

Getting Started with OmniSci resources on Harvard's Cannon Computation Cluster

Contents

Introduction	1
Getting started	1
Start OmniSci or PostGIS instance on FASRC	2
Create VPN Tunnel to new Instance	4
Accessing OmniSci Immerse from your browser via the tunnel	8
Running scripts	10
Useful Links	11

Introduction

This work was funded by OmniSci Technologies and by NSF grants #1841403 and 2027540.

This is a guide for researchers interested in using OmniSci and PostGIS resources on Harvards' Cluster. Additional details are available here: https://github.com/cga-harvard/GIS_Apps_on_HPC/wiki.

These instructions are Windows oriented and use Putty. Linux users should not have trouble applying these.

Note: In addition to the tools CGA has installed, OmniSci and PostGIS, the cluster has a wide range of powerful tools available: <https://portal.rc.fas.harvard.edu/p3/build-reports/>.

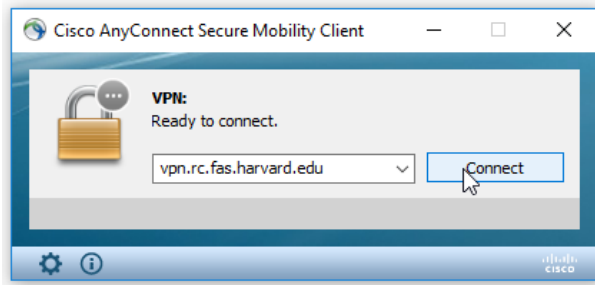
Getting started

You will need a Harvard account to start. Please follow instructions here for obtaining an FASRC account:

<https://docs.rc.fas.harvard.edu/kb/how-do-i-get-a-research-computing-account/>

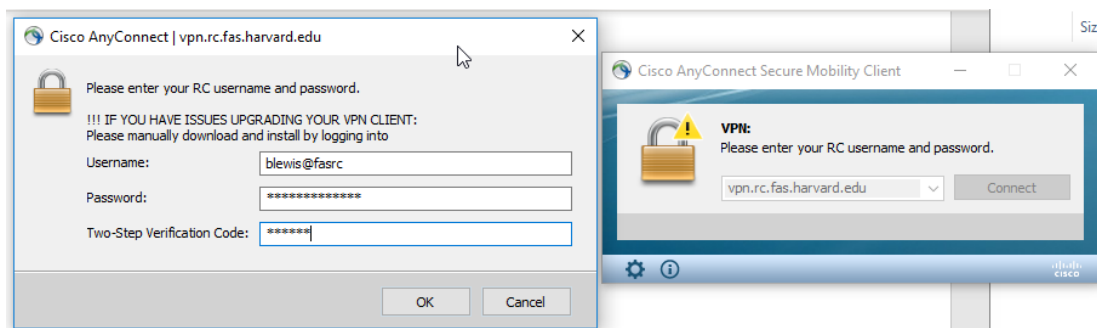
Once you have a Harvard account (using Globus it may be possible to provide access without a Harvard account):

Open VPN connection to FASRC



Use FASRC username, password, and two-step verification code.

Note user name is your-user-name@fasrc.



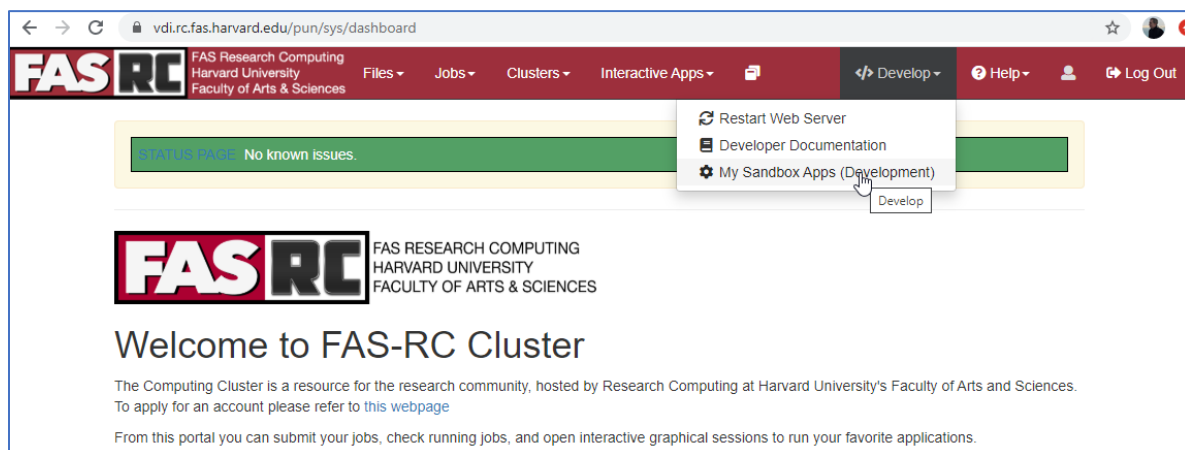
Start OmniSci or PostGIS instance on FASRC

Enter this URL from your browser, rather than clicking on link.

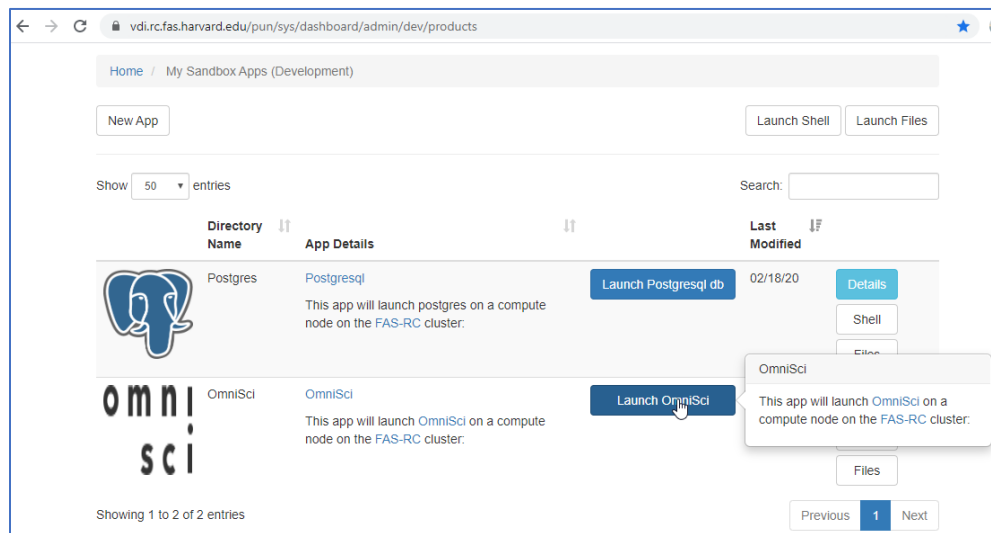
<https://vdi.rc.fas.harvard.edu/>.

If you are prompted for a user name and password, use your base FASRC user name, in this example it would be “blewis” and enter the same password you used for making your VPN connection.

You will be redirected to this page. Go to “My Sandbox Apps”.



Which is this URL: <https://vdi.rc.fas.harvard.edu/pun/sys/dashboard/admin/dev/products>
Where you will see two options, Postgres and OmniSci.



To start an OmniSci instance, click “Launch OmniSci” to start new server running OmniSci.

A screenshot of the 'OmniSci' configuration page. The page title is 'OmniSci'. Below the title, it says 'This app will launch OmniSci on a compute node on the FAS-RC cluster:'. The page has several configuration fields: 'Partition' (set to 'gpu_test'), 'Memory Allocation in GB' (set to '8'), 'Number of cores' (set to '1'), 'Number of GPUs' (set to '1'), 'Allocated Time (expressed in MM , or HH:MM:SS , or DD-HH:MM)' (set to '8:00:00'), and 'location to map omnisci-storage' (set to '/n/holyscratch01/cga/\$USER/\$SLURM_JOB_ID'). On the left side, there is a sidebar with a list of 'Interactive Apps' including 'Desktops', 'FAS-RC Remote Visualization', 'FAS-RC Remote Desktop', 'Containerized FAS-RC Remote Desktop', 'FAS CGA', 'OmniSci', 'Postgresql db', 'FAS Informatics', 'Jupyter Lab (scipy-notebook)', 'RStudio Server (Bioconductor + tidyverse)', 'GUIs', 'Desktop Environment for Totalview', 'Matlab', 'SAS', 'Stata', 'Servers', and 'JBrowse'.

For most cases use the “gpu_test” partition.

DO NOT USE THE INTERACTIVE APPS DEFAULT “fas_gpu” partition unless you have already tested your whole work flow and know that you need to allocate the resources for up to a week (168 hours). This will take credits from the CGA account and we only want to use it when we need to.

The instance generally takes a few minutes to launch. Once it is launched, go here to see your instance(s) running:

https://vdi.rc.fas.harvard.edu/pun/sys/dashboard/batch_connect/sessions

OmniSci (51411901) 1 node | 4 cores | Running

Host: `>_aagk80gpu52.rc.fas.harvard.edu` Delete

Created at: 2020-04-06 11:07:36 EDT

Time Remaining: 167 hours and 57 minutes

Session ID: 9f867037-aea8-435c-b049-6d184ed392c1

Connect to Omnisci

For the time being the proxy does not work for this application.
You can connect by tunneling via the login nodes:

1. `ssh -NL 8192:aagk80gpu52.rc.fas.harvard.edu:8192 blewis@login.rc.fas.harvard.edu`
2. open <http://localhost:8192> in your browser link

Note your instance name and port: **aagk80gpu52.rc.fas.harvard.edu:8192**.

You will use them to create a tunnel to access server running OmniSci Immerse via a browser.

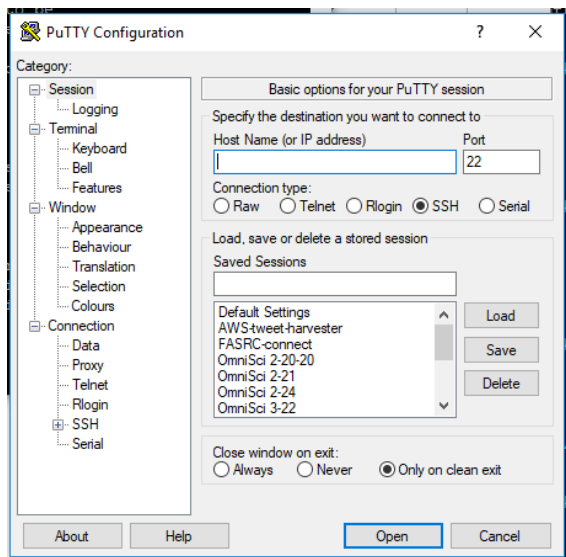
Create VPN Tunnel to new Instance

You will use this information to create a tunnel using putty Windows (Putty) or Linux command line `ssh -NL aagk80gpu52.rc.fas.harvard.edu:8192blewis@login.rc.fas.harvard.edu`

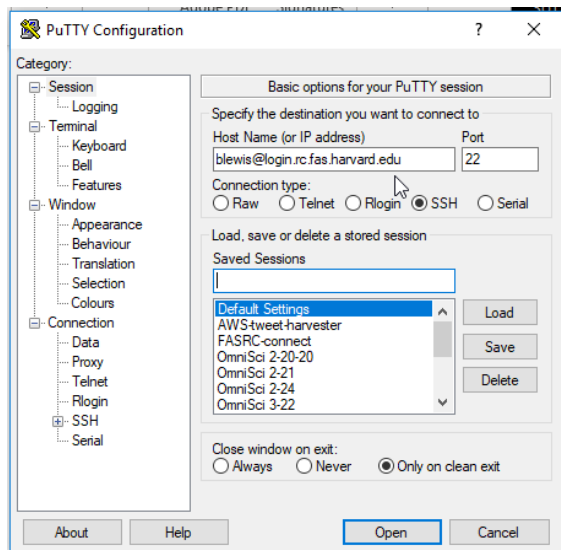
and this information to access Immerse once you have created the tunnel
open <http://localhost:8192> your browser [link](#)

Creating tunnel using Putty

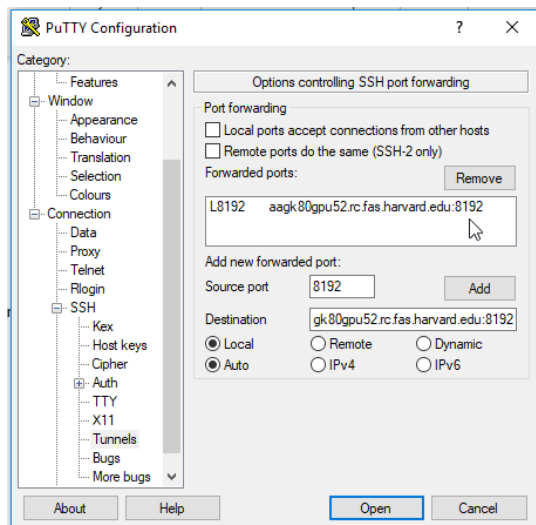
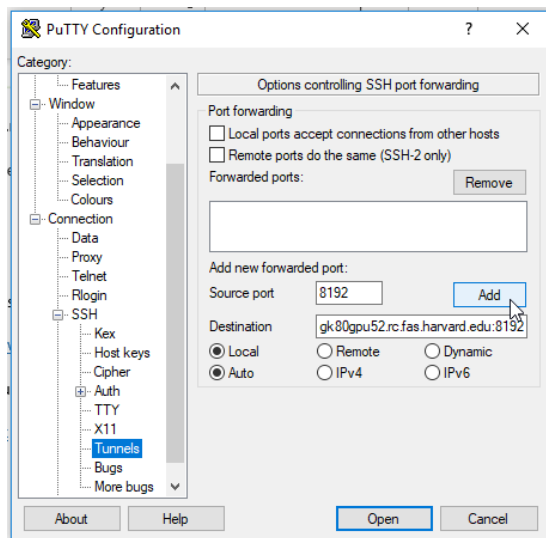
Open Putty:



Add user name/host name, and port:

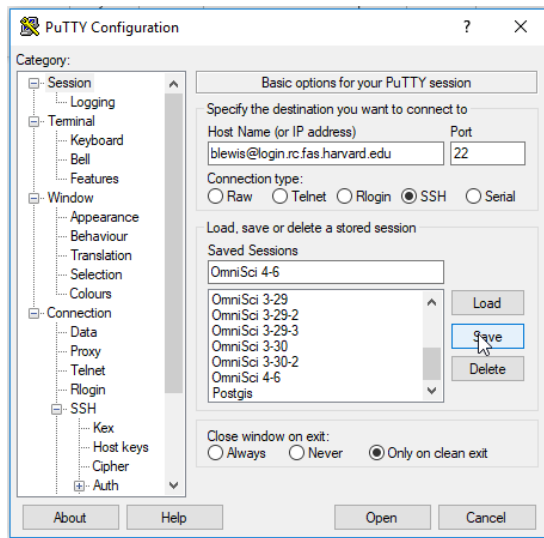


Go to SSH, then Tunnel. Past in your instance name and port, click Add:



Save your settings. Go to Sessions and give a name for the session, click Save.

As long as this instance exists, you can quickly create a tunnel to it using Putty and your saved setting.



The default life of an instance is one week. It is possible to extend an instance for a longer period of time by making a request to FASRC admins. It also may be possible to request machines with more RAM and CPUs.

Clicking Open on your Putty configuration will open a terminal and you will be prompted for your Harvard PIN password. Then you will be prompted for your third party authentication code.

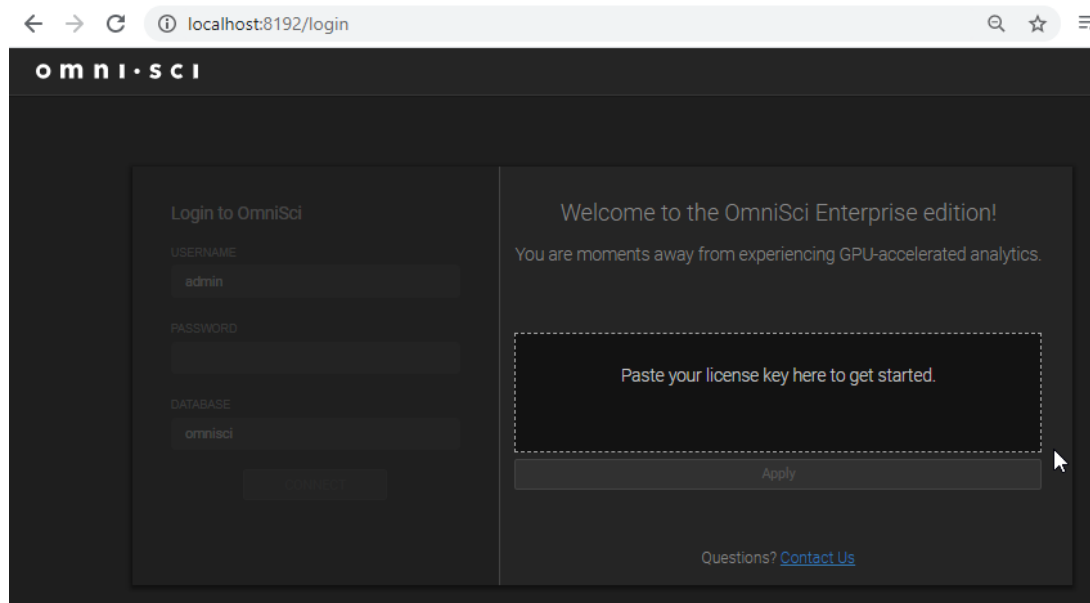
```
Using username "blewis".
Using keyboard-interactive authentication.
Password:
Using keyboard-interactive authentication.
Verification code:
```

Now you should be in.

NOTE: Do not perform computations on the home node you arrive on.

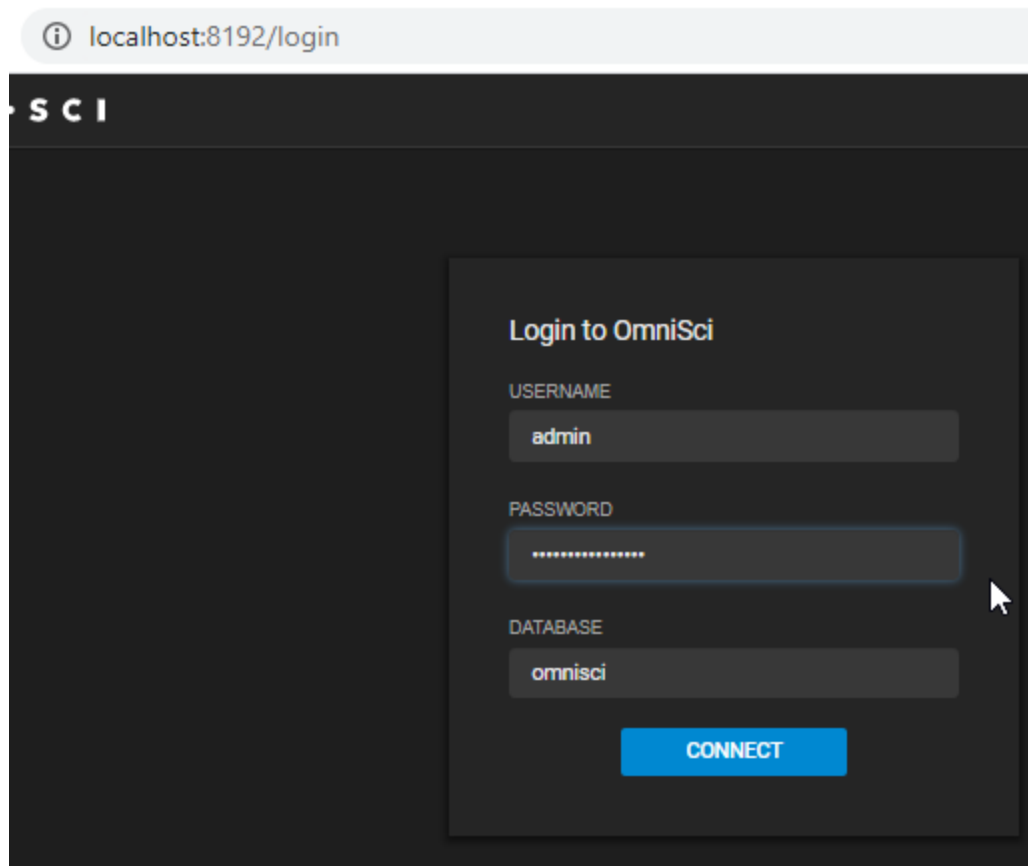
For any compute work you must SSH to the machine that you set up, i.e. aagk80gpu52.rc.fas.harvard.edu.

Accessing OmniSci Immerse from your browser via the tunnel

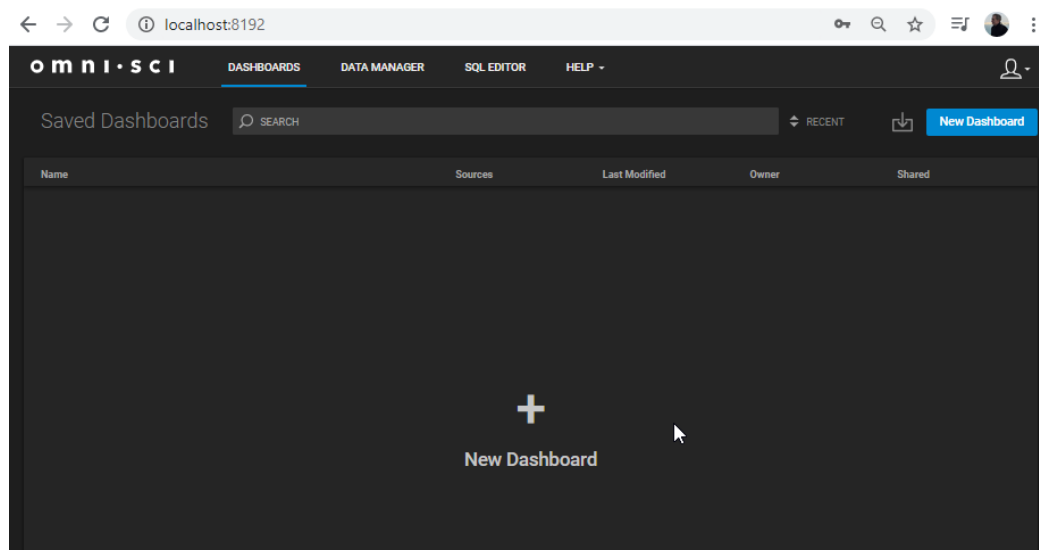


Paste in License Key and click Apply.

Login using user name and password



Now you can now start loading data and building dashboards



Running scripts

For any script you run which accesses your omnisci instance you will need to update the correct port in the script to use to access the OmniSci backend for loading data. This backend TCP port you will obtain by clicking on the Session ID.

OmniSci (51411901) 1 node | 4 cores | Running

Host: `>_aagk80gpu52.rc.fas.harvard.edu` Delete

Created at: 2020-04-06 11:07:36 EDT

Time Remaining: 167 hours and 57 minutes

Session ID: 9f867037-aea8-435c-b049-6d184ed392c1

☐ Connect to Omnisci

For the time being the proxy does not work for this application.
You can connect by tunneling via the login nodes:

1. `ssh -NL 8192:aagk80gpu52.rc.fas.harvard.edu:8192 blewis@login.rc.fas.harvard.edu`
2. open `http://localhost:8192` in your browser link

Which will take you to this page of information about the instance you just created. Click on “output.log” and click “Download”. Open log file.

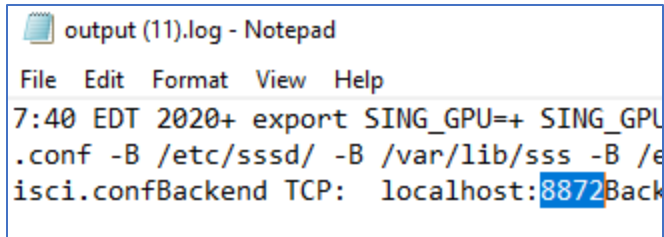
File Explorer Go To... Open in Terminal New File New Dir Upload Show Dotfiles Show Owner/Mode

`/n/home09/blewis/fasrc/data/sys/dashboard/batch_connect/dev/OmniSci/output/e7cbc5da-afdb-47a8-a1df-adedf29b3797/`

View Edit Az Rename/Move Download Copy Paste (Un)Select All Delete

name	size	modified date
..	<dir>	
after.sh	442b	02/18/2020
before.sh	2.44kb	06/24/2020
connection.yml	77b	06/24/2020
job_script_content.sh	3.09kb	06/24/2020
job_script_options.json	508b	06/24/2020
output.log	2.65kb	06/24/2020
script.sh	1.27kb	06/24/2020
script_original.sh	1.19kb	06/24/2020
user_defined_context.json	326b	06/24/2020

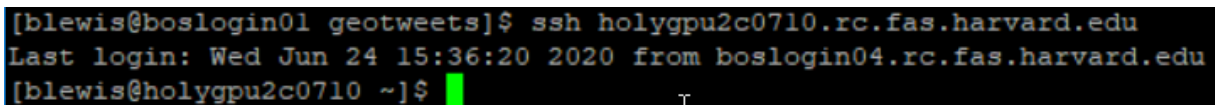
Scroll to bottom and you will find the Backend TCP. In this case it is 8872.



```
output (11).log - Notepad
File Edit Format View Help
7:40 EDT 2020+ export SING_GPU=+ SING_GPU
.conf -B /etc/sss/ -B /var/lib/sss -B /e
iscsi.confBackend TCP: localhost:8872Back
```

Again, before running processes you must ssh to your GPU instance after logging in to FASRC. Example:
`ssh aagk80gpu46.rc.fas.harvard.edu`

Or another example from the actual command line:



```
[blewis@boslogin01 geotweets]$ ssh holygpu2c0710.rc.fas.harvard.edu
Last login: Wed Jun 24 15:36:20 2020 from boslogin04.rc.fas.harvard.edu
[blewis@holygpu2c0710 ~]$
```

1. Connect to your compute node
2. For scripts involving OmniSci, PostGIS, and geotweets:
 - a. Load Conda: `module load Anaconda3/5.0.1-fasrc02`
3. Create environment, activate env and install libraries
 - a. `conda create -n geotweets python=3.6`
 - b. `source activate geotweets`
 - c. `pip install pandas`
 - d. `pip install geopandas`
 - e. `pip install numpy`
 - f. `pip install shapely`
 - g. `pip install pymapd`
4. Change the input file name to your CSV and run the script
 - a. `python3 /n/holyscratch01/cga/dkakkars/scripts/geotweets.py`

For more details please see Wiki https://github.com/cga-harvard/GIS_Apps_on_HPC/wiki.

Useful Links

- Intro to the Cannon Cluster <https://www.rc.fas.harvard.edu/wp-content/uploads/2019/12/Intro-to-Cannon.pdf>
- FASRC Quick Start Guide <https://docs.rc.fas.harvard.edu/kb/quickstart-guide/>
- Create Account and Access FAQ <https://docs.rc.fas.harvard.edu/kb/access-and-login/>
- How to Run Jobs <https://docs.rc.fas.harvard.edu/kb/running-jobs/>
- SLURM Commands <https://docs.rc.fas.harvard.edu/kb/convenient-slurm-commands/>
- HUIT Security Policy <https://security.harvard.edu/>

- Research Data Security Policy <https://vpr.harvard.edu/pages/harvard-research-data-security-policy>