JupyterHub Workshop 2016

Meeting Friday July 22nd, 2016

Quick Tips

- **Interact** for file sharing (URL based content sharing—makes it easier for users)
  Ryan Lovett—UC Berkeley Statistics
- **Ansible** for deployment
-

Common Themes/Visions: (possible projects)

- Find out how many users JupyterHub can support
- Supporting many (200+) users on single JupyterHub server
- Steep learning curve for git (Interact [mentioned in "Quick Tips"] as possible solution)
- Saving history when you exit the shell
- Integrating .ipynb files with Google Drive -- "open with"
-

Overall Notes

Brian Granger – Cal Poly SLO
"Deploying JupyterHub"

Small/medium groups of mostly trusted
users for non-dev ops folks

Ansible Based Deployment

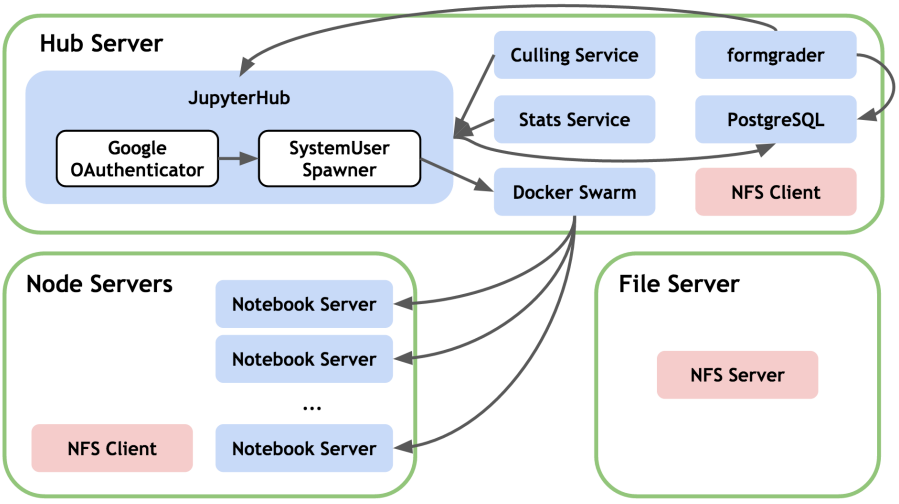
- We have encoded this deployment scenario as a set of configurable ansible roles [https://github.com/jupyterhub/jupyterhub-deployteaching]
- Used to teach data science at Cal Poly over multiple quarter, multiple instructors
- Can build a new server in <1 hour

Install uses conda and some pip

Jess Hamrick – UC Berkeley
"Scaling Up JupyterHub for Education"

Scaling up to include more users (200+)

1. Isolating users with Docker
    - Isolated environment
2. Isolating 200+ users with Docker Swarm
3. Persistent files using NFS



4. Managing everything with Ansible and Docker Compose

[DockerSpawner and SystemUserSpawner:
https://github.com/jupyterhub/dockerspawner]

Talking to JupyterHub:

1. Get an API token
2. Access the JupyterHub API using http://<url>:8081/hub/api/users
   - Headers: {'authorization': 'token <hub_api_token>'}
   - https://github.com/jupyterhub/jupyterhub/tree/master/jupyterhub/apihandlers

Getting a JupyterHub URL:

1. Run your service at:
   - http://<service_url>:<service_port>
2. Access the proxy API
   - POST http://<url>:8001/api/routes/myservice
   - Headers: {'authorization': 'token <proxy_api_token>'}
   - Body: {'target': 'http://<service_url>:<service_port>'}
3. Access your service at:
   - http://<url>:8000/myservice

Authenticating with JupyterHub:

1. Get an API token:
   - JupyterHub token
2. Use the HubAuth service:
   - http://jupyterhub.readthedocs.io/en/latest/api/services.auth.html
3. Example use case: nbgrader formgrade

Ryan Lovett – UC Berkeley

"Foundations of Data Science" Course

Computer Science & Statistics Dept.

1. Isolating users with Docker
   - Isolated environment
2. Isolating 200+ users with Docker Swarm

Persistent files using NFS