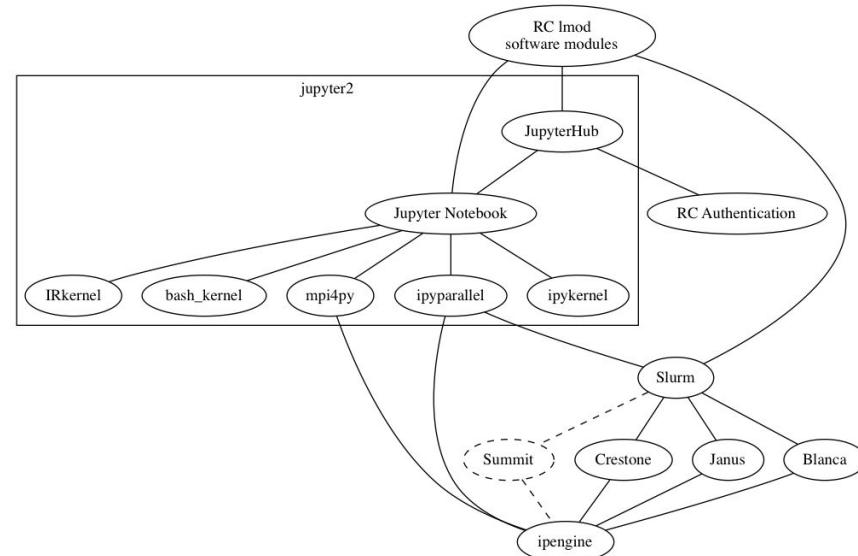


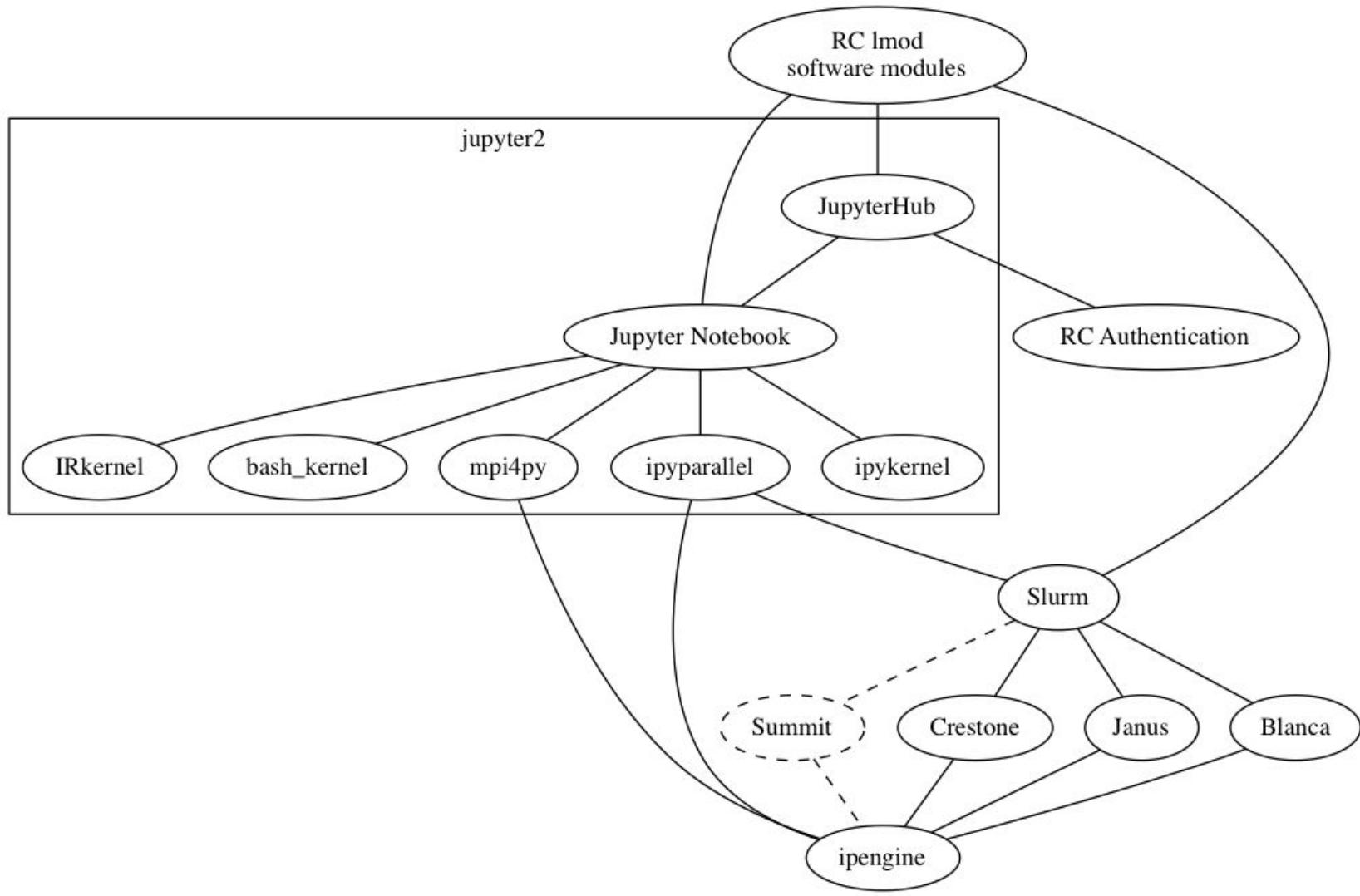
# JupyterHub Deployment

at University of Colorado Boulder

# Overview

- The existing RC environment
- Installing JupyterHub in a virtualenv
- Configuring JupyterHub
- Running JupyterHub as a system service
- Upstream authentication
- Additional kernels
- ipyparallel and mpi4py
- User initialization
- Configuring selinux
- Future improvements and more detail





<https://goo.gl/w79Hpn>

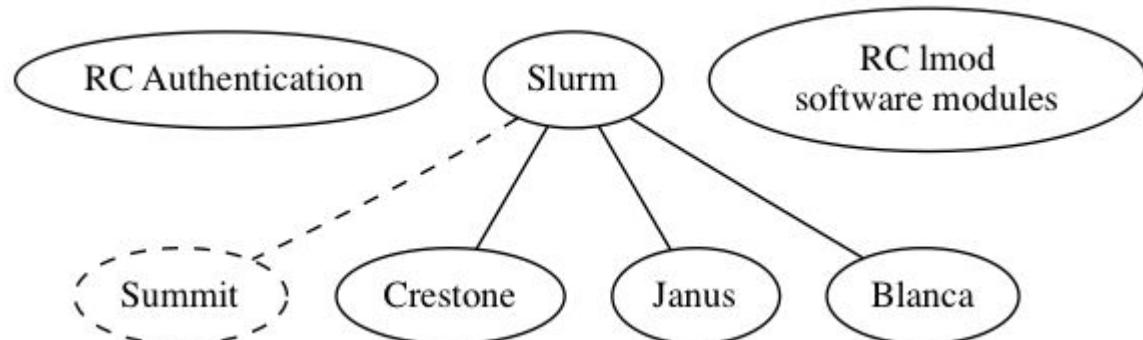
<https://drive.google.com/folderview?id=0B5QNtkHgWksyc2pvUWotRHVGZ3M>

github.com/ResearchComputing/XSEDE-2016-jupyterhub

# The existing RC environment

# The existing RC environment

- Slurm clusters
  - Janus
  - Blanca
  - Crestone
  - Summit
- Lmod “environment modules”
- Shared filesystems
  - /home/
  - /projects/
  - /curc/tools/
  - /work/
  - Various scratch systems
- Identity management
  - LDAP
  - Kerberos
  - RADIUS



# Setting up the initial virtualenv



```
joan5896 — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc +[+]
[joan5896@jupyter2 ~]$ sudo mkdir -p /opt/jupyterhub
[joan5896@jupyter2 ~]$ sudo chgrp rcopspgrp /opt/jupyterhub
[joan5896@jupyter2 ~]$ sudo sudo chmod g+w /opt/jupyterhub
[joan5896@jupyter2 ~]$ module load intel mkl python/3.4.3 nodejs
[joan5896@jupyter2 ~]$ virtualenv /opt/jupyterhub/ --system-site-packages
```

```
joan5896 — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc +  
[joan5896@jupyter2 ~]$ sudo mkdir -p /opt/jupyterhub  
[joan5896@jupyter2 ~]$ sudo chgrp rcopspgrp /opt/jupyterhub  
[joan5896@jupyter2 ~]$ sudo sudo chmod g+w /opt/jupyterhub  
[joan5896@jupyter2 ~]$ module load intel mkl python/3.4.3 nodejs  
[joan5896@jupyter2 ~]$ virtualenv /opt/jupyterhub/ --system-site-packages  
Using base prefix '/curc/tools/x86_64/rh6/software/python/3.4.3/intel/15.0.2'  
New python executable in /opt/jupyterhub/bin/python3.4  
Also creating executable in /opt/jupyterhub/bin/python  
Installing setuptools, pip, wheel...done.  
[joan5896@jupyter2 ~]$
```

pip install

```
joan5896 — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc
[joan5896@jupyter2 ~]$ source /opt/jupyterhub/bin/activate
(jupyterhub) [joan5896@jupyter2 ~]$ pip install jupyterhub jupyter
```

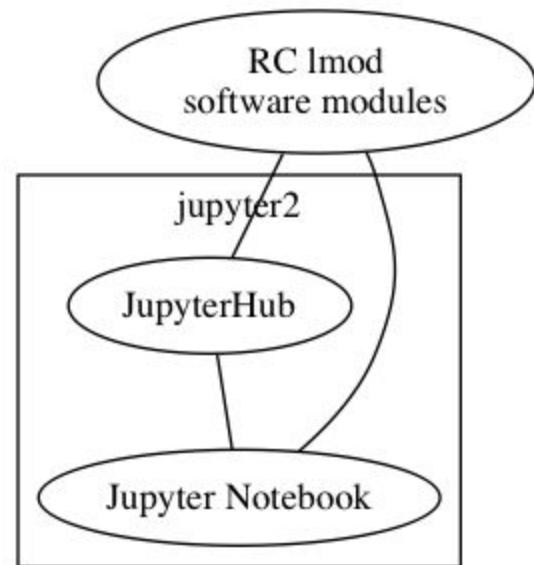
```
joan5896 — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc +
```

```
[joan5896@jupyter2 ~]$ source /opt/jupyterhub/bin/activate
(jupyterhub) [joan5896@jupyter2 ~]$ pip install jupyterhub jupyter
Collecting jupyterhub
  Downloading jupyterhub-0.6.1-py3-none-any.whl (1.3MB)
    100% |██████████| 1.4MB 461kB/s
Collecting jupyter
  Downloading jupyter-1.0.0-py2.py3-none-any.whl
Collecting jinja2 (from jupyterhub)
  Downloading Jinja2-2.8-py2.py3-none-any.whl (263kB)
    100% |██████████| 266kB 1.7MB/s
Collecting sqlalchemy>=1.0 (from jupyterhub)
  Downloading SQLAlchemy-1.0.13.tar.gz (4.8MB)
    99% |██████████| 4.8MB 19.2MB/s eta 0:00:01
```

```
joan5896 — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc +
```

Stored in directory: /home/joan5896/.cache/pip/wheels/a3/fa/dc/0198eed9ad95489  
b8a4f45d14dd5d2aee3f8984e46862c5748  
Running setup.py bdist\_wheel for terminado ... done  
Stored in directory: /home/joan5896/.cache/pip/wheels/3b/c2/ea/af635ffb63857a8  
c2ddd22da6a4b52f5b7ea3065db94ef5d7c  
Running setup.py bdist\_wheel for simplegeneric ... done  
Stored in directory: /home/joan5896/.cache/pip/wheels/7b/31/08/c85e74c84188cbe  
c6a6827beec4d640f2bd78ae003dc1ec09d  
Successfully built sqlalchemy tornado MarkupSafe terminado simplegeneric  
Installing collected packages: MarkupSafe, jinja2, sqlalchemy, decorator, ipytho  
n-genutils, traitlets, pamela, requests, backports-abc, tornado, jupyterhub, sim  
plegeneric, backports.shutil-get-terminal-size, ptyprocess, pexpect, pickleshare  
, ipython, jupyter-core, pyzmq, jupyter-client, ipykernel, pygments, mistune, js  
onschema, nbformat, entrypoints, nbconvert, terminado, notebook, widgetsnbextens  
ion, ipywidgets, jupyter-console, qtconsole, jupyter  
Successfully installed MarkupSafe-0.23 backports-abc-0.4 backports.shutil-get-te  
rminal-size-1.0.0 decorator-4.0.10 entrypoints-0.2.2 ipykernel-4.3.1 ipython-4.2  
.0 ipython-genutils-0.1.0 ipywidgets-5.1.5 jinja2-2.8 jsonschema-2.5.1 jupyter-1  
.0.0 jupyter-client-4.3.0 jupyter-console-4.1.1 jupyter-core-4.1.0 jupyterhub-0.  
6.1 mistune-0.7.2 nbconvert-4.2.0 nbformat-4.0.1 notebook-4.2.1 pamela-0.2.1 pex  
pect-4.1.0 pickleshare-0.7.2 ptyprocess-0.5.1 pygments-2.1.3 pyzmq-15.2.0 qtcons  
ole-4.2.1 requests-2.10.0 simplegeneric-0.8.1 sqlalchemy-1.0.13 terminado-0.6 to  
rnado-4.3 traitlets-4.2.1 widgetsnbextension-1.2.3  
(jupyterhub) [joan5896@jupyter2 ~]\$

# first run



joan5896 — joan5896@jupyter2:~ — ssh \* ssh joan5896@jupyter2.rc — 80x24  
~ — joan5896@jupyter2:~ — ssh \* ssh joan5896@jupyter2.rc

```
(jupyterhub) [joan5896@jupyter2 ~]$ jupyterhub --no-ssl
```

```
joan5896 — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc +
```

joan5896/jupyterhub\_cookie\_secret  
[W 2016-06-26 22:21:22.564 JupyterHub app:304]  
Generating CONFIGPROXY\_AUTH\_TOKEN. Restarting the Hub will require restarting the proxy.  
Set CONFIGPROXY\_AUTH\_TOKEN env or JupyterHub.proxy\_auth\_token config to avoid this message.

[W 2016-06-26 22:21:22.578 JupyterHub app:757] No admin users, admin interface will be unavailable.

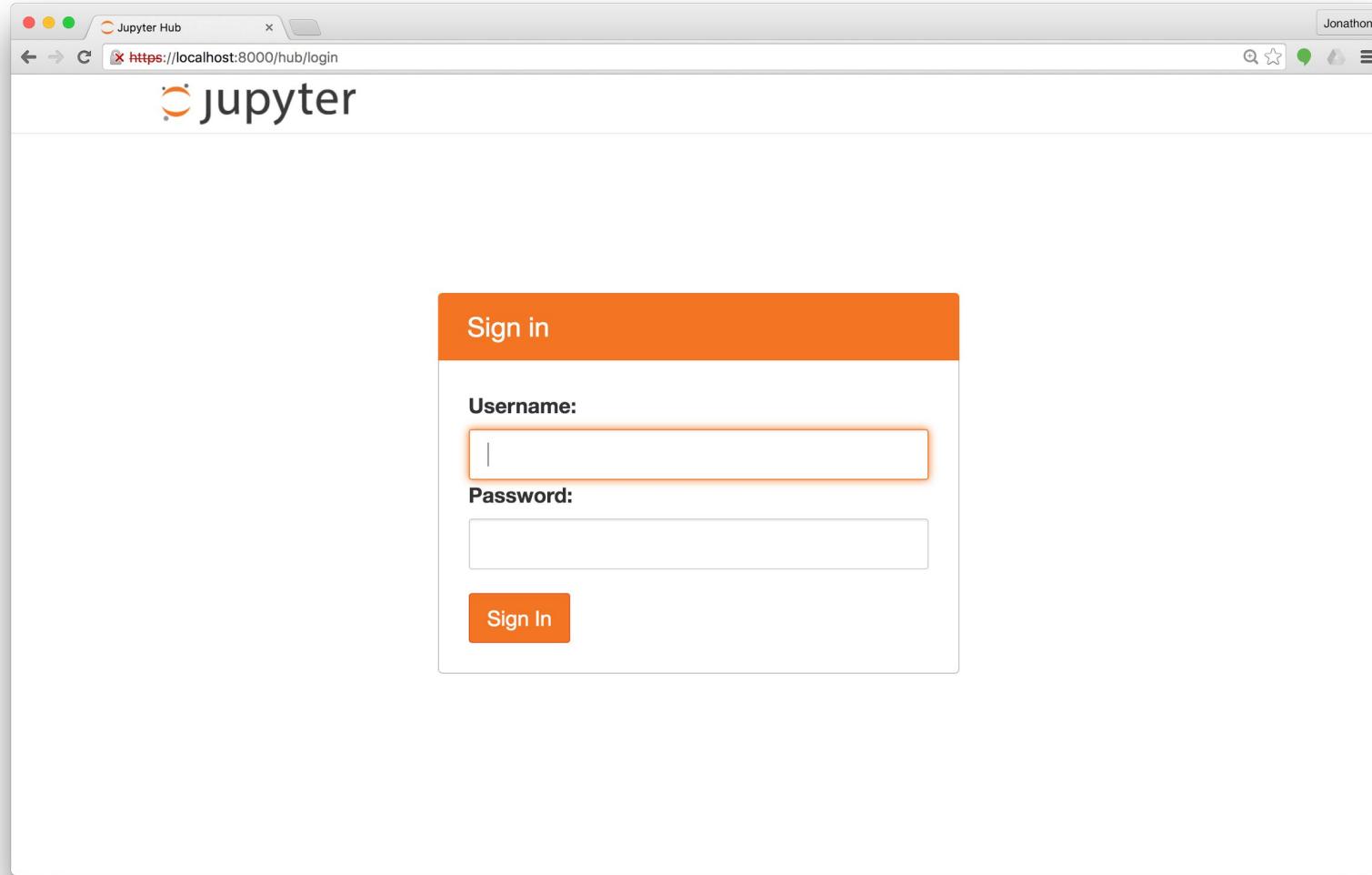
[W 2016-06-26 22:21:22.578 JupyterHub app:758] Add any administrative users to `c.Authenticator.admin\_users` in config.

[I 2016-06-26 22:21:22.579 JupyterHub app:785] Not using whitelist. Any authenticated user will be allowed.

[I 2016-06-26 22:21:22.631 JupyterHub app:1231] Hub API listening on http://127.0.0.1:8081/hub/

[W 2016-06-26 22:21:22.647 JupyterHub app:959] Running JupyterHub without SSL. There better be SSL termination happening somewhere else...

[I 2016-06-26 22:21:22.647 JupyterHub app:968] Starting proxy @ http://\*:8000/ 22:21:22.861 - info: [ConfigProxy] Proxying http://\*:8000 to http://127.0.0.1:8081  
22:21:22.866 - info: [ConfigProxy] Proxy API at http://127.0.0.1:8001/api/routes  
[I 2016-06-26 22:21:22.877 JupyterHub app:1254] JupyterHub is now running at http://127.0.0.1:8000/



# initial configuration

```
joan5896 — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc
(jupyterhub) [joan5896@jupyter2 ~]$ jupyterhub --help
```

```
joan5896 — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc
Q admin + Done
Path to SSL key file for the public facing interface of the proxy
Use with ssl_cert
--config=<Unicode> (JupyterHub.config_file)
    Default: 'jupyterhub_config.py'
    The config file to load
-f <Unicode> (JupyterHub.config_file)
    Default: 'jupyterhub_config.py'
    The config file to load

To see all available configurables, use `--help-all`

Examples
-----
generate default config file:

    jupyterhub --generate-config -f /etc/jupyterhub/jupyterhub.py

spawn the server on 10.0.1.2:443 with https:

    jupyterhub --ip 10.0.1.2 --port 443 --ssl-key my_ssl.key --ssl-cert my_ssl.cert

(jupyterhub-tutorial) [joan5896@jupyter2 ~]$
```

```
joan5896 — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc
(jupyterhub) [joan5896@jupyter2 ~]$ jupyterhub --generate-config \
> -f jupyterhub.py
Writing default config to: jupyterhub.py
(jupyterhub) [joan5896@jupyter2 ~]$
```

```
joan5896 — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc

# Configuration file for jupyterhub.

#-----
# Configurable configuration
#-----

#-----
# LoggingConfigurable configuration
#-----


# A parent class for Configurables that log.
#
# Subclasses have a log trait, and the default behavior is to get the logger
# from the currently running Application.

#-----
# SingletonConfigurable configuration
#-----


# A configurable that only allows one instance.
#
# This class is for classes that should only have one instance of itself or
# *any* subclass. To create and retrieve such a class use the
jupyterhub.py
```

A screenshot of a Mac OS X desktop showing a web browser window. The window title is "jupyterhub.readthedocs.io/en/latest/". The address bar also shows "cripps.local | Syncthing" and other tabs like "Summit", "16 Abandone...", "Go", "Free Music Archive", "Pocket", "CU", "CURC", "civilfritz", "Topics", "Git tutorial", "How To Make ...ular Science", and "here".

The main content area displays the "JupyterHub" documentation. The page title is "Docs » JupyterHub". There is a "Edit on GitHub" button. The main heading is "JupyterHub". Below it, a paragraph explains that JupyterHub is a server for multiple users. A larger text block details how to use JupyterHub, mentioning Unix servers, authentication (using pam), and spawners (like Docker). It also notes that JupyterHub runs as three parts: a multi-user hub, a configurable proxy, and single-user servers.

**JupyterHub**

JupyterHub is a server that gives multiple users access to Jupyter notebooks, running an independent Jupyter notebook server for each user.

To use JupyterHub, you need a Unix server (typically Linux) running somewhere that is accessible to your team on the network. The JupyterHub server can be on an internal network at your organisation, or it can run on the public internet (in which case, take care with [security](#)). Users access JupyterHub in a web browser, by going to the IP address or domain name of the server.

Different [authenticators](#) control access to JupyterHub. The default one (pam) uses the user accounts on the server where JupyterHub is running. If you use this, you will need to create a user account on the system for each user on your team. Using other authenticators, you can allow users to sign in with e.g. a Github account, or with any single-sign-on system your organisation has.

Next, [spawners](#) control how JupyterHub starts the individual notebook server for each user. The default spawner will start a notebook server on the same machine running under their system username. The other main option is to start each server in a separate container, often using Docker.

JupyterHub runs as three separate parts:

- The multi-user Hub (Python & Tornado)
- A [configurable http proxy](#) (NodeJS)
- Multiple single-user Jupyter notebook servers (Python & Tornado)

Basic principles:

- Hub spawns proxy
- Proxy forwards ~all requests to hub by default

**jupyterhub.readthedocs.io**

cripps.local | Syncthing Summit 16 Abandoned Goosebumps Go ▾ Free Music Archive + Pocket CU ▾ CURC ▾ civilfritz ▾ Topics ▾ Git tutorial ▾ How To Make ...ular Science here

# How to configure JupyterHub

JupyterHub is configured in two ways:

1. Configuration file
2. Command-line arguments

## Configuration file

By default, JupyterHub will look for a configuration file (which may not be created yet) named `jupyterhub_config.py` in the current working directory. You can create an empty configuration file with:

```
jupyterhub --generate-config
```

This empty configuration file has descriptions of all configuration variables and their default values. You can load a specific config file with:

```
jupyterhub -f /path/to/jupyterhub_config.py
```

See also: [general docs](#) on the config system Jupyter uses.

## Command-line arguments

Type the following for brief information about the command-line arguments:

```
jupyterhub -h
```

jupyterhub.readthedocs.io/en/latest/getting-started.html

```
jupyterhub --help-all
```

```
joan5896 — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc
+
# https://jupyterhub.readthedocs.io/en/latest/getting-started.html#configuration
#-file

c.JupyterHub.ip = '0.0.0.0'
c.JupyterHub.port = 443

c.JupyterHub.hub_ip = 'jupyter2.rc.int.colorado.edu'

c.Spawner.env_keep.extend(['LD_LIBRARY_PATH'])

c.JupyterHub.cookie_secret_file = '/opt/jupyterhub/var/jupyterhub_cookie_secret'

c.JupyterHub.db_url = '/opt/jupyterhub/var/jupyterhub.sqlite'

c.JupyterHub.ssl_cert = '/opt/jupyterhub/etc/jupyter.rc.int.colorado.edu.crt'
c.JupyterHub.ssl_key = '/opt/jupyterhub/etc/jupyter.rc.int.colorado.edu.key'

c.PAMAuthenticator.service = 'jupyterhub'

c.Authenticator.admin_users = { 'holtat', 'joan5896' }

~
~
~

"/opt/jupyterhub/etc/jupyterhub.py" 19L, 641C
```

Jupyterhub.service  
and the wrapper script

```
joan5896 — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc
[joan5896@jupyter2 ~]$ cat /etc/systemd/system/jupyterhub.service
[Unit]
Description=JupyterHub – A multi-user server for Jupyter notebooks

[Service]
Type=simple
ExecStart=/usr/local/bin/jupyterhub -f /opt/jupyterhub/etc/jupyterhub.py

[Install]
WantedBy=multi-user.target
[joan5896@jupyter2 ~]$
```

```
joan5896 — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc
[joan5896@jupyter2 ~]$ cat /usr/local/bin/jupyterhub
#!/bin/bash

function main
{
    source /etc/profile.d/modules.sh
    module load intel mkl python/3.4.3 nodejs
    source /opt/jupyterhub/bin/activate
    exec jupyterhub "$@"
}

main "$@"
[joan5896@jupyter2 ~]$
```

```
joan5896 — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc +
```

```
[joan5896@jupyter2 ~]$ sudo systemctl start jupyterhub
[joan5896@jupyter2 ~]$ sudo systemctl status jupyterhub
● jupyterhub.service - JupyterHub - A multi-user server for Jupyter notebooks
  Loaded: loaded (/etc/systemd/system/jupyterhub.service; enabled; vendor prese
t: disabled)
    Active: active (running) since Tue 2016-06-28 13:47:16 MDT; 19s ago
      Main PID: 30592 (jupyterhub)
        CGroup: /system.slice/jupyterhub.service
                  └─30592 /opt/jupyterhub/bin/python3.4 /opt/jupyterhub/bin/jupyter...
                      ├─30626 node /curc/tools/x86_64/rh6/software/nodejs/4.2.4/node-v4....
```

```
Jun 28 13:47:17 jupyter2.rc.int.colorado.edu jupyterhub[30592]: [I 2016-06-28...
Jun 28 13:47:18 jupyter2.rc.int.colorado.edu jupyterhub[30592]: [W 2016-06-28...
Jun 28 13:47:18 jupyter2.rc.int.colorado.edu jupyterhub[30592]: Generating CO...
Jun 28 13:47:18 jupyter2.rc.int.colorado.edu jupyterhub[30592]: Set CONFIGPRO...
Jun 28 13:47:18 jupyter2.rc.int.colorado.edu jupyterhub[30592]: [I 2016-06-28...
Jun 28 13:47:18 jupyter2.rc.int.colorado.edu jupyterhub[30592]: [I 2016-06-28...
Jun 28 13:47:18 jupyter2.rc.int.colorado.edu jupyterhub[30592]: [I 2016-06-28...
Jun 28 13:47:18 jupyter2.rc.int.colorado.edu jupyterhub[30592]: 13:47:18.716 ...
Jun 28 13:47:18 jupyter2.rc.int.colorado.edu jupyterhub[30592]: 13:47:18.721 ...
Jun 28 13:47:18 jupyter2.rc.int.colorado.edu jupyterhub[30592]: [I 2016-06-28...
Hint: Some lines were ellipsized, use -l to show in full.
[joan5896@jupyter2 ~]$
```

Jupyter Hub

https://jupyter2.rc.int.colorado.edu/hub/login

# jupyter

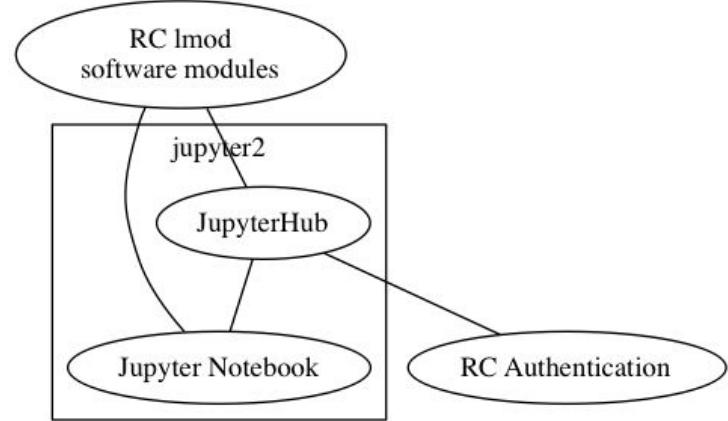
### Sign in

**Username:**

**Password:**

**Sign In**

# Authentication and PAM



The screenshot shows a web browser window with the URL [jupyterhub.readthedocs.io/en/latest/authenticators.html](https://jupyterhub.readthedocs.io/en/latest/authenticators.html). The page is titled "Writing a custom Authenticator".

**Left Sidebar (Table of Contents):**

- USER DOCUMENTATION
  - Getting started with JupyterHub
  - Further reading
  - How JupyterHub works
  - Web Security in JupyterHub
- CONFIGURATION
- Writing a custom Authenticator
  - Basics of Authenticators
  - Normalizing and validating usernames
  - OAuth and other non-password logins
- Writing a custom Spawner
- Troubleshooting
- DEVELOPER DOCUMENTATION
- The JupyterHub API
- COMMUNITY DOCUMENTATION
- ABOUT JUPYTERHUB
- Summary of changes in JupyterHub
- QUESTIONS? SUGGESTIONS?
- Jupyter mailing list
- Inviter website

**Page Content:**

**Section Header:** Writing a custom Authenticator

The **Authenticator** is the mechanism for authorizing users. Basic authenticators use simple username and password authentication. JupyterHub ships only with a [PAM](#)-based Authenticator, for logging in with local user accounts.

You can use custom Authenticator subclasses to enable authentication via other systems. One such example is using [GitHub OAuth](#).

Because the username is passed from the Authenticator to the Spawner, a custom Authenticator and Spawner are often used together.

See a list of custom Authenticators [on the wiki](#).

**Section Header:** Basics of Authenticators

A basic Authenticator has one central method:

**Authenticator.authenticate**

```
Authenticator.authenticate(handler, data)
```

This method is passed the tornado RequestHandler and the POST data from the login form. Unless the login form has been customized, `data` will have two keys:

jupyterhub.readthedocs.io/en/latest/authenticators.html

```
joan5896 — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc
[joan5896@jupyter2 ~]$ grep -i pam /opt/jupyterhub/etc/jupyterhub.py
c.PAMAuthenticator.service = 'jupyterhub'
[joan5896@jupyter2 ~]$
```

Jupyter Hub

https://jupyter2.rc.int.colorado.edu/hub/login

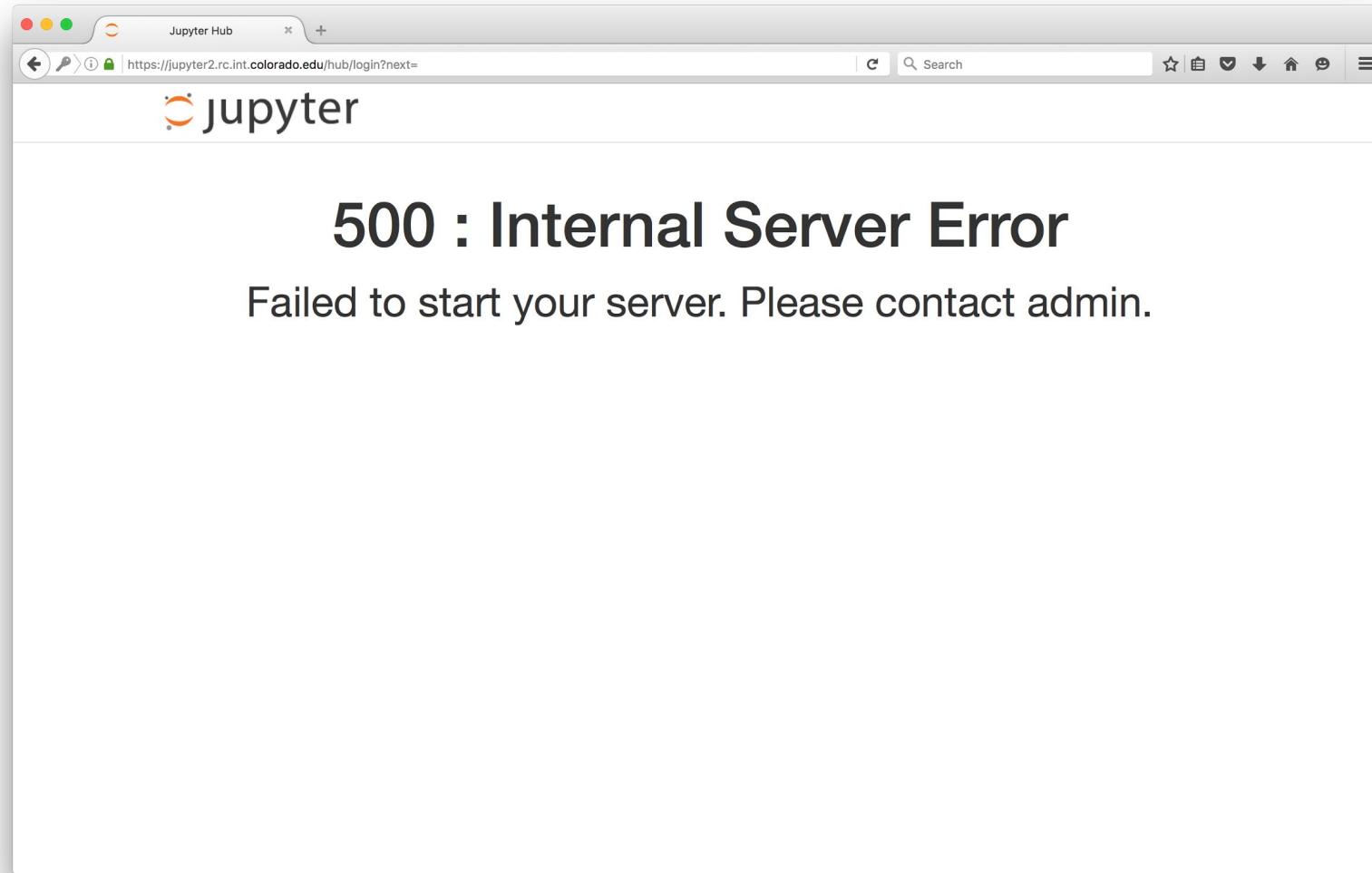
# jupyter

### Sign in

**Username:**

**Password:**

**Sign In**



```
joan5896 — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc
[joan5896@jupyter2 ~]$ sudo journalctl -xfu jupyterhub.service
```

```
joan5896 — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc +
```

Jun 28 14:18:54 jupyter2.rc.int.colorado.edu jupyterhub[30592]: File "/curc/tools/x86\_64/rh6/software/python/3.4.3/intel/15.0.2/lib/python3.4/subprocess.py", line 859, in \_\_init\_\_  
Jun 28 14:18:54 jupyter2.rc.int.colorado.edu jupyterhub[30592]: restore\_signals, start\_new\_session)  
Jun 28 14:18:54 jupyter2.rc.int.colorado.edu jupyterhub[30592]: File "/curc/tools/x86\_64/rh6/software/python/3.4.3/intel/15.0.2/lib/python3.4/subprocess.py", line 1457, in \_execute\_child  
Jun 28 14:18:54 jupyter2.rc.int.colorado.edu jupyterhub[30592]: raise child\_exception\_type(errno\_num, err\_msg)  
Jun 28 14:18:54 jupyter2.rc.int.colorado.edu jupyterhub[30592]: PermissionError:  
[Errno 13] Permission denied  
Jun 28 14:18:54 jupyter2.rc.int.colorado.edu jupyterhub[30592]: [E 2016-06-28 14:18:54.952 JupyterHub log:99] {  
Jun 28 14:18:54 jupyter2.rc.int.colorado.edu jupyterhub[30592]: "Accept": "text/html,application/xhtml+xml,application/xml;q=0.9,\*/\*;q=0.8",  
Jun 28 14:18:54 jupyter2.rc.int.colorado.edu jupyterhub[30592]: "X-Forwarded-For": "128.138.138.146",  
Jun 28 14:18:54 jupyter2.rc.int.colorado.edu jupyterhub[30592]: "Content-Type": "application/x-www-form-urlencoded",  
Jun 28 14:18:54 jupyter2.rc.int.colorado.edu jupyterhub[30592]: "X-Forwarded-Proto": "https",  
Jun 28 14:18:54 jupyter2.rc.int.colorado.edu jupyterhub[30592]: "Referer": "https://jupyter2.rc.int.colorado.edu/hub/login",

```
joan5896 — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc +[

[joan5896@jupyter2 ~]$ sudo ausearch --message avc --start recent --interpret
-----
type=SYSCALL msg=audit(06/28/2016 14:18:54.825:4387) : arch=x86_64 syscall=execve success=no exit=-13(Permission denied) a0=0x1c758f0 a1=0x1f164a0 a2=0x1cec7f0 a3=0x1 items=0 ppid=30592 pid=30874 auid=joan5896 uid=joan5896 gid=joan5896pgrp euid=joan5896 suid=joan5896 fsuid=joan5896 egid=joan5896pgrp sgid=joan5896pgrp fsgid=joan5896pgrp tty=(none) ses=28 comm=jupyterhub exe=/opt/jupyterhub/bin/python3.4 subj=system_u:system_r:unconfined_service_t:s0 key=(null)
type=AVC msg=audit(06/28/2016 14:18:54.825:4387) : avc: denied { transition } for pid=30874 comm=jupyterhub path=/opt/jupyterhub/bin/jupyterhub-singleuser dev="sda2" ino=35957599 scontext=system_u:system_r:unconfined_service_t:s0 tcontext=u:u:u:u tclass=process
[joan5896@jupyter2 ~]$
```

```
[joan5896@jupyter2 ~]$ sudo ausearch --message avc --start recent --interpret
-----
type=SYSCALL msg=audit(06/28/2016 14:18:54.825:4387) : arch=x86_64 syscall=execve success=no exit=-13(Permission denied) a0=0x1c758f0 a1=0x1f164a0 a2=0x1cec7f0 a3=0x1 items=0 ppid=30592 pid=30874 auid=joan5896 uid=joan5896 gid=joan5896pgrp euid=joan5896 suid=joan5896 fsuid=joan5896 egid=joan5896pgrp sgid=joan5896pgrp fsgid=joan5896pgrp tty=(none) ses=28 comm=jupyterhub exe=/opt/jupyterhub/bin/python3.4 subj=system_u:system_r:unconfined_service_t:s0 key=(null)
type=AVC msg=audit(06/28/2016 14:18:54.825:4387) : avc: denied { transition } for pid=30874 comm=jupyterhub path=/opt/jupyterhub/bin/jupyterhub-singleuser dev="sda2" ino=35957599 scontext=system_u:system_r:unconfined_service_t:s0 tcontext=u
nconfined_u:unconfined_r:unconfined_t:s0 tclass=process
[joan5896@jupyter2 ~]$ sudo setenforce permissive
[joan5896@jupyter2 ~]$
```

Home Search Control Panel Logout

# jupyter

Files [Running](#) [IPython Clusters](#)

Select items to perform actions on them.

[Upload](#) [New ▾](#) [⟳](#)

<input type="checkbox"/>	
<input type="checkbox"/>	<a href="#">bak</a>
<input type="checkbox"/>	<a href="#">bin</a>
<input type="checkbox"/>	<a href="#">curc-bench</a>
<input type="checkbox"/>	<a href="#">curc-bench-runs</a>
<input type="checkbox"/>	<a href="#">curc-puppet</a>
<input type="checkbox"/>	<a href="#">data</a>
<input type="checkbox"/>	<a href="#">Desktop</a>
<input type="checkbox"/>	<a href="#">Documents</a>
<input type="checkbox"/>	<a href="#">Downloads</a>
<input type="checkbox"/>	<a href="#">intel</a>
<input type="checkbox"/>	<a href="#">Jupyter</a>
<input type="checkbox"/>	<a href="#">jupyterhub-tutorial</a>

Jupyter/ x

https://jupyter2.rc.int.colorado.edu/user/joan5896/tree/Jupyter

Search Control Panel Logout

jupyter

Files Running IPython Clusters

Select items to perform actions on them.

Upload New 

   / Jupyter

 ..

  Bash.ipynb

  MPI.ipynb

  Parallel.ipynb

  Python2.ipynb

  Python3.ipynb

  R.ipynb

The screenshot shows a Jupyter Notebook interface running in a web browser. The title bar indicates it's a Python3 notebook. The toolbar includes standard file operations like Open, Save, and New, along with Cell and Kernel selection buttons. A dropdown menu shows 'Python 3' is selected.

The main area displays the following code execution:

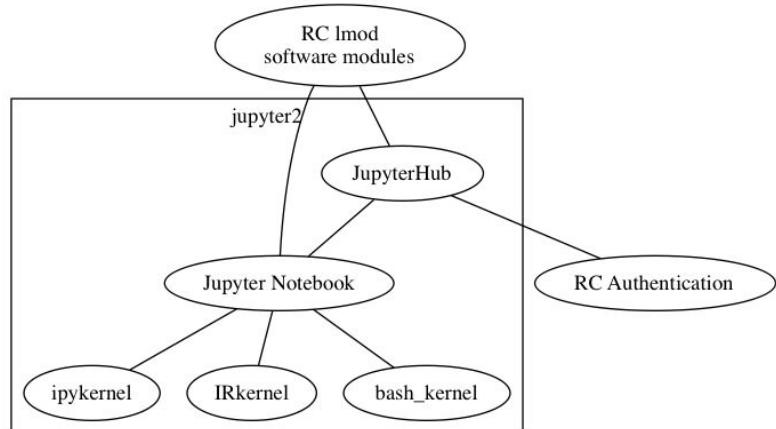
```
In [1]: print('Hello, world!')  
Hello, world!
```

```
In [2]: import sys  
sys.version  
Out[2]: '3.4.3 (default, Jul 7 2015, 08:31:16) \n[GCC Intel(R) C++ gcc 4.4 mode]  
,
```

```
In [7]: import socket  
socket.gethostname()  
Out[7]: 'jupyter2.rc.int.colorado.edu'
```

```
In [ ]:
```

# Additional kernels



bash\_kernel

```
joan5896 — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc
[joan5896@jupyter2 ~]$ module load intel mkl impi python/3.4.3
[joan5896@jupyter2 ~]$ source /opt/jupyterhub/bin/activate
(jupyterhub) [joan5896@jupyter2 ~]$ pip install bash_kernel
```

```
joan5896 — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc +
```

```
[joan5896@jupyter2 ~]$ module load intel mkl impi python/3.4.3
[joan5896@jupyter2 ~]$ source /opt/jupyterhub/bin/activate
(jupyterhub) [joan5896@jupyter2 ~]$ pip install bash_kernel
Collecting bash_kernel
  Downloading bash_kernel-0.4.1-py2.py3-none-any.whl
Requirement already satisfied (use --upgrade to upgrade): pexpect>=3.3 in /opt/jupyterhub/lib/python3.4/site-packages (from bash_kernel)
Requirement already satisfied (use --upgrade to upgrade): ptyprocess>=0.5 in /opt/jupyterhub/lib/python3.4/site-packages (from pexpect>=3.3->bash_kernel)
Installing collected packages: bash-kernel
Successfully installed bash-kernel-0.4.1
(jupyterhub) [joan5896@jupyter2 ~]$
```

```
joan5896 — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc +
```

```
[joan5896@jupyter2 ~]$ module load intel mkl impi python/3.4.3
[joan5896@jupyter2 ~]$ source /opt/jupyterhub/bin/activate
(jupyterhub) [joan5896@jupyter2 ~]$ pip install bash_kernel
Collecting bash_kernel
  Downloading bash_kernel-0.4.1-py2.py3-none-any.whl
Requirement already satisfied (use --upgrade to upgrade): pexpect>=3.3 in /opt/jupyterhub/lib/python3.4/site-packages (from bash_kernel)
Requirement already satisfied (use --upgrade to upgrade): ptyprocess>=0.5 in /opt/jupyterhub/lib/python3.4/site-packages (from pexpect>=3.3->bash_kernel)
Installing collected packages: bash-kernel
Successfully installed bash-kernel-0.4.1
(jupyterhub) [joan5896@jupyter2 ~]$ \
> JUPYTER_DATA_DIR=/opt/jupyterhub/share/jupyter \
> python -m bash_kernel.install
Installing IPython kernel spec
(jupyterhub) [joan5896@jupyter2 ~]$ █
```

```
joan5896 — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc +
```

```
(jupyterhub) [joan5896@jupyter2 ~]$ pip install bash_kernel
Collecting bash_kernel
  Downloading bash_kernel-0.4.1-py2.py3-none-any.whl
Requirement already satisfied (use --upgrade to upgrade): pexpect>=3.3 in /opt/jupyterhub/lib/python3.4/site-packages (from bash_kernel)
Requirement already satisfied (use --upgrade to upgrade): ptyprocess>=0.5 in /opt/jupyterhub/lib/python3.4/site-packages (from pexpect>=3.3->bash_kernel)
Installing collected packages: bash-kernel
Successfully installed bash-kernel-0.4.1
(jupyterhub) [joan5896@jupyter2 ~]$ \
> JUPYTER_DATA_DIR=/opt/jupyterhub/share/jupyter \
> python -m bash_kernel.install
Installing IPython kernel spec
(jupyterhub) [joan5896@jupyter2 ~]$ ls -R /opt/jupyterhub/share/jupyter/kernels
/opt/jupyterhub/share/jupyter/kernels:
bash

/opt/jupyterhub/share/jupyter/kernels/bash:
kernel.json
(jupyterhub) [joan5896@jupyter2 ~]$ cat /opt/jupyterhub/share/jupyter/kernels/bash/kernel.json
{"argv": ["/opt/jupyterhub/bin/python3", "-m", "bash_kernel", "-f", "{connection_file}"], "codemirror_mode": "shell", "display_name": "Bash", "env": {"PS1": "$"}, "language": "bash"}(jupyterhub) [joan5896@jupyter2 ~]$
```

joan5896 — joan5896@jupyter2:~ — ssh \* ssh joan5896@jupyter2.rc — 80x24  
~ — joan5896@jupyter2:~ — ssh \* ssh joan5896@jupyter2.rc

```
(jupyterhub) [joan5896@jupyter2 ~]$ sudo systemctl restart jupyterhub  
(jupyterhub) [joan5896@jupyter2 ~]$
```

Jupyter/ +

https://jupyter2.rc.int.colorado.edu/user/joan5896/tree/Jupyter

Search

Control Panel Logout

Files Running IPython Clusters

Select items to perform actions on them.

Upload New 

- Text File
- Folder
- Terminal
- Notebooks
- Bash
- Python 3

Home / Jupyter

- ..
- Bash.ipynb
- MPI.ipynb
- Parallel.ipynb
- Python2.ipynb
- Python3.ipynb
- R.ipynb

https://jupyter2.rc.int.colorado.edu/user/joan5896/tree/Jupyter#

The screenshot shows a Jupyter Notebook interface running a Bash kernel. The top navigation bar includes tabs for 'Jupyter' and 'Bash', and a search bar. The main area displays a series of terminal-like input cells and their outputs:

- In [1]:** hostname  
jupyter2.rc.int.colorado.edu
- In [3]:** whoami  
joan5896
- In [4]:** ls  
Bash.ipynb Parallel.ipynb Python2.ipynb Python3.ipynb R.ipynb
- In [5]:** ls ~  
batch\_script nc110.tgz  
bin netcat-0.7.1  
Changelog netcat-0.7.1.tar.gz  
curc-bench netcat.blurb  
curc-bench-runs netcat.c  
curc-puppet Pictures  
data Projects  
Desktop Public

# IRKernel



irkernel.github.io

cripps.local | Syncthing Summit 16 Abandon... Goosebumps Go ▾ Free Music Archive + Pocket CU ▾ CURC ▾ civilfritz ▾ Topics ▾ Git tutorial ▾ How To Make ...ular Science here

IRkernel

REQUIREMENTS → INSTALLATION → RUNNING → FAQ

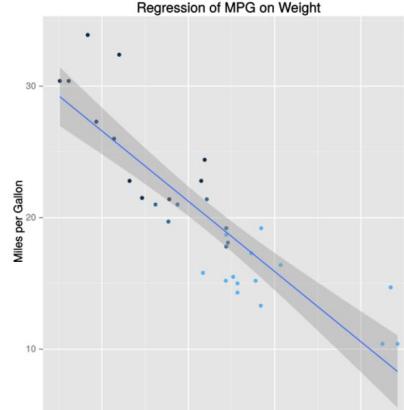
# R kernel for Jupyter



In [1]: `library(ggplot2)`

In [2]: `ggplot(mtcars, aes(x=wt, y=mpg)) + geom_point() + geom_smooth()`

Regression of MPG on Weight



Miles per Gallon

cyl

In [3]: `mtcars %>% group_by(cyl) %>% summarise(mpg = mean(mpg))`

irkernel.github.io

```
joan5896 — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc
[joan5896@jupyter2 ~]$ sudo yum -y install R
```

irkernel.github.io/installation/

cripps.local | Syncthing Summit 16 Abandoned Goosebumps Go ↴ Free Music Archive + Pocket CU ↴ CURC ↴ civilfritz ↴ Topics ↴ Git tutorial ↴ How To Make ...ular Science here

IRkernel

IRkernel IRdisplay repr

REQUIREMENTS → INSTALLATION → RUNNING → FAQ

# Installation

BINARY SOURCE DEVELOPMENT

## Installing via supplied binary packages (default on Windows + Mac OS X)

We are currently in the process of submitting our packages to CRAN. Until then, you can install via devtools with the following lines in an R console:

```
install.packages(c('repr', 'pbdZMQ', 'devtools')) # repr is already on CRAN
devtools::install_github('IRkernel/IRdisplay')
devtools::install_github('IRkernel/IRkernel')
IRkernel::installspec()
```

To update packages which are not yet on CRAN, you have to rerun `devtools::install_github(...)` lines.

## Making the kernel available to Jupyter

If this fails, search the output for something like:

```
** testing if installed package can be loaded
Error in dyn.load(file, DLLpath = DLLpath, . 
  unable to load shared object '/Users/[you]/
dlopen(/Users/[you]/anaconda/lib/R/library
Referenced from: /Users/[you]/anaconda/lib
Reason: image not found
```

This would mean that the R binary package of `rzmq` was compiled against a different version of `libzmq` from the one on your system.  
It can be fixed by installing from source (click the "Source" tab)

irkernel.github.io/installation/

```
joan5896 — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc
[joan5896@jupyter2 ~]$ sudo yum -y install zeromq zeromq-devel
```

```
joan5896 — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc
[joan5896@jupyter2 ~]$ sudo yum -y install openssl-devel libcurl-devel
```

```
joan5896 — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc
[[joan5896@jupyter2 ~]$ mkdir -p /opt/jupyterhub/lib/R
[[joan5896@jupyter2 ~]$ export R_LIBS=/opt/jupyterhub/lib/R
[joan5896@jupyter2 ~]$ R -e "install.packages( \
>   c('repr', 'pbZIP4', 'devtools'), \
>   repos=c('https://cloud.r-project.org/'), \
>   lib='/opt/jupyterhub/lib/R')"
```

```
joan5896 — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc +
```

```
unload                                html
update_packages                         html
use_build_ignore                        html
use_data                                html
use_data_raw                            html
use_git                                 html
use_git_hook                           html
use_github                             html
use_github_links                        html
use_news_md                            html
use_package                            html
use_readme_rmd                          html
wd                                     html
with_debug                             html
** building package indices
** installing vignettes
** testing if installed package can be loaded
* DONE (devtools)

The downloaded source packages are in
  '/tmp/RtmpGUWg3K/downloaded_packages'
>
>
[joan5896@jupyter2 ~]$
```

```
joan5896 — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc
[joan5896@jupyter2 ~]$ R -e "devtools::install_github('IRkernel/IRdisplay', \
>   lib='/opt/jupyterhub/lib/R')"
```

```
joan5896 — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc
[joan5896@jupyter2 ~]$ R -e "devtools::install_github('IRkernel/IRkernel', \
>   lib='/opt/jupyterhub/lib/R')"
```

```
joan5896 — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc
+  
--library='/opt/jupyterhub/lib/R' --install-tests  
  
* installing *source* package 'IRkernel' ...  
** R  
** inst  
** tests  
** preparing package for lazy loading  
** help  
*** installing help indices  
  converting help for package 'IRkernel'  
    finding HTML links ... done  
    Comm-class                      html  
    Comm_Manager-class               html  
    IRkernel                         html  
    Kernel-class                     html  
    comm_manager                     html  
    installspec                      html  
    main                             html  
** building package indices  
** testing if installed package can be loaded  
* DONE (IRkernel)  
>  
>  
[joan5896@jupyter2 ~]$
```

```
joan5896 — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc
[joan5896@jupyter2 ~]$ module load intel mkl python/3.4.3
[joan5896@jupyter2 ~]$ source /opt/jupyterhub/bin/activate
(jupyterhub) [joan5896@jupyter2 ~]$ \
> JUPYTER_DATA_DIR=/opt/jupyterhub/share/jupyter \
> R -e "IRkernel::installspec()"
```

```
joan5896 — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc
(jupyterhub) [joan5896@jupyter2 ~]$ cat /opt/jupyterhub/share/jupyter/kernels/r/kernel.json
{
    "argv": ["/usr/lib64/R/bin/R", "--slave", "-e", "IRkernel::main()", "--args",
    "{connection_file}"],
    "display_name": "R",
    "language": "R"
}
(jupyterhub) [joan5896@jupyter2 ~]$
```

```
joan5896 — joan5896@jupyter2:~ — ssh ✘ ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh ✘ ssh joan5896@jupyter2.rc

#!/bin/bash

function main
{
    source /etc/profile.d/modules.sh
    module load intel mkl python/3.4.3 nodejs
    source /opt/jupyterhub/bin/activate
    export R_LIBS=/opt/jupyterhub/lib/R
    exec jupyterhub "$@"
}

main "$@"
~
~
~
~
~
~
~
~
~
~
~/usr/local/bin/jupyterhub" 14L, 232C written
```

```
joan5896 — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc
# https://jupyterhub.readthedocs.io/en/latest/getting-started.html#configuration
#-file

c.JupyterHub.ip = '0.0.0.0'
c.JupyterHub.port = 443

c.JupyterHub.hub_ip = 'jupyter2.rc.int.colorado.edu'

c.Spawner.env_keep.extend(['LD_LIBRARY_PATH', 'R_LIBS'])

c.JupyterHub.cookie_secret_file = '/opt/jupyterhub/var/jupyterhub_cookie_secret'

c.JupyterHub.db_url = '/opt/jupyterhub/var/jupyterhub.sqlite'

c.JupyterHub.ssl_cert = '/opt/jupyterhub/etc/jupyter.rc.int.colorado.edu.crt'
c.JupyterHub.ssl_key = '/opt/jupyterhub/etc/jupyter.rc.int.colorado.edu.key'

c.PAMAuthenticator.service = 'jupyterhub'

c.Authenticator.admin_users = { 'holtat', 'joan5896' }

~/
~/
~/

"/opt/jupyterhub/etc/jupyterhub.py" 19L, 651C written
```

Jupyter/ +

https://jupyter2.rc.int.colorado.edu/user/joan5896/tree/Jupyter

Search

Control Panel Logout

# jupyter

Files Running IPython Clusters

Select items to perform actions on them.

Upload New 

- ..
- [Bash.ipynb](#)
- [MPI.ipynb](#)
- [Parallel.ipynb](#)
- [Python2.ipynb](#)
- [Python3.ipynb](#)
- [R.ipynb](#)

[Text File](#)

[Folder](#)

[Terminal](#)

[Notebooks](#)

[Bash](#)

[Python 3](#)

[R](#)

https://jupyter2.rc.int.colorado.edu/user/joan5896/tree/Jupyter#

Jupyter/ R

https://jupyter2.rc.int.colorado.edu/user/joan5896/notebooks/Jupyter/R.ipynb

Search Control Panel Logout

jupyter R (autosaved)

File Edit View Insert Cell Kernel Help

CellToolbar

In [1]: `sprintf("Hello, world!");`

'Hello, world!'

In [2]: `sessionInfo();`

```
R version 3.3.0 (2016-05-03)
Platform: x86_64-redhat-linux-gnu (64-bit)
Running under: Red Hat Enterprise Linux

locale:
[1] LC_CTYPE=en_US.UTF-8      LC_NUMERIC=C
[3] LC_TIME=en_US.UTF-8       LC_COLLATE=en_US.UTF-8
[5] LC_MONETARY=en_US.UTF-8   LC_MESSAGES=en_US.UTF-8
[7] LC_PAPER=en_US.UTF-8     LC_NAME=C
[9] LC_ADDRESS=C              LC_TELEPHONE=C
[11] LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C

attached base packages:
[1] stats      graphics   grDevices utils      datasets   methods    base

loaded via a namespace (and not attached):
```

IPython[2] kernel

IP[y]:  
IPython

```
joan5896 — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc +  
[joan5896@jupyter2 ~]$ module load intel mkl impi python/2.7.10  
[joan5896@jupyter2 ~]$ sudo mkdir /opt/ipykernel  
[joan5896@jupyter2 ~]$ sudo chgrp rcopsgrp /opt/ipykernel  
[joan5896@jupyter2 ~]$ sudo chmod g+w /opt/ipykernel  
[joan5896@jupyter2 ~]$ virtualenv /opt/ipykernel --system-site-packages  
New python executable in /opt/ipykernel/bin/python  
Installing setuptools, pip, wheel...done.  
[joan5896@jupyter2 ~]$
```

```
joan5896 — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc
[joan5896@jupyter2 ~]$ source /opt/ipykernel/bin/activate
ipykernel [joan5896@jupyter2 ~]$ pip install ipykernel
```

```
joan5896 — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc
[joan5896@jupyter2 ~]$ ipython kernel install --name python2 \
> --prefix /opt/jupyterhub
```

```
joan5896 — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc
[joan5896@jupyter2 ~]$ cat /opt/jupyterhub/share/jupyter/kernels/python2/kernel.json
{
  "display_name": "Python 2",
  "language": "python",
  "argv": [
    "/opt/ipykernel/bin/python",
    "-m",
    "ipykernel",
    "-f",
    "{connection_file}"
  ]
}[joan5896@jupyter2 ~]$
```

Jupyter/

https://jupyter2.rc.int.colorado.edu/user/joan5896/tree/Jupyter#

# jupyter

Control Panel Logout

Files Running IPython Clusters

Select items to perform actions on them.

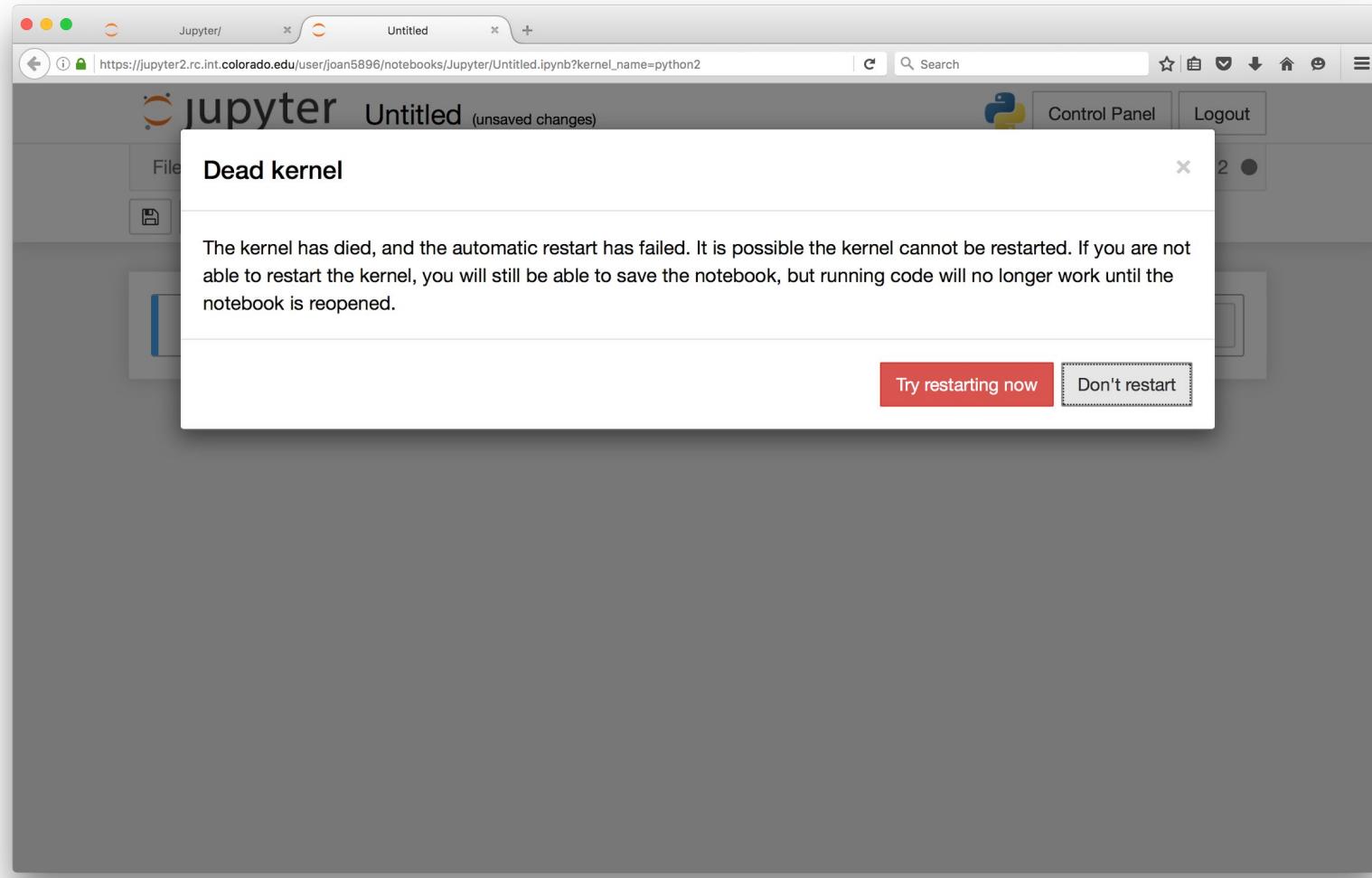
Upload New

- / Jupyter
- ..
- Bash.ipynb
- MPI.ipynb
- Parallel.ipynb
- Python2.ipynb
- Python3.ipynb
- R.ipynb

- Text File
- Folder
- Terminal

---

- Notebooks
- Bash
- Python 2
- Python 3
- R



```
joan5896 — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc
[joan5896@jupyter2 ~]$ cat /opt/jupyterhub/share/jupyter/kernels/python2/kernel.json
{
  "display_name": "Python 2",
  "language": "python",
  "argv": [
    "/opt/ipykernel/bin/python",
    "-m",
    "ipykernel",
    "-f",
    "{connection_file}"
  ]
}[joan5896@jupyter2 ~]$
```

```
joan5896 — joan5896@jupyter2:~ — ssh ✘ ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh ✘ ssh joan5896@jupyter2.rc
+  
#!/bin/bash  
  
function main  
{  
    source /etc/profile.d/modules.sh  
    module load intel mkl python/2.7.10  
    source /opt/ipykernel/bin/activate  
    exec python2.7 "$@"  
}  
  
main "$@"  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~/opt/jupyterhub/bin/python2.7" 11L, 182C written
```

```
joan5896 — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc +
```

```
{  
    "display_name": "Python 2",  
    "language": "python",  
    "argv": ["/opt/jupyterhub/bin/python2.7", "-m", "ipykernel", "-f", "{connection_file}"]}  
}  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~/opt/jupyterhub/share/jupyter/kernels/python2/kernel.json" 11L, 164C
```

The screenshot shows a Jupyter Notebook interface running in a web browser. The title bar indicates it's a Python 2 notebook. The toolbar includes standard file operations like File, Edit, View, Insert, Cell, Kernel, and Help, along with a CellToolbar dropdown.

The notebook contains three cells:

- In [1]:** `import sys  
print sys.version`  
Output:  
2.7.10 (default, Mar 9 2016, 13:20:31)  
[GCC Intel(R) C++ gcc 4.4 mode]
- In [2]:** `print "Look, ma! No parenthesis!"`  
Output:  
Look, ma! No parenthesis!
- In [ ]:** (empty cell)

Even more kernels!

GitHub, Inc. [github.com/ipython/ipython/wiki/IPython-kernels-for-other-languages](https://github.com/ipython/ipython/wiki/IPython-kernels-for-other-languages)

This repository Search Pull requests Issues Gist Watch 720 Star 9,469 Fork 2,782

Code Issues Pull requests Wiki Pulse Graphs

# IPython kernels for other languages

Evan Hubinger edited this page 9 days ago · 107 revisions

## IPython/Jupyter kernels:

The Kernel Zero, is of course [IPython](#), which you can get through [ipykernel](#), and still comes (for now) as a dependency of [jupyter](#). The IPython kernel can be thought as a reference implementation, here are other available kernels:

Name	Jupyter/IPython Version	Language(s) Version	3rd party dependencies	Example Notebooks
sas_kernel	Jupyter 4.0	python >= 3.3	SAS 9.4 or higher	
IPyKernel	Jupyter 4.0	python 2.7, >= 3.3	pymzmq	
IJulia		julia >= 0.3		
IHaskell		ghc >= 7.6		
IRuby		ruby >= 2.1		

Pages 139

Find a Page...

Home

A gallery of interesting IPython Notebooks

Code blocks and other ideas

Cookbook: Branding the IPython notebook

Cookbook: Connecting to a remote kernel via ssh

Cookbook: Dated logging

Cookbook: deploying ipython

Cookbook: Dynamic prompt

Cookbook: Hotkeys for faster notebook editing

Cookbook: Ignoring some

<https://github.com/ipython/ipython/wiki/IPython-kernels-for-other-languages>

cripps.local | Syncthing Summit 16 Abandon... Goosebumps Go ▾ Free Music Archive + Pocket CU ▾ CURC ▾ civilfritz ▾ Topics ▾ Git tutorial ▾ How To Make ...ular Science here

# IP[y]: IPython Interactive Computing

home | search | documentation » IPython developer's guide » previous | next | modules | index

**Warning**

This documentation is for an old version of IPython. You can find docs for newer versions [here](#).

## Making kernels for IPython

A 'kernel' is a program that runs and introspects the user's code. IPython includes a kernel for Python code, and people have written kernels for [several other languages](#).

When IPython starts a kernel, it passes it a connection file. This specifies how to set up communications with the frontend.

There are two options for writing a kernel:

1. You can reuse the IPython kernel machinery to handle the communications, and just describe how to execute your code. This is much simpler if the target language can be driven from Python. See [Making simple Python wrapper kernels](#) for details.
2. You can implement the kernel machinery in your target language. This is more work initially, but the people using your kernel might be more likely to contribute to it if it's in the language they know.

### Connection files

Your kernel will be given the path to a connection file when it starts (see [Kernel specs](#) for how to specify the command line arguments for your kernel). This file, which is accessible only to the current user, will contain a JSON dictionary looking something like this:

```
{ "control_port": 50160, "shell_port": 57503, "transport": "tcp", "signature_scheme": "hmac-sha256", "stdin_port": 52597, "hb_port": 42540, "ip": "127.0.0.1", "iopub_port": 40885, "key": "a0436f6c-1916-498b-8eb9-e81ab9368e }
```

The transport, ip and five \_port fields specify five socket in the example above would be:

<https://ipython.org/ipython-doc/3/development/kernels.html>

**Table Of Contents**

[Making kernels for IPython](#)

- Connection files
- Handling messages
- Kernel specs

**Previous topic**

[Messaging in IPython](#)

**Next topic**

[Making simple Python wrapper kernels](#)

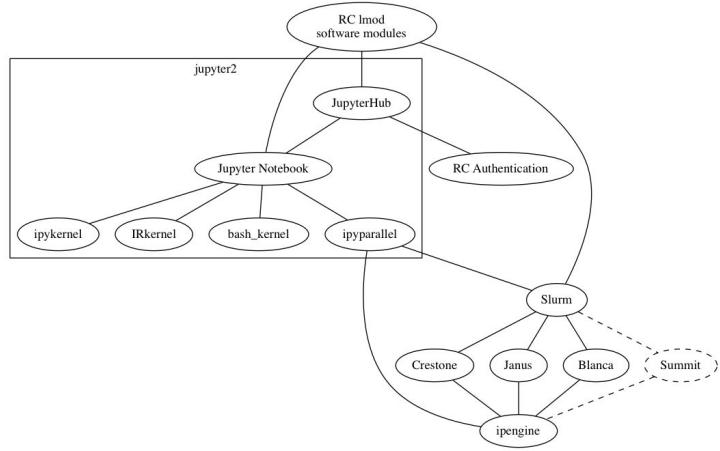
**This Page**

[Show Source](#)

**Quick search**

Enter search terms or a module, class or function name.

# ipyparallel



cripps.local | Syncthing Summit 16 Abandone... Goosebumps Go ▾ Free Music Archive + Pocket CU ▾ CURC ▾ civilfritz ▾ Topics ▾ Git tutorial ▾ How To Make ...ular Science here

# ipyparallel.readthedocs.io/en/latest/

## ipyparallel

Search docs

- Changes in IPython Parallel
- Overview and getting started
- Starting the IPython controller and engines
- IPython's Direct interface
- Parallel Magic Commands
- The IPython task interface
- TheAsyncResult object
- Using MPI with IPython
- IPython's Task Database
- Security details of IPython
- Parallel examples
- DAG Dependencies
- Details of Parallel Computing with IPython
- Transitioning from IPython.kernel to ipyparallel
- Messaging for Parallel Computing
- Connection Diagrams of The IPython ZMQ Cluster
- ipyparallel

Read the Docs v: latest ▾

## Docs » Using IPython for parallel computing

[Edit on GitHub](#)

### Note

This documentation is for a development version of IPython. There may be significant differences from the latest stable release.

# Using IPython for parallel computing

Release: 5.2.0.dev

Date: Jun 29, 2016

## Installing IPython Parallel

As of 4.0, IPython parallel is now a standalone package called `ipyparallel`. You can install it with:

```
pip install ipyparallel
```

or:

```
conda install ipyparallel
```

And if you want the IPython clusters tab extension in your Jupyter Notebook dashboard:

```
ipcluster nbextension enable
```

<http://ipyparallel.readthedocs.io/>

```
joan5896 — root@jupyter2:~ — ssh ▾ ssh joan5896@jupyter2.rc — 80x24
~ — root@jupyter2:~ — ssh ▾ ssh joan5896@jupyter2.rc
[rcops@jupyter2 ~]$ module load intel mkl python/3.4.3 zeromq
[rcops@jupyter2 ~]$ pip install ipyparallel
```

```
joan5896 — root@jupyter2:~ — ssh ▾ ssh joan5896@jupyter2.rc — 80x24
~ — root@jupyter2:~ — ssh ▾ ssh joan5896@jupyter2.rc +
```

```
Collecting pexpect (from ipython>=4->ipyparallel)
  Using cached pexpect-4.1.0-py2.py3-none-any.whl
Collecting simplegeneric>0.8 (from ipython>=4->ipyparallel)
  Downloading simplegeneric-0.8.1.zip
Collecting jupyter-core (from jupyter-client->ipyparallel)
  Using cached jupyter_core-4.1.0-py2.py3-none-any.whl
Collecting backports-abc>=0.4 (from tornado>=4->ipyparallel)
  Using cached backports_abc-0.4-py2.py3-none-any.whl
Collecting ptyprocess>=0.5 (from pexpect->ipython>=4->ipyparallel)
  Using cached ptyprocess-0.5.1-py2.py3-none-any.whl
Installing collected packages: pyzmq, ipython-genutils, setuptools, decorator, traitlets, pickleshare, ptyprocess, pexpect, simplegeneric, ipython, backports-abc, tornado, jupyter-core, jupyter-client, ipykernel, ipyparallel
  Running setup.py install for pyzmq
    Found existing installation: setuptools 12.0.5
      Uninstalling setuptools-12.0.5:
        Successfully uninstalled setuptools-12.0.5
  Running setup.py install for simplegeneric
  Running setup.py install for tornado
Successfully installed backports-abc-0.4 decorator-4.0.10 ipykernel-4.3.1 ipyparallel-5.1.1 ipython-4.2.1 ipython-genutils-0.1.0 jupyter-client-4.3.0 jupyter-core-4.1.0 pexpect-4.1.0 pickleshare-0.7.2 ptyprocess-0.5.1 pyzmq-15.2.0 setuptools-24.0.1 simplegeneric-0.8.1 tornado-4.3 traitlets-4.2.2
[rcops@jupyter2 ~]$
```

```
joan5896 — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc
[rcops@jupyter2 ~]$ module switch python/3.4.3 python/2.7.10
[rcops@jupyter2 ~]$ pip install ipyparallel
```

```
joan5896 — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc +
```

Running setup.py bdist\_wheel for backports.ssl-match-hostname ... done  
Stored in directory: /home/rcops/.cache/pip/wheels/5d/72/36/b2a31507b613967b72  
8edc33378a5ff2ada0f62855b93c5ae1  
Running setup.py bdist\_wheel for simplegeneric ... done  
Stored in directory: /home/rcops/.cache/pip/wheels/7b/31/08/c85e74c84188cbec6a  
6827beec4d640f2bd78ae003dc1ec09d  
Successfully built tornado pyzmq backports.ssl-match-hostname simplegeneric  
Installing collected packages: backports.ssl-match-hostname, singledispatch, certifi, backports-abc, tornado, decorator, ipython-genutils, traitlets, jupyter-core, pyzmq, jupyter-client, futures, pathlib2, pickleshare, simplegeneric, backports.shutil-get-terminal-size, setuptools, ptyprocess, pexpect, ipython, ipykernel, ipyparallel  
Found existing installation: setuptools 18.0.1  
Uninstalling setuptools-18.0.1:  
Successfully uninstalled setuptools-18.0.1  
Successfully installed backports-abc-0.4 backports.shutil-get-terminal-size-1.0.  
0 backports.ssl-match-hostname-3.5.0.1 certifi-2016.2.28 decorator-4.0.10 futures-3.0.5 ipykernel-4.3.1 ipyparallel-5.1.1 ipython-4.2.1 ipython-genutils-0.1.0 jupyter-client-4.3.0 jupyter-core-4.1.0 pathlib2-2.1.0 pexpect-4.2.0 pickleshare-0.7.2 ptyprocess-0.5.1 pyzmq-15.3.0 setuptools-24.0.2 simplegeneric-0.8.1 singledispatch-3.4.0.3 tornado-4.3 traitlets-4.2.2  
You are using pip version 8.0.2, however version 8.1.2 is available.  
You should consider upgrading via the 'pip install --upgrade pip' command.  
[rcops@jupyter2 ~]\$

```
joan5896 — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc
[joan5896@jupyter2 ~]$ cat ~/.jupyter/nbconfig/tree.json
{
  "load_extensions": {
    "ipyparallel/main": true
  }
}[joan5896@jupyter2 ~]$
```

```
joan5896 — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc
[joan5896@jupyter2 ~]$ cat ~/.jupyter/jupyter_notebook_config.json
{
  "NotebookApp": {
    "nbserver_extensions": {
      "ipyparallel.nbextension": true
    }
  }
}[joan5896@jupyter2 ~]$
```

Jupyter/

<https://jupyter2.rc.int.colorado.edu/user/joan5896/tree/Jupyter#ipyclusters> Search

# jupyter

Control Panel Logout

Files Running IPython Clusters

IPython parallel computing clusters

profile	status	# of engines	action
default	stopped	<input type="button" value="0"/>	<input type="button" value="Start"/>
crestone-cpu	stopped	<input type="button" value="0"/>	<input type="button" value="Start"/>
crestone-node	stopped	<input type="button" value="0"/>	<input type="button" value="Start"/>
janus-cpu	stopped	<input type="button" value="0"/>	<input type="button" value="Start"/>
janus-node	stopped	<input type="button" value="0"/>	<input type="button" value="Start"/>

The screenshot shows a Mac OS X desktop environment with a browser window open to the IPython documentation page on ReadTheDocs. The URL in the address bar is <http://ipython.readthedocs.io/en/stable/development/config.html#profiles>. The browser has a standard OS X look with red, green, and blue window control buttons.

The IPython documentation sidebar on the left includes the following sections:

- Introduction
- What's new in IPython
- Installation
- Using IPython for interactive work
- Configuration and customization
- Developer's guide for third party tools and libraries
  - How IPython works
  - Making simple Python wrapper kernels
  - Execution semantics in the IPython kernel
  - New IPython Console Lexer
  - Writing code for Python 2 and 3
- Overview of the IPython configuration system
  - Configuration file location
  - Profiles
  - IPython GUI Support Notes
- Developer's guide to core IPython
- The IPython API
- About IPython

The main content area is titled "Profiles". It defines a profile as a directory containing configuration and runtime files. It explains that users often want to maintain separate configurations for numerical computing (NumPy and SciPy) and symbolic computing (SymPy). Profiles make it easy to keep separate config files, logs, and histories for each purpose.

A code example shows the command: `$ ipython --profile=sympy`. The text explains that this tells the `ipython` command line program to get its configuration from the "sympy" profile. It notes that file names for various profiles do not change; they are distinguished by their names.

The general pattern for creating a new profile is shown as: `$ ipython profile create <name>`. This adds a directory named `<profile_<name>` to the IPython directory. You can then load this profile by adding `--profile=<name>` to your command line options.

At the bottom of the page, there is a note: "IPython extends the config loader for Python files so that you can inherit config from another".

```
joan5896 — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc + ]
```

```
[joan5896@jupyter2 ~]$ ls -d .ipython/profile_*
.ipython/profile_crestone-cpu  .ipython/profile_janus-cpu
.ipython/profile_crestone-node .ipython/profile_janus-node
.ipython/profile_default
[joan5896@jupyter2 ~]$ █
```

```
joan5896 — joan5896@jupyter2:~ — ssh + ssh joan5896@jupyter2.rc — 93x27
~ — joan5896@jupyter2:~ — ssh + ssh joan5896@jupyter2.rc

# Configuration file for ipcluster.

c.IPClusterEngines.engine_launcher_class = 'ipyparallel.apps.launcher.SlurmEngineSetLauncher'
c.SlurmLauncher.qos = 'crestone'
c.SlurmLauncher.timelimit = '4:00:00'
# c.SlurmLauncher.account = ''
# c.SlurmLauncher.machines = ''
# c.SlurmLauncher.mem = ''
# c.SlurmLauncher.resources = ''

c.SlurmEngineSetLauncher.batch_template = """#!/bin/bash

#SBATCH --qos {qos}
#SBATCH --job-name ipyparallel-crestone-cpu-engine
#SBATCH --ntasks {n}
#SBATCH --cpus-per-task 1
#SBATCH --time {timelimit}
#SBATCH --output {profile_dir}/log/slurm.out

PATH=/bin:/usr/bin:${{PATH}}
source /etc/profile.d/modules.sh
module load intel mkl python/3.4.3 impi zeromq

mpexec ipengine --profile-dir="{profile_dir}" --cluster-id="{cluster_id}"
"""

~
".ipython/profile_crestone-cpu/ipcluster_config.py" 25L, 764C written
```

```
joan5896 — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc +
```

```
[joan5896@jupyter2 ~]$ sudo firewall-cmd --zone trusted \
> --add-source 10.225.0.0/16 --permanent
success
[joan5896@jupyter2 ~]$ sudo firewall-cmd --zone trusted \
> --add-port 32768-61000/tcp --permanent
success
[joan5896@jupyter2 ~]$ sudo firewall-cmd --reload
success
[joan5896@jupyter2 ~]$ sudo firewall-cmd --zone trusted --list-all
trusted
  interfaces:
  sources: 10.225.0.0/16
  services:
  ports: 32768-61000/tcp
  masquerade: no
  forward-ports:
  icmp-blocks:
  rich rules:

[joan5896@jupyter2 ~]$
```

Jupyter/

<https://jupyter2.rc.int.colorado.edu/user/joan5896/tree/Jupyter#ipyclusters>

# jupyter

Control Panel Logout

Files Running IPython Clusters

IPython parallel computing clusters

profile	status	# of engines	action
default	stopped	<input type="button" value="0"/>	<input type="button" value="Start"/>
crestone-cpu	running	4	<input type="button" value="Stop"/>
crestone-node	stopped	<input type="button" value="0"/>	<input type="button" value="Start"/>
janus-cpu	stopped	<input type="button" value="0"/>	<input type="button" value="Start"/>
janus-node	stopped	<input type="button" value="0"/>	<input type="button" value="Start"/>

The screenshot shows a Jupyter Notebook interface running in a web browser. The title bar indicates the notebook is titled "Parallel" and was last autosaved. The top menu bar includes File, Edit, View, Insert, Cell, Kernel, and Help. A Python 3 kernel is selected. The toolbar below the menu includes icons for file operations, cell execution, and cell type selection (Markdown). The main content area displays two code cells:

**In [22]:**

```
import ipyparallel  
  
# attach to a running cluster  
cluster = ipyparallel.Client(profile='crestone-cpu')  
print('profile:', cluster.profile)  
print("IDs:", cluster.ids) # Print process id numbers
```

profile: crestone-cpu  
IDs: [0, 1, 2, 3]

**In [23]:**

```
# The %px magic executes a single Python command on  
# the engines specified by the targets attribute of  
# the DirectView instance.  
#  
# http://ipython.org/ipython-doc/stable/parallel/magics.html#px  
%px import socket  
%px print("hosts:", socket.gethostname())
```

Jupyter/ Parallel

https://jupyter2.rc.int.colorado.edu/user/jjoan5896/notebooks/Jupyter/Parallel.ipynb

jupyter Parallel (autosaved)

Control Panel Logout

File Edit View Insert Cell Kernel Help Python 3

In [23]:

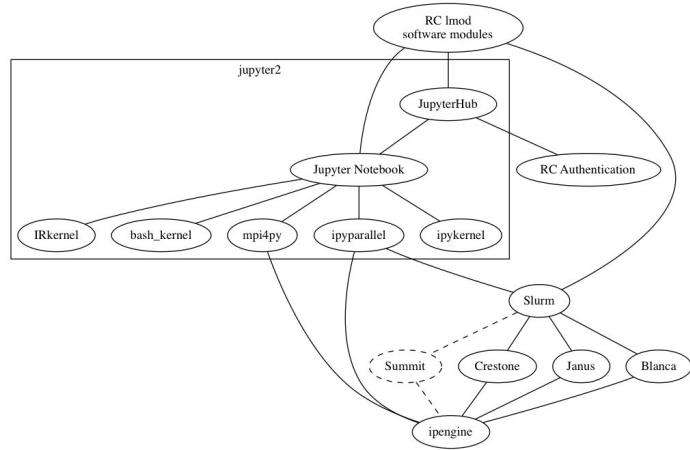
```
# The %px magic executes a single Python command on
# the engines specified by the targets attribute of
# the DirectView instance.
#
# http://ipython.org/ipython-doc/stable/parallel/magics.html#px
%px import socket
%px print("hosts:", socket.gethostname())

# calculate square numbers in parallel. Print result.
squares = cluster[:].map_sync(lambda x: x**2, range(32))
print("squares:", squares)
```

[stdout:0] hosts: cnode0114.rc.int.colorado.edu  
[stdout:1] hosts: cnode0114.rc.int.colorado.edu  
[stdout:2] hosts: cnode0114.rc.int.colorado.edu  
[stdout:3] hosts: cnode0114.rc.int.colorado.edu  
squares: [0, 1, 4, 9, 16, 25, 36, 49, 64, 81, 100, 121, 144, 169, 196, 225, 256, 289, 324, 361, 400, 441, 484, 529, 576, 625, 676, 729, 784, 841, 900, 961]

In [ ]:

# mpi4py



```
joan5896 — root@jupyter2:~ — ssh ▾ ssh joan5896@jupyter2.rc — 80x24
~ — root@jupyter2:~ — ssh ▾ ssh joan5896@jupyter2.rc
[rcops@jupyter2 ~]$ module load intel mkl impi python/3.4.3
[rcops@jupyter2 ~]$ pip install mpi4py
```

```
joan5896 — root@jupyter2:~ — ssh ▾ ssh joan5896@jupyter2.rc — 80x24
~ — root@jupyter2:~ — ssh ▾ ssh joan5896@jupyter2.rc +
```

```
[rcops@jupyter2 ~]$ module load intel mkl impi python/3.4.3
[rcops@jupyter2 ~]$ pip install mpi4py
You are using pip version 7.1.0, however version 8.1.2 is available.
You should consider upgrading via the 'pip install --upgrade pip' command.
Collecting mpi4py
  Downloading mpi4py-2.0.0.tar.gz (1.3MB)
    100% |██████████| 1.3MB 199kB/s
Installing collected packages: mpi4py
  Running setup.py install for mpi4py
Successfully installed mpi4py-2.0.0
[rcops@jupyter2 ~]$
```

```
joan5896 — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc
[rcops@jupyter2 ~]$ module switch python/3.4.3 python/2.7.10
[rcops@jupyter2 ~]$ pip install mpi4py
```

```
joan5896 — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc +
```

```
[rcops@jupyter2 ~]$ module switch python/3.4.3 python/2.7.10
[rcops@jupyter2 ~]$ pip install mpi4py
Collecting mpi4py
  Downloading mpi4py-2.0.0.tar.gz (1.3MB)
    100% |██████████| 1.3MB 154kB/s
Building wheels for collected packages: mpi4py
  Running setup.py bdist_wheel for mpi4py ... done
  Stored in directory: /home/rcops/.cache/pip/wheels/e4/54/93/c216e777e447e0261d
961c6480f5cb0d8284e7aa1920df8fc0
Successfully built mpi4py
Installing collected packages: mpi4py
Successfully installed mpi4py-2.0.0
You are using pip version 8.0.2, however version 8.1.2 is available.
You should consider upgrading via the 'pip install --upgrade pip' command.
[rcops@jupyter2 ~]$
```

Jupyter MPI (autosaved)

Control Panel Logout

File Edit View Insert Cell Kernel Help Python 3

CellToolbar

## An example Jupyter notebook for parallel programming using MPI

```
In [33]: import ipyparallel  
  
# attach to a running cluster  
cluster = ipyparallel.Client(profile='crestone-cpu')  
print('profile:', cluster.profile)  
print("IDs:", cluster.ids) # Print process id numbers
```

```
profile: crestone-cpu  
IDs: [0, 1, 2, 3]
```

```
In [34]: %%px  
  
# %%px can be used as a Cell Magic, which accepts  
# some arguments for controlling the execution.  
#  
# http://ipyparallel.readthedocs.org/en/latest/magics.html#px-cell-magic
```

The screenshot shows a Jupyter Notebook interface running on a Mac OS X system. The browser tab is titled "Jupyter/MPI". The notebook URL is <https://jupyter2.rc.int.colorado.edu/user/jjoan5896/notebooks/Jupyter/MPI.ipynb>. The top navigation bar includes "Control Panel" and "Logout" buttons.

The notebook has a Python 3 kernel selected. The toolbar includes standard file operations (File, Edit, View, Insert, Cell, Kernel, Help) and a CellToolbar with various icons for cell management.

Cell 34 (In [34]):

```
%px

# %%px can be used as a Cell Magic, which accepts
# some arguments for controlling the execution.
#
# http://ipyparallel.readthedocs.org/en/latest/magics.html#px-cell-magic

import mpi4py.MPI

comm = mpi4py.MPI.COMM_WORLD
rank = comm.Get_rank()
print(rank)

[stdout:0] 3
[stdout:1] 2
[stdout:2] 1
[stdout:3] 0
```

Cell 35 (In [35]):

```
%%px

from mpi4py import MPI
import numpy

comm = MPI.COMM_WORLD
```

Jupyter/ MPI

https://jupyter2.rc.int.colorado.edu/user/jjoan5896/notebooks/Jupyter/MPI.ipynb

# jupyter MPI (autosaved)

Control Panel Logout

File Edit View Insert Cell Kernel Help Python 3

In [35]: %%px

```
from mpi4py import MPI
import numpy

comm = MPI.COMM_WORLD
rank = comm.Get_rank()

# passing MPI datatypes explicitly
if rank == 0:
    data = numpy.arange(100, dtype='i')
    numpy.random.shuffle(data)
    comm.Send([data, MPI.INT], dest=1, tag=77)
    print("{0}: sent data to 1: {1}".format(rank, data))
elif rank == 1:
    data = numpy.empty(100, dtype='i')
    comm.Recv([data, MPI.INT], source=0, tag=77)
    print("{0}: received data from 0: {1}".format(rank, data))
else:
    print("{0}: idle".format(rank))

[stdout:0] 3: idle
[stdout:1] 2: idle
[stdout:2]
1: received data from 0: [18 99 91 30 87 61 79 62 81 78 19 69 75 90 53 82
```

Jupyter MPI (autosaved)

Control Panel Logout

File Edit View Insert Cell Kernel Help Python 3

CellToolbar

```
comm.recv([data, MPI.INT], source=0, tag=11)
print("{0}: received data from 0: {1}".format(rank, data))
else:
    print("{0}: idle".format(rank))

[stdout:0] 3: idle
[stdout:1] 2: idle
[stdout:2]
1: received data from 0: [18 99 91 30 87 61 79 62 81 78 19 69 75 90 53 82
 28 98 94 3 72 71 5 51 65
 49 24 86 74 54 44 10 12 50 46 40 93 32 26 11 4 68 37 76 92 57 13 20 33
31
 8 58 0 52 60 96 77 73 80 42 85 29 67 43 15 14 55 84 36 59 48 16 27 89
34
 25 39 88 6 41 63 22 21 9 47 64 95 2 83 97 23 45 1 56 66 35 7 70 17
38]
[stdout:3]
0: sent data to 1: [18 99 91 30 87 61 79 62 81 78 19 69 75 90 53 82 28 98
 94 3 72 71 5 51 65
 49 24 86 74 54 44 10 12 50 46 40 93 32 26 11 4 68 37 76 92 57 13 20 33
31
 8 58 0 52 60 96 77 73 80 42 85 29 67 43 15 14 55 84 36 59 48 16 27 89
34
 25 39 88 6 41 63 22 21 9 47 64 95 2 83 97 23 45 1 56 66 35 7 70 17
38]
```

# User initialization

```
joan5896 — joan5896@jupyter2:~ — ssh ✧ ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh ✧ ssh joan5896@jupyter2.rc

#!/bin/bash

function main
{
    /curc/tools/jupyterhub/libexec/jupyter_init.sh
    /curc/tools/jupyterhub/libexec/ipython_init.sh
    exec jupyterhub-singleuser "$@"
}

main "$@"
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~/opt/jupyterhub/bin/jupyterhub-singleuser-autoinit" 10L, 180C
```

```
joan5896 — joan5896@jupyter2:~ — ssh ▾ ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh ▾ ssh joan5896@jupyter2.rc
+
# https://jupyterhub.readthedocs.io/en/latest/getting-started.html#configuration
#-file

c.JupyterHub.ip = '0.0.0.0'
c.JupyterHub.port = 443

c.JupyterHub.hub_ip = 'jupyter2.rc.int.colorado.edu'

c.Spawner.env_keep.extend(['LD_LIBRARY_PATH', 'R_LIBS'])

c.JupyterHub.cookie_secret_file = '/opt/jupyterhub/var/jupyterhub_cookie_secret'

c.JupyterHub.db_url = '/opt/jupyterhub/var/jupyterhub.sqlite'

c.JupyterHub.ssl_cert = '/opt/jupyterhub/etc/jupyter.rc.int.colorado.edu.crt'
c.JupyterHub.ssl_key = '/opt/jupyterhub/etc/jupyter.rc.int.colorado.edu.key'

c.PAMAuthenticator.service = 'jupyterhub'

c.Authenticator.admin_users = { 'holtat', 'joan5896' }

c.Spawner.cmd = ['jupyterhub-singleuser-autoinit']
^
"/opt/jupyterhub/etc/jupyterhub.py" 21L, 703C
```

selinux

```
[joan5896@jupyter2 ~]$ sudo ausearch --message avc --start recent --interpret
-----
type=SYSCALL msg=audit(06/28/2016 14:18:54.825:4387) : arch=x86_64 syscall=execve success=no exit=-13(Permission denied) a0=0x1c758f0 a1=0x1f164a0 a2=0x1cec7f0 a3=0x1 items=0 ppid=30592 pid=30874 auid=joan5896 uid=joan5896 gid=joan5896pgrp euid=joan5896 suid=joan5896 fsuid=joan5896 egid=joan5896pgrp sgid=joan5896pgrp fsgid=joan5896pgrp tty=(none) ses=28 comm=jupyterhub exe=/opt/jupyterhub/bin/python3.4 subj=system_u:system_r:unconfined_service_t:s0 key=(null)
type=AVC msg=audit(06/28/2016 14:18:54.825:4387) : avc: denied { transition } for pid=30874 comm=jupyterhub path=/opt/jupyterhub/bin/jupyterhub-singleuser dev="sda2" ino=35957599 scontext=system_u:system_r:unconfined_service_t:s0 tcontext=u
nconfined_u:unconfined_r:unconfined_t:s0 tclass=process
[joan5896@jupyter2 ~]$ sudo setenforce permissive
[joan5896@jupyter2 ~]$
```

```
joan5896 — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc
[joan5896@jupyter2 ~]$ sudo ausearch --message avc --start today --raw \
> | audit2allow

=====
unconfined_service_t =====
allow unconfined_service_t unconfined_t:process transition;
[joan5896@jupyter2 ~]$
```

```
joan5896 — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh • ssh joan5896@jupyter2.rc +
```

```
[joan5896@jupyter2 ~]$ sudo ausearch --message avc --start today --raw \
> | audit2allow
```

```
#===== unconfined_service_t =====
allow unconfined_service_t unconfined_t:process transition;
```

```
[joan5896@jupyter2 ~]$ sudo ausearch --message avc --start today --raw \
> | audit2allow -M curc-jupyterhub
```

```
joan5896 — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc +
```

```
[joan5896@jupyter2 ~]$ sudo ausearch --message avc --start today --raw \
> | audit2allow

===== unconfined_service_t =====
allow unconfined_service_t unconfined_t:process transition;
[joan5896@jupyter2 ~]$ sudo ausearch --message avc --start today --raw \
> | audit2allow -M curc-jupyterhub
***** IMPORTANT *****
To make this policy package active, execute:

semodule -i curc-jupyterhub.pp

[joan5896@jupyter2 ~]$ sudo semodule -i curc-jupyterhub.pp
[joan5896@jupyter2 ~]$
```

```
joan5896 — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc — 80x24
~ — joan5896@jupyter2:~ — ssh * ssh joan5896@jupyter2.rc +
```

```
[joan5896@jupyter2 ~]$ sudo ausearch --message avc --start today --raw \
> | audit2allow

===== unconfined_service_t =====
allow unconfined_service_t unconfined_t:process transition;
[joan5896@jupyter2 ~]$ sudo ausearch --message avc --start today --raw \
> | audit2allow -M curc-jupyterhub
***** IMPORTANT *****
To make this policy package active, execute:

semodule -i curc-jupyterhub.pp

[joan5896@jupyter2 ~]$ sudo semodule -i curc-jupyterhub.pp
[joan5896@jupyter2 ~]$ sudo setenforce enforcing
```

Jupyter Hub

https://jupyter2.rc.int.colorado.edu/hub/login

# jupyter

### Sign in

**Username:**

**Password:**

**Sign In**

Home Search Control Panel Logout

# jupyter

Files [Running](#) [IPython Clusters](#)

Select items to perform actions on them.

[Upload](#) [New ▾](#) [⟳](#)

<input type="checkbox"/>	
<input type="checkbox"/>	<a href="#">bak</a>
<input type="checkbox"/>	<a href="#">bin</a>
<input type="checkbox"/>	<a href="#">curc-bench</a>
<input type="checkbox"/>	<a href="#">curc-bench-runs</a>
<input type="checkbox"/>	<a href="#">curc-puppet</a>
<input type="checkbox"/>	<a href="#">data</a>
<input type="checkbox"/>	<a href="#">Desktop</a>
<input type="checkbox"/>	<a href="#">Documents</a>
<input type="checkbox"/>	<a href="#">Downloads</a>
<input type="checkbox"/>	<a href="#">intel</a>
<input type="checkbox"/>	<a href="#">Jupyter</a>
<input type="checkbox"/>	<a href="#">jupyterhub-tutorial</a>

potential improvements

# remote kernels (rk)

<https://github.com/korniichuk/rk>

Stop using lmod modules?

additional IPython Parallel cluster  
parameters

Home https://jupyter2.rc.int.colorado.edu/user/joan5896/tree#ipyclusters

Control Panel Logout

# jupyter

Files Running IPython Clusters

IPython parallel computing clusters

profile	status	# of engines	action
default	stopped		<a href="#">Start</a>
crestone-cpu	stopped		<a href="#">Start</a>
crestone-node	stopped		<a href="#">Start</a>
janus-cpu	stopped		<a href="#">Start</a>
janus-node	stopped		<a href="#">Start</a>

additional IPython Parallel cluster status  
detail

Home https://jupyter2.rc.int.colorado.edu/user/joan5896/tree#ipyclusters Search

# jupyter

Control Panel Logout

Files Running IPython Clusters

IPython parallel computing clusters

profile	status	# of engines	action
default	stopped		
crestone-cpu	running	12	
crestone-node	stopped		
janus-cpu	stopped		
janus-node	stopped		

navigation above the initial working  
directory

Home <https://jupyter2.rc.int.colorado.edu/user/joan5896/tree#notebooks> Search

# jupyter

Control Panel Logout

Files Running IPython Clusters

Select items to perform actions on them.

bin

curc-bench

curc-bench-runs

curc-puppet

data

Desktop

Documents

Downloads

intel

Jupyter

Upload New

run JupyterHub with reduced privileges

# cgroups for single-user notebook servers

# Contact us! Collaborate with us!

- Thomas Hauser
  - [thomas.hauser@colorado.edu](mailto:thomas.hauser@colorado.edu)
- Jonathon Anderson
  - [jonathon.anderson@colorado.edu](mailto:jonathon.anderson@colorado.edu)