range of disciplines and communities.

Source: **https://bids.berkeley.edu/research/project-jupyter**

Jupyter Notebooks

- Web-based interactive computing platform
- Allows users to author computational apps
- code, equations, narrative text, interactive user interfaces, and other rich media.

- Enables collaborative creation of notebooks
- Can be used across a wide range of discipines



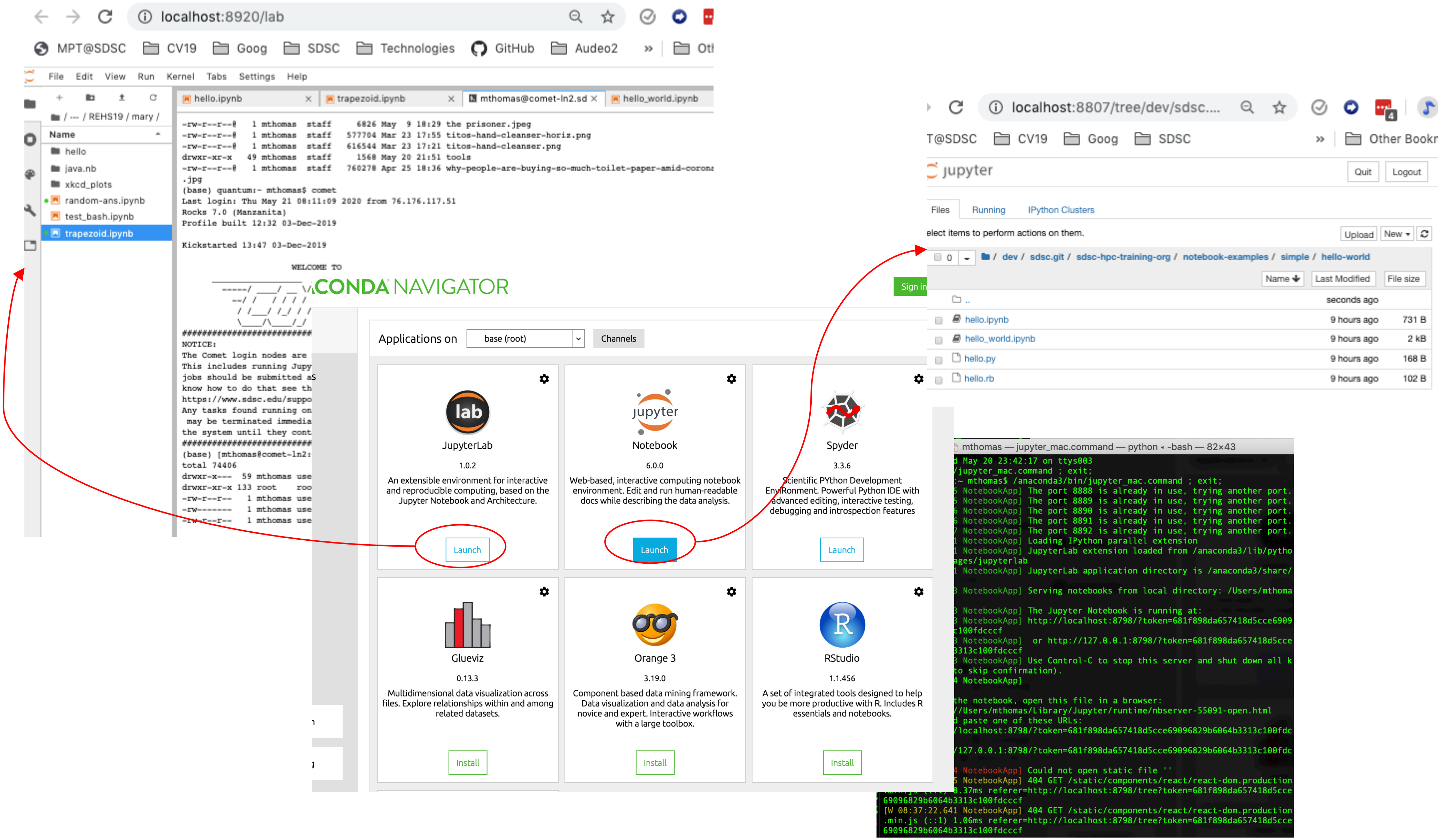
https://towardsdatascience.com/the-jupyterlab-credential-store-9cc3a0b9356

JupyterLab

- Jupyter’s next-generation interface, JupyterLab facilitates data scientists to compose the interface that suits their needs.
- Flexible, extensible user interface - supports diversity of workflows in data science.
- Runs using same Jupyter server as Notebook interface → allows it to be accessed remotely on shared infrastructure (for example, via a JupyterHub)

Source: https://bids.berkeley.edu/research/project-jupyter

Jupyter Env - Desktop



```
-rw-r--r--@   1 mthomas  staff     6826 May  9 18:29 the prisoner.jpeg
-rw-r--r--@   1 mthomas  staff   577704 Mar 23 17:55 titos-hand-cleanser-horiz.png
-rw-r--r--@   1 mthomas  staff   616544 Mar 23 17:21 titos-hand-cleanser.png
drwxr-xr-x   49 mthomas  staff     1568 May 20 21:51 tools
-rw-r--r--@   1 mthomas  staff   760278 Apr 25 18:36 why-people-are-buying-so-much-toilet-paper-amid-corona
.jpg
(base) quantum:~ mthomas$ comet
Last login: Thu May 21 08:11:09 2020 from 76.176.117.51
Rocks 7.0 (Manzanita)
Profile built 12:32 03-Dec-2019

Kickstarted 13:47 03-Dec-2019

                                WELCOME TO
      __________________
       -----/ ____/ __ \
          --/ /    / / /
           / /___/ /_/ /
           \____/\____/
###########################
NOTICE:
The Comet login nodes are
This includes running Jupy
jobs should be submitted as
know how to do that see th
https://www.sdsc.edu/suppo
Any tasks found running on
 may be terminated immedia
the system until they cont
###########################
(base) [mthomas@comet-ln2:
total 74406
drwxr-x---   59 mthomas use
drwxr-xr-x  133 root    roo
-rw-r--r--    1 mthomas use
-rw-------    1 mthomas use
-rw-r--r--    1 mthomas use
```

JupyterLab
1.0.2
An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.
Launch

Notebook
6.0.0
Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.
Launch

Spyder
3.3.6
Scientific PYthon Development EnviRonment. Powerful Python IDE with advanced editing, interactive testing, debugging and introspection features
Launch

Glueviz
0.13.3
Multidimensional data visualization across files. Explore relationships within and among related datasets.
Install

Orange 3
3.19.0
Component based data mining framework. Data visualization and data analysis for novice and expert. Interactive workflows with a large toolbox.
Install

RStudio
1.1.456
A set of integrated tools designed to help you be more productive with R. Includes R essentials and notebooks.
Install

elect items to perform actions on them.

/ dev / sdsc.git / sdsc-hpc-training-org / notebook-examples / simple / hello-world

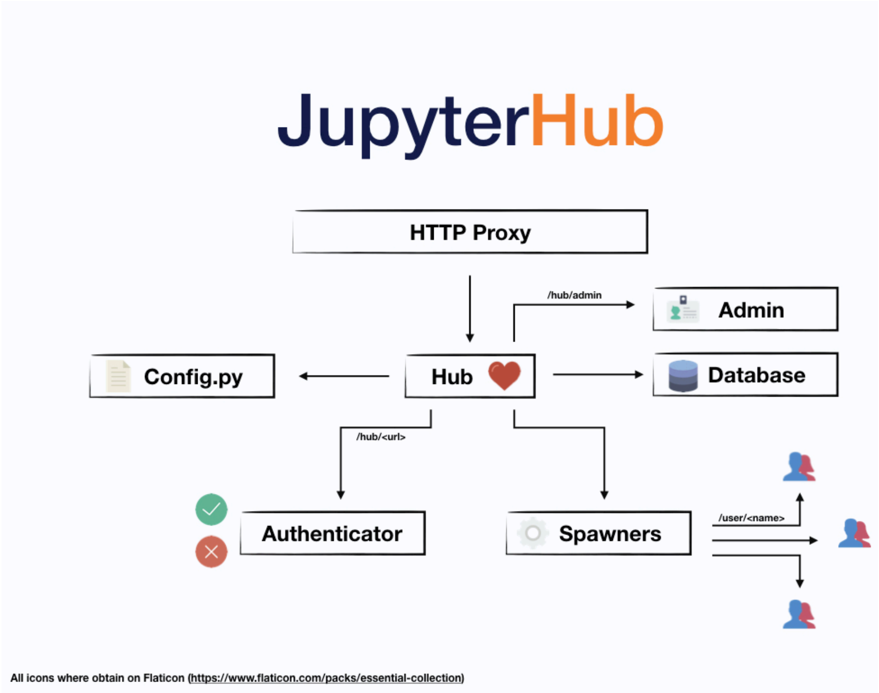
Name	Last Modified	File size
..	seconds ago	
hello.ipynb	9 hours ago	731 B
hello_world.ipynb	9 hours ago	2 kB
hello.py	9 hours ago	168 B
hello.rb	9 hours ago	102 B

mthomas — jupyter_mac.command — python ◂ -bash — 82×43

```
d May 20 23:42:17 on ttys003
/jupyter_mac.command ; exit;
:~ mthomas$ /anaconda3/bin/jupyter_mac.command ; exit;
5 NotebookApp] The port 8888 is already in use, trying another port.
5 NotebookApp] The port 8889 is already in use, trying another port.
6 NotebookApp] The port 8890 is already in use, trying another port.
6 NotebookApp] The port 8891 is already in use, trying another port.
7 NotebookApp] The port 8892 is already in use, trying another port.
1 NotebookApp] Loading IPython parallel extension
1 NotebookApp] JupyterLab extension loaded from /anaconda3/lib/pytho
ages/jupyterlab
1 NotebookApp] JupyterLab application directory is /anaconda3/share/

3 NotebookApp] Serving notebooks from local directory: /Users/mthoma

3 NotebookApp] The Jupyter Notebook is running at:
3 NotebookApp] http://localhost:8798/?token=681f898da657418d5cce6909
c100fdcccf
3 NotebookApp]  or http://127.0.0.1:8798/?token=681f898da657418d5cce
3313c100fdcccf
3 NotebookApp] Use Control-C to stop this server and shut down all k
to skip confirmation).
4 NotebookApp]

the notebook, open this file in a browser:
//Users/mthomas/Library/Jupyter/runtime/nbserver-55091-open.html
d paste one of these URLs:
/localhost:8798/?token=681f898da657418d5cce69096829b6064b3313c100fdc

/127.0.0.1:8798/?token=681f898da657418d5cce69096829b6064b3313c100fdc

4 NotebookApp] Could not open static file ''
5 NotebookApp] 404 GET /static/components/react/react-dom.production
8.37ms referer=http://localhost:8798/tree?token=681f898da657418d5cce
69096829b6064b3313c100fdcccf
[W 08:37:22.641 NotebookApp] 404 GET /static/components/react/react-dom.production
.min.js (::1) 1.06ms referer=http://localhost:8798/tree?token=681f898da657418d5cce
69096829b6064b3313c100fdcccf
```

JupyterHub

- Provides remote access to Jupyter servers via Web browser.
- Make high-powered computational environments and resources more accessible to students, researchers, and collaborators.
- Runs in the cloud or on your own hardware
- Makes it possible to serve a pre-configured data science environment to any user in the world.
- Used in education and large-scale courses as well as in collaborative and massively-open data analytics projects.



All icons where obtain on Flaticon (https://www.flaticon.com/packs/essential-collection)

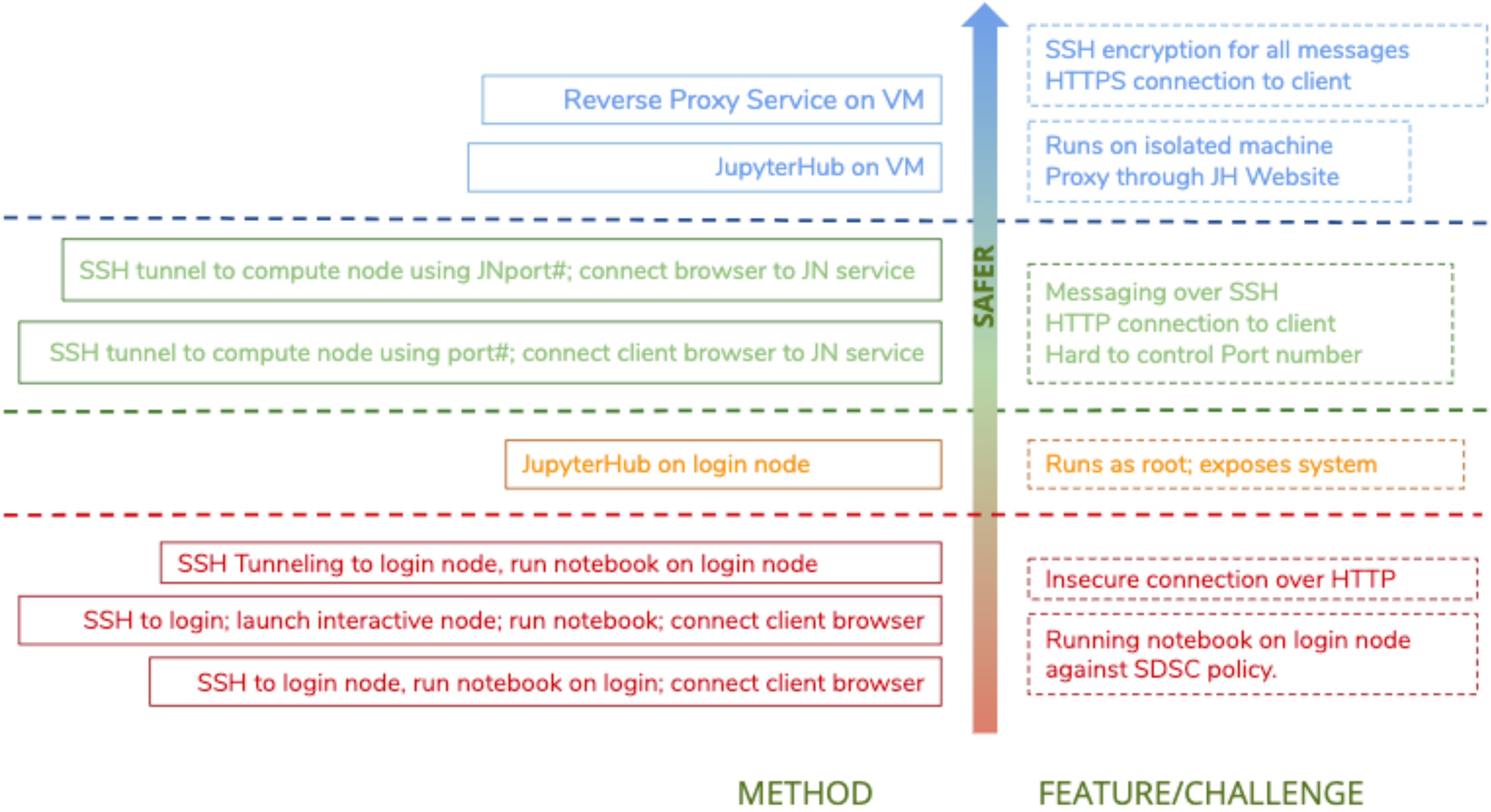
https://jupyterhub.readthedocs.io/en/stable/

Jupyter Notebook Security

Not All Methods are Secure

- Notebooks on Comet/Level of security
- Security concerns
  - HTTP vs HTTPS
  - SSH vs SSH tunneling (HTTP)
- Most insecure method: HTTP (public IP)
  - Next levels of security: tunneling
  - Mention Jupyter Hub - somewhat more secure (out of the box)???
  - More secure - Reverse Proxy

Methods for Running Notebooks



Reverse Proxy Service on VM

JupyterHub on VM

SSH encryption for all messages
HTTPS connection to client

Runs on isolated machine
Proxy through JH Website

SSH tunnel to compute node using JNport#; connect browser to JN service

SSH tunnel to compute node using port#; connect client browser to JN service

Messaging over SSH
HTTP connection to client
Hard to control Port number

JupyterHub on login node

Runs as root; exposes system

SSH Tunneling to login node, run notebook on login node

SSH to login; launch interactive node; run notebook; connect client browser

SSH to login node, run notebook on login; connect client browser

Insecure connection over HTTP

Running notebook on login node
against SDSC policy.

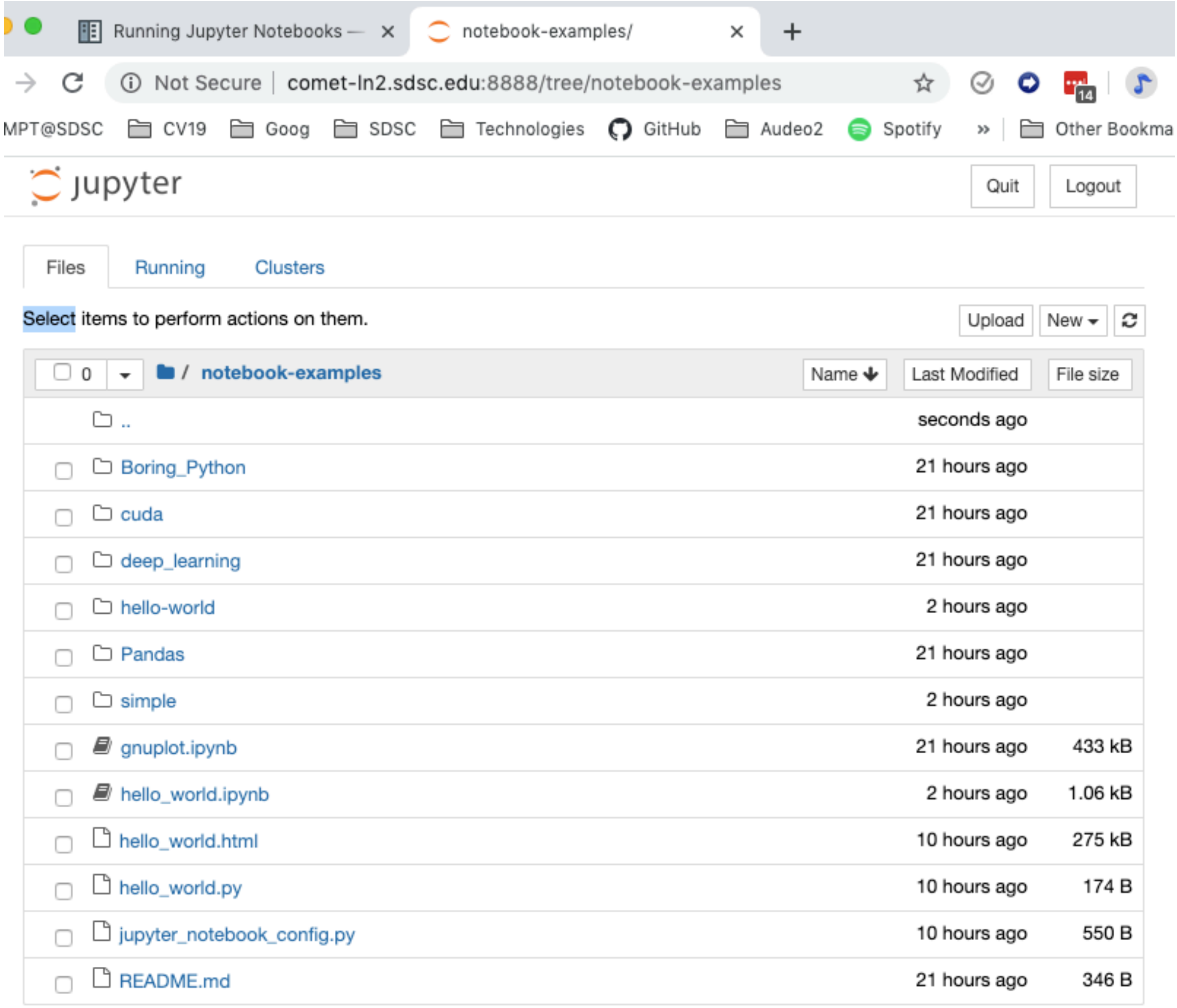
METHOD

FEATURE/CHALLENGE

Key Vulnerability:
Notebooks 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)
  - 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 unsecure connections your account is vulnerable to hacking



Select items to perform actions on them.

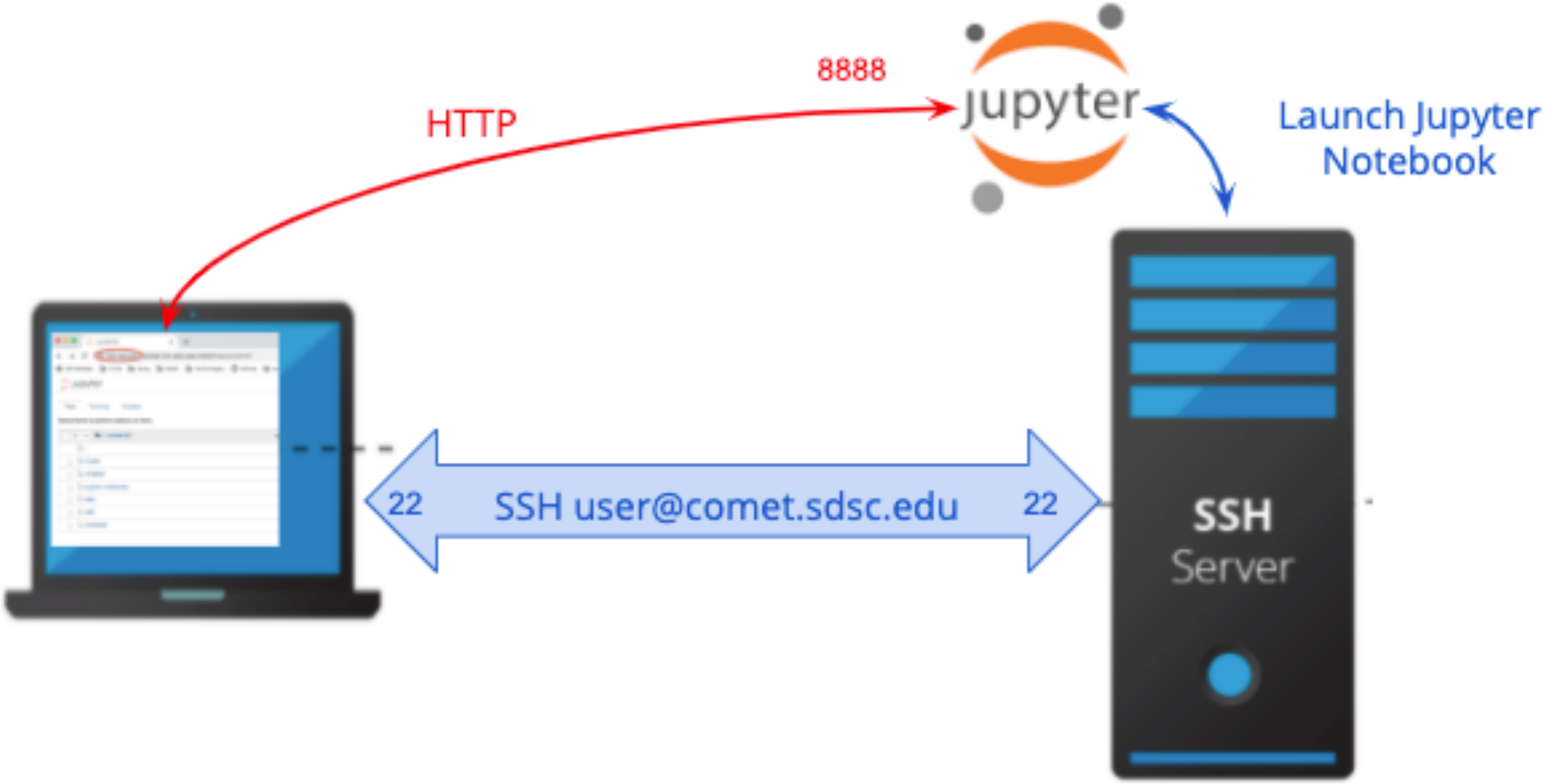
/ notebook-examples

Name	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

Methods for Running Notebooks on Comet

- Connection scenarios:
  - Connection to Notebook over HTTP (very insecure)
  - Connection to Notebook over SSH tunneling (secure)
  - Connection to Notebook over HTTPS using the Jupyter Reverse Proxy Service (very secure)
  - Coming Soon: Galyleo remote notebook launcher
- Notebooks can be run on the following nodes:
  - Login node
  - Interactive node
  - Compute node
  - GPU node

Why Connection over HTTP (unsecure)



Improve Security: SSH Tunneling

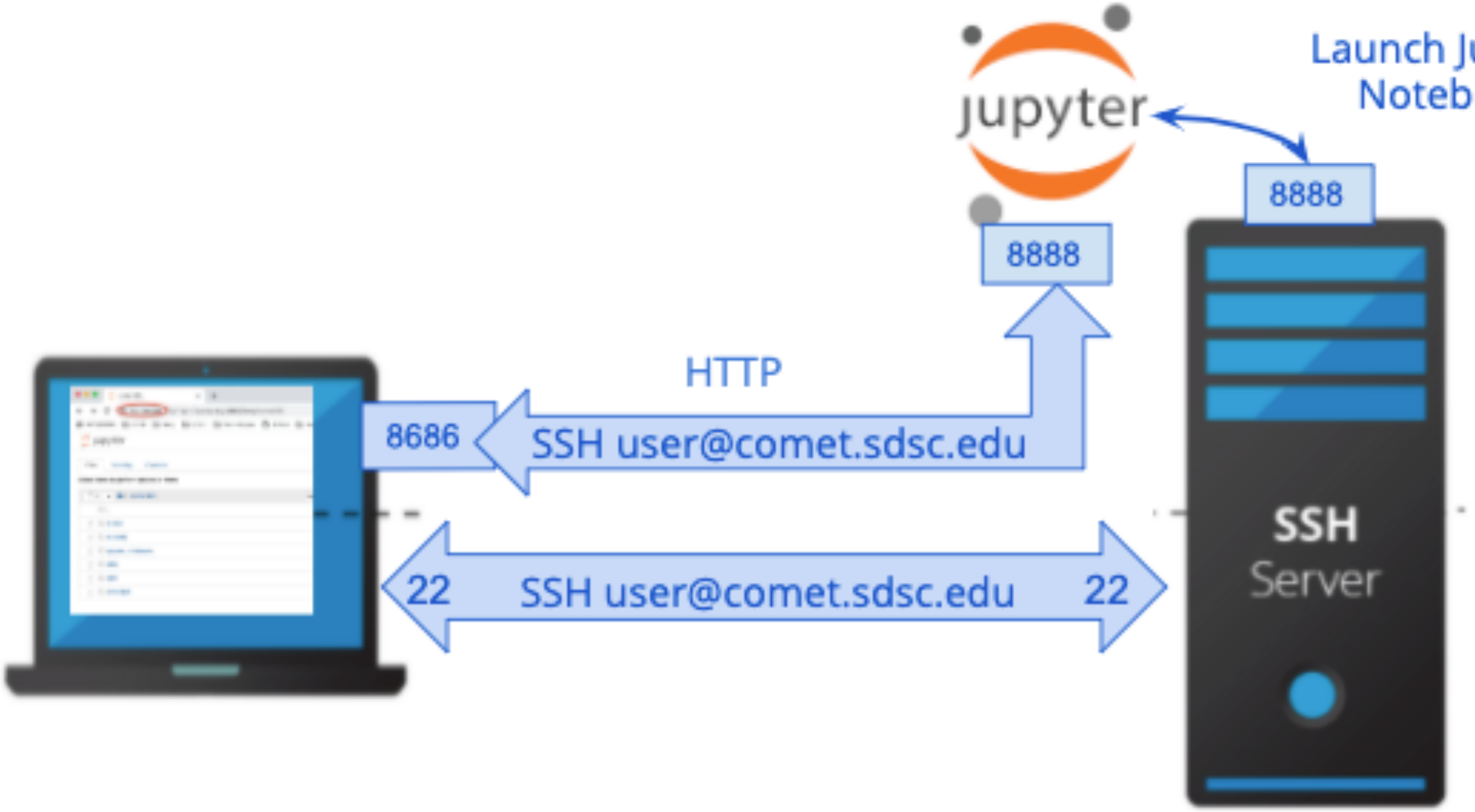
See: https://comet-notebooks-101.readthedocs.io/en/latest/methods/tunneling.html

- Port forwarding via **SSH tunneling** creates a secure connection between a local computer and a remote machine through which services can be relayed.
- Connections are encrypted
- Useful for transmitting information that uses an unencrypted protocol (IMAP, VNC, HTTP server).
- 3 Types:
  - **Local port forwarding** **(will use for notebook servers)**: connections *from SSH client* are forwarded *via the SSH server*, then *to a destination server.*
  - **Remote port forwarding**: connections *from the SSH server* are forwarded *via the SSH client*, then *to a destination server*
  - **Dynamic port forwarding**: connections from *programs* forwarded *via the SSH client*, then *via the SSH server*, and finally *to destination servers.*

Source: https://help.ubuntu.com/community/SSH/OpenSSH/PortForwarding

Secure Connection over SSH Tunneling

Uses Local Port Forwarding to connect to a Jupyter Notebook Server



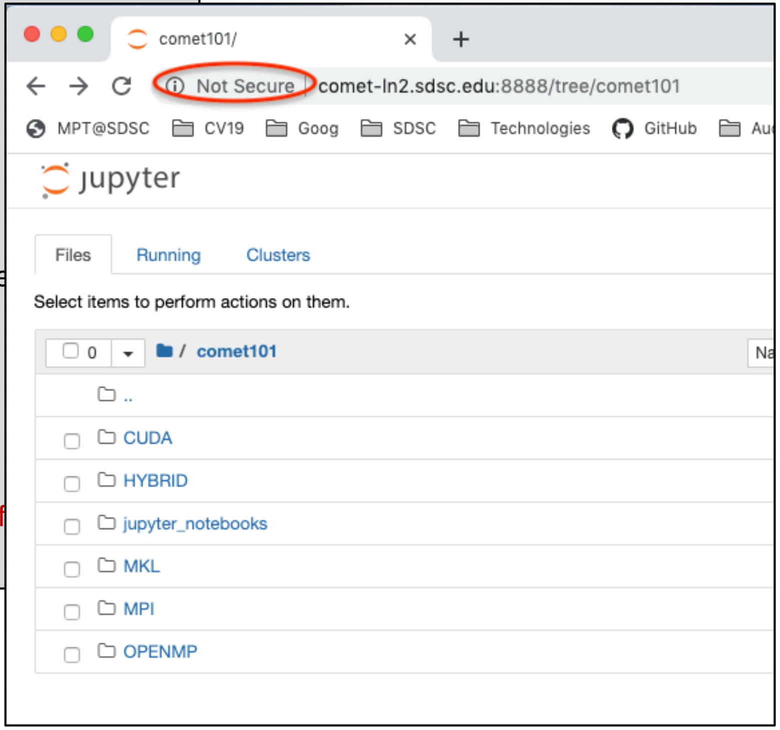
Very secure but somewhat complicated and hard to keep running

SSH Tunneling @ Work:

Uses Local Port Forwarding to connect to a Jupyter Notebook Server

```
(base) quantum:Docs username$ ssh –L 8888:127.0.0.1:8888 username@comet.sdsc.edu
```

```
(base) [username@comet-ln2:~] jupyter notebook --no-browser --ip=`/bin/hostname`
[I 12:03:54.005 NotebookApp] JupyterLab extension loaded from
/home/username/miniconda3/lib/python3.7/site-packages/jupyterlab
[I 12:03:54.005 NotebookApp] JupyterLab application directory is
/home/username/miniconda3/share/jupyter/lab
[I 12:03:54.497 NotebookApp] Serving notebooks from local directory: /home/username
[I 12:03:54.497 NotebookApp] The Jupyter Notebook is running at:
[I 12:03:54.498 NotebookApp] http://comet-
ln2.sdsc.edu:8888/?token=bc1a7238d7dd6d401cd099a7e863d5bfb6db8a6a7f19a243
[I 12:03:54.498 NotebookApp]  or
http://127.0.0.1:8888/?token=bc1a7238d7dd6d401cd099a7e863d5bfb6db8a6a7f19a243
[I 12:03:54.498 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice
confirmation).
[C 12:03:54.505 NotebookApp]

    To access the notebook, open this file in a browser:
        file:///home/username/.local/share/jupyter/runtime/nbserver-650-open.html
    Or copy and paste one of these URLs:
        http://comet-ln2.sdsc.edu:8888/?token=bc1a7238d7dd6d401cd099a7e863d5bfb6db8a6a7f
     or http://127.0.0.1:8888/?token=bc1a7238d7dd6d401cd099a7e863d5bfb6db8a6a7f19a243
```



comet-ln2.sdsc.edu:8888/tree/comet101

Select items to perform actions on them.

SDSC Jupyter Reverse Proxy Service (JRPS)

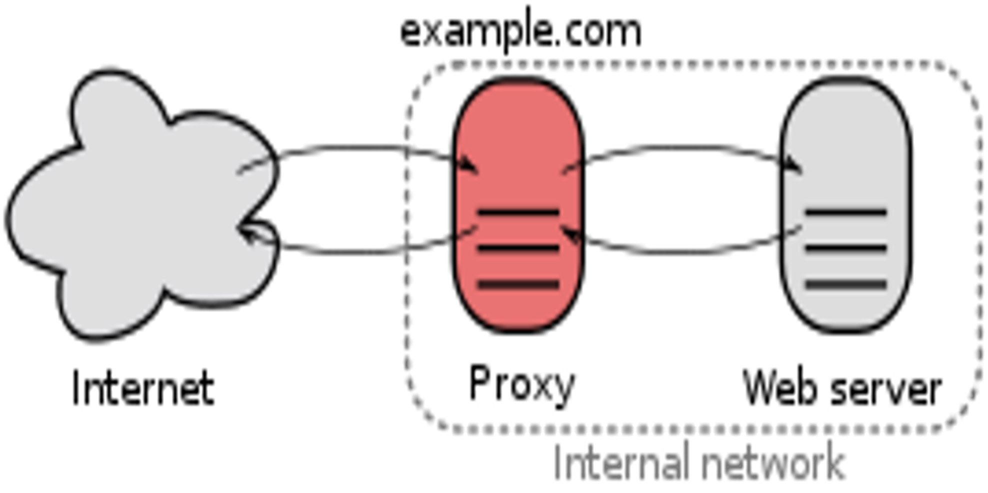
(beta version)

https://comet-notebooks-101.readthedocs.io/en/latest/methods/reverseProxy.html

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

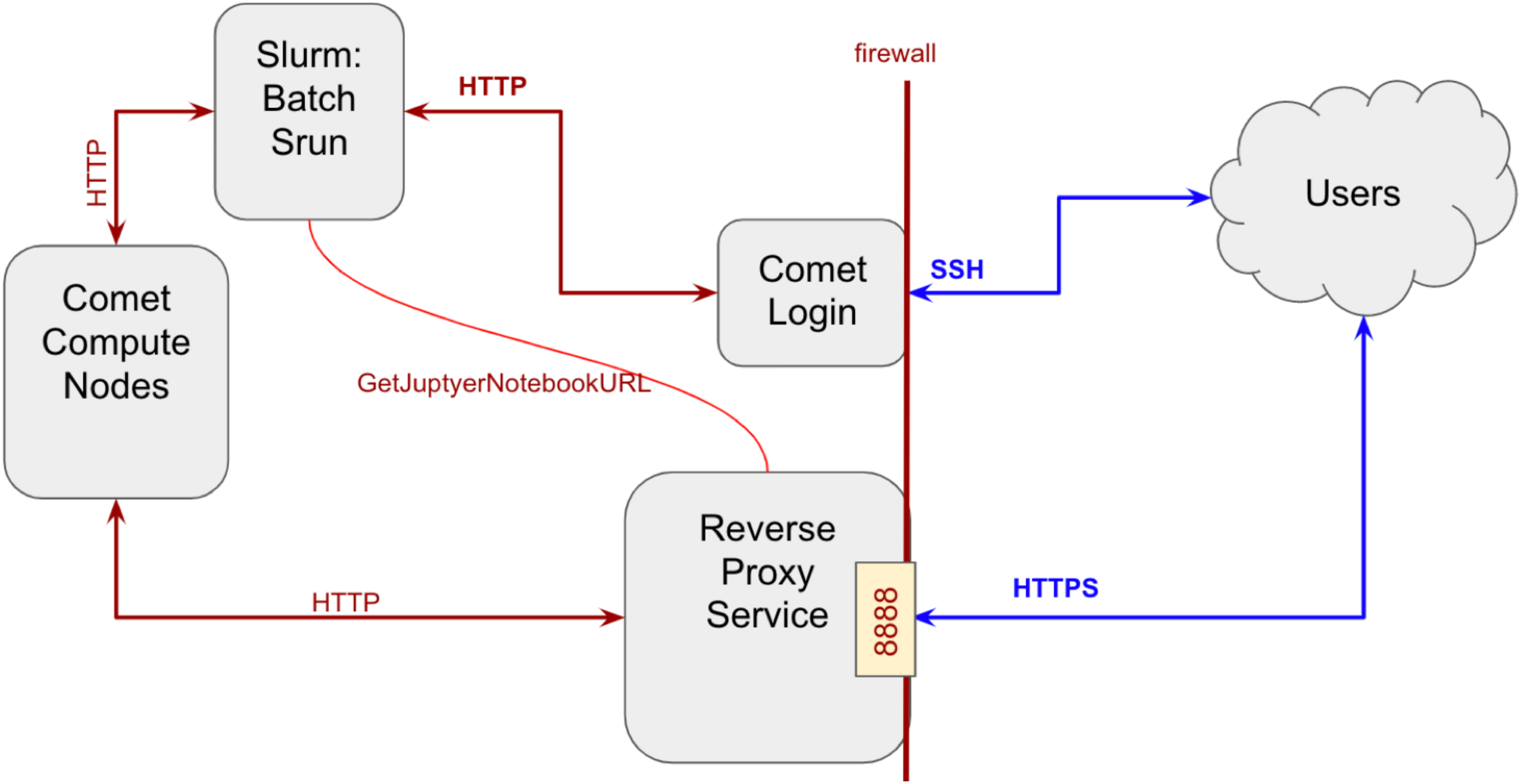
What is a Reverse Proxy?

- A reverse proxy takes requests from the Internet and forwards them to servers in an internal network. Those making requests to the proxy may not be aware of the internal network.



Img Source: Wikipedia reverse proxy

JRPS Architecture



SDSC Jupyter Reverse Proxy Service

- RPS is a new approach that will allow users to launch standard Jupyter Notebooks on on any Comet compute node using a secure reverse proxy server.
- The notebooks will be hosted on the internal cluster network as an HTTP service using standard jupyter commands.
- The service will then be made available to the user outside of the cluster firewall as an HTTPS connection between the external user's web browser and the reverse proxy server.
- The goal is to minimize software changes for our users while improving the security of user notebooks running on our HPC systems.
- The JRPS service is capable of running on any HPC system capable of supporting the RP server (needs Apache)

SDSC Reverse Proxy Service Overview

- Using RPS is very simple and requires no tunneling and is secure (produces HTTPS URLs).
- To use RPS:
  - SSH to a comet login node.
  - Clone the Repo:

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

  - Check your software environment on the login node: Anaconda, conda, Jupyter (notebooks, lab), and other Python packages needed for you application.
    - See: https://comet-notebooks-101.readthedocs.io/en/latest/prerequisites.html

- Follow conda/mininconda installation instructions
- Clone the JRPS repo:

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

- Start the notebook
- Capture the URL & enter into a web browser
- Monitor the Job queue

```
(base) [mthomas@comet-ln2:~/reverse-proxy] ./start_notebook
/home/mthomas/.jupyter
Assuming user is mthomas
Your notebook is here:
https://babbling-cedar-deviation.comet-user-
content.sdsc.edu?token=b3877c3146f6bfb83ebbbcd14a2b83e4
Using default partition: compute
No time allotment given. Default is 30 mins
No batch script specified. Using ./batch/batch_notebook.sh
Submitted batch job 35161444
(base) [mthomas@comet-ln2:~/reverse-proxy] squeue -u mthomas
             JOBID PARTITION     NAME     USER ST       TIME  NODES
NODELIST(REASON)
          35161444   compute batch_no  mthomas PD       0:00      1 (Resources)
(base) [mthomas@comet-ln2:~/reverse-proxy] squeue -u mthomas
             JOBID PARTITION     NAME     USER ST       TIME  NODES
NODELIST(REASON)
          35161444   compute batch_no  mthomas PD       0:00      1 (Resources)
(base) [mthomas@comet-ln2:~/reverse-proxy] squeue -u mthomas
             JOBID PARTITION     NAME     USER ST       TIME  NODES
NODELIST(REASON)
          35161444   compute batch_no  mthomas  R       0:33      1 comet-18-29
```

SDSC Reverse Proxy Service Team

- Project Team:
  - Scott Sakai (SDSC)
  - Marty Kandes (SDSC)
  - Mary Thomas (SDSC)
  - James McDougall (UCSD Undergraduate)

- Project Status:
  - JRPS is in beta testing.
  - Please give it a try. If you have trouble, help@xsede.org
  - Send feedback to mpthomas at ucsd dot edu.