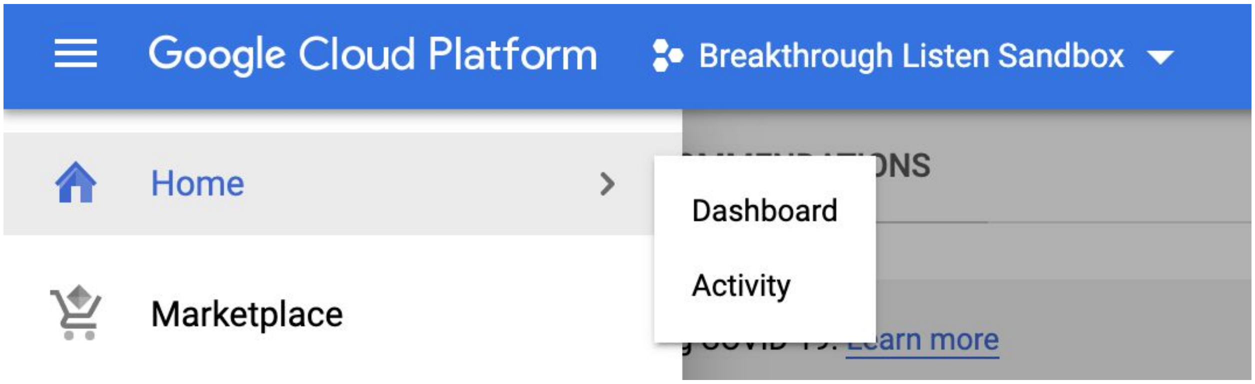
Guide to Creating a Virtual Machine on Breakthrough Listen's Virtual Sandbox

Written by Piper Stacey, Summer Intern, 2020
Email piper.23@dartmouth.edu with questions

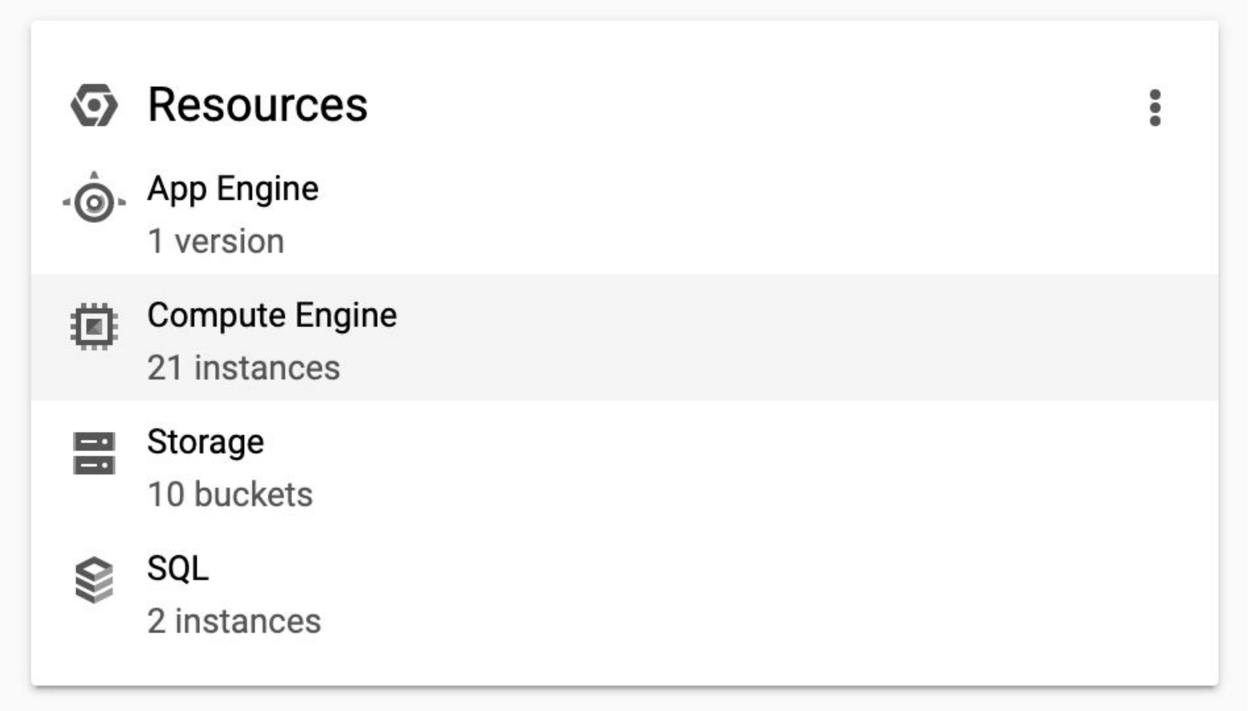
Table of Contents:

Step 1: Creating a Virtual Machine

a) Get an invite to BL's Virtual Sandbox on GCP
b) Navigate to GCP's home screen (go to **Dashboard**)



c) Navigate to **Resources** ➡ **Compute Engine**



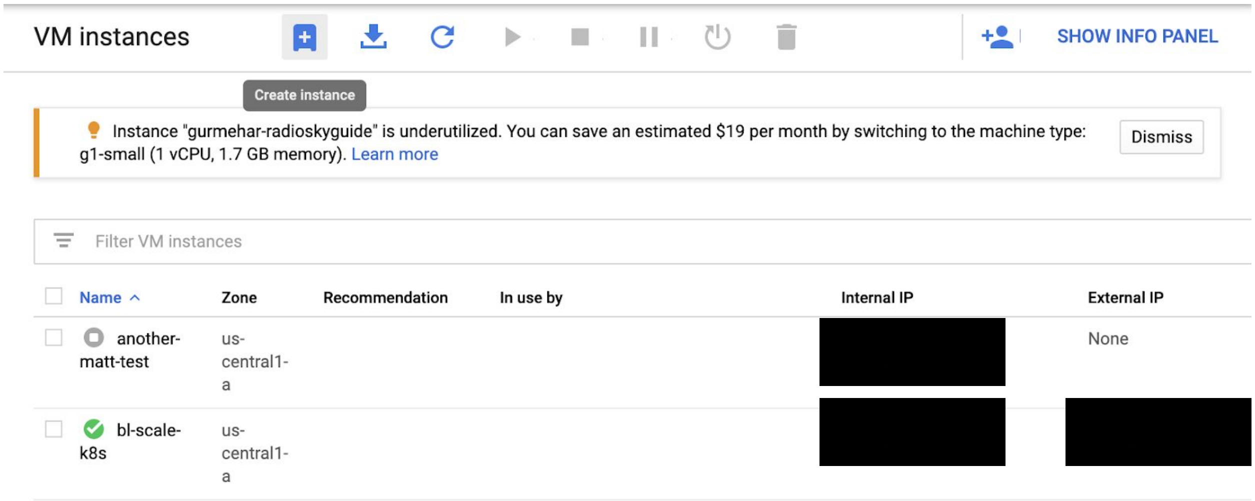
Resources

App Engine
1 version

Compute Engine
21 instances

Storage
10 buckets

SQL
2 instances

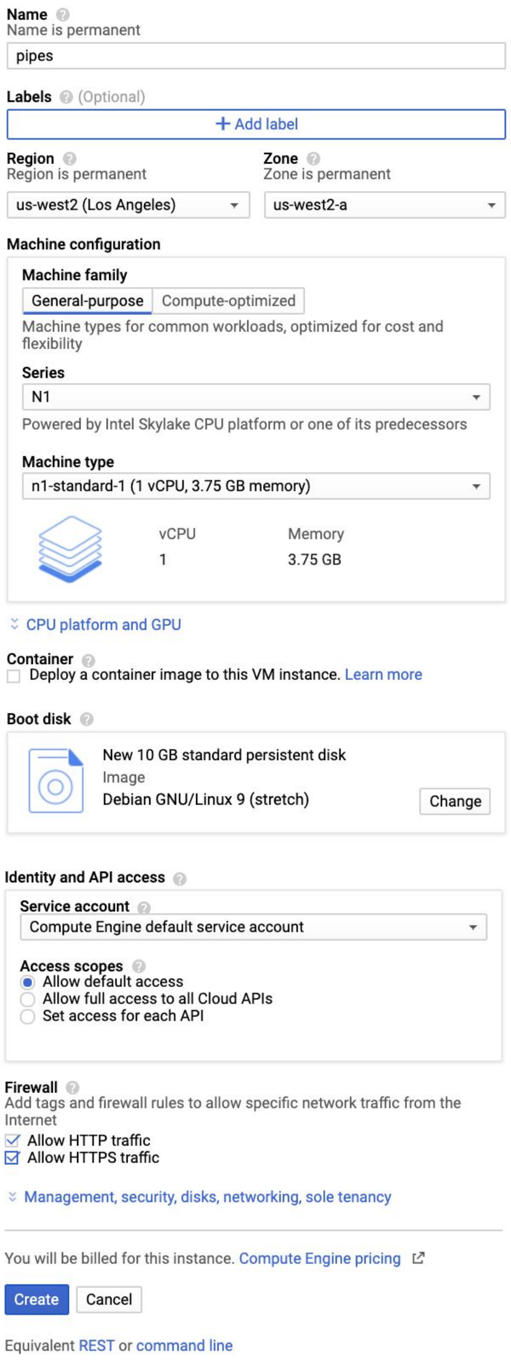


Instance "gurmehar-radioskyguide" is underutilized. You can save an estimated $19 per month by switching to the machine type: g1-small (1 vCPU, 1.7 GB memory). Learn more

□	Name ^	Zone	Recommendation	In use by	Internal IP	External IP
□	○ another-matt-test	us-central1-a			[redacted]	None
□	✓ bl-scale-k8s	us-central1-a			[redacted]	[redacted]

d) Click **Create Instance** in the top bar

e) Create your VM with the specifications you prefer. Follow this screenshot for standard specifications (IMPORTANT: click both the Allow HTTP Traffic and Allow HTTPS Traffic boxes at the end of this webpage as specified in this screenshot)



Name
Name is permanent
pipes

Labels (Optional)

Region
Region is permanent
us-west2 (Los Angeles)

Zone
Zone is permanent
us-west2-a

Machine configuration

Machine family

Machine types for common workloads, optimized for cost and flexibility

Series
N1
Powered by Intel Skylake CPU platform or one of its predecessors

Machine type
n1-standard-1 (1 vCPU, 3.75 GB memory)

vCPU	Memory
1	3.75 GB

Container
□ Deploy a container image to this VM instance. Learn more

Boot disk
New 10 GB standard persistent disk
Image
Debian GNU/Linux 9 (stretch)

Identity and API access

Service account
Compute Engine default service account

Access scopes
● Allow default access
○ Allow full access to all Cloud APIs
○ Set access for each API

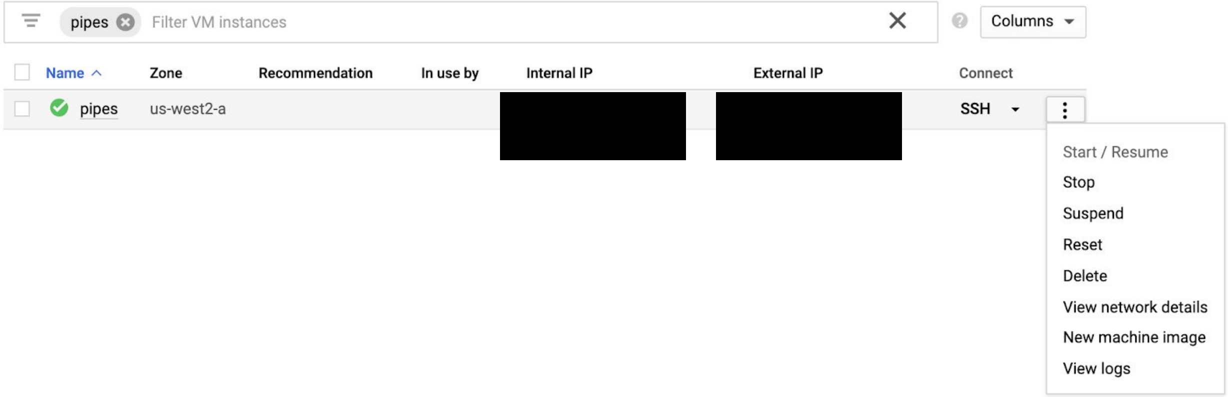
Firewall
Add tags and firewall rules to allow specific network traffic from the Internet
☑ Allow HTTP traffic
☑ Allow HTTPS traffic

You will be billed for this instance. Compute Engine pricing

Equivalent REST or command line

f) Click **Create**

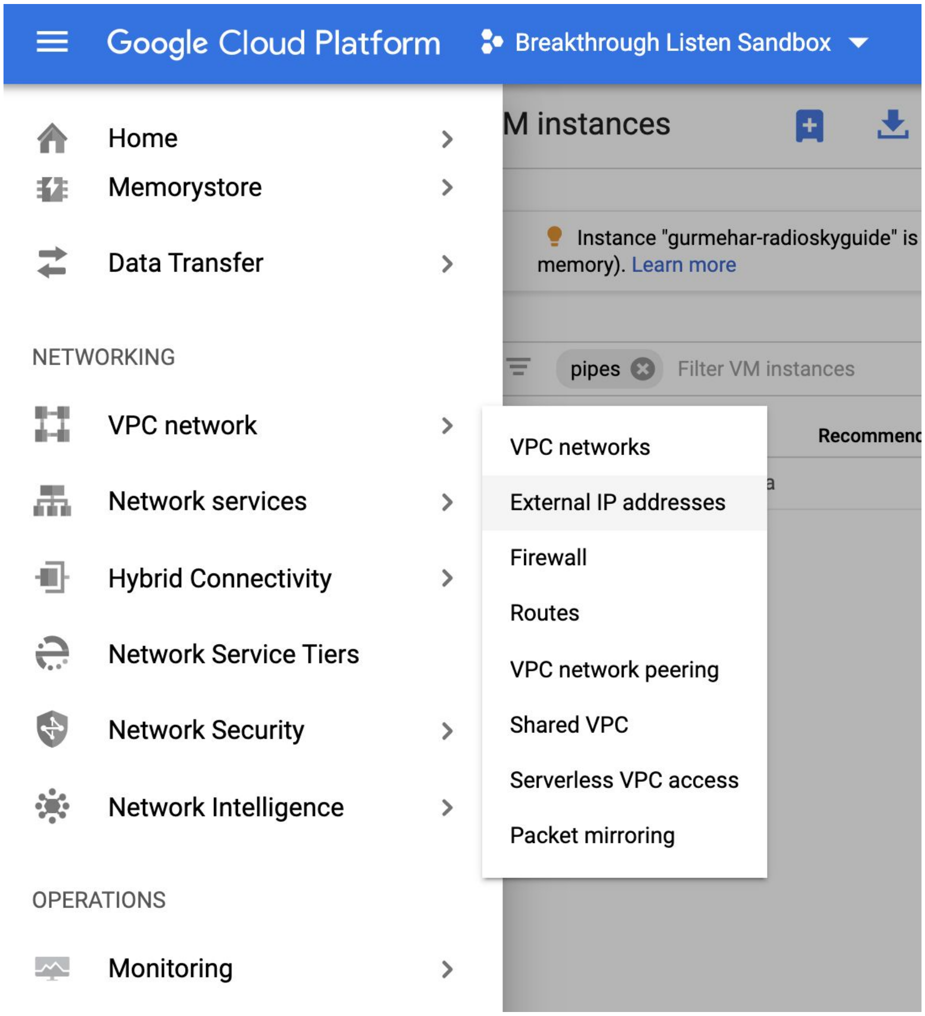
g) Tip: Click stop to stop being billed for this instance.



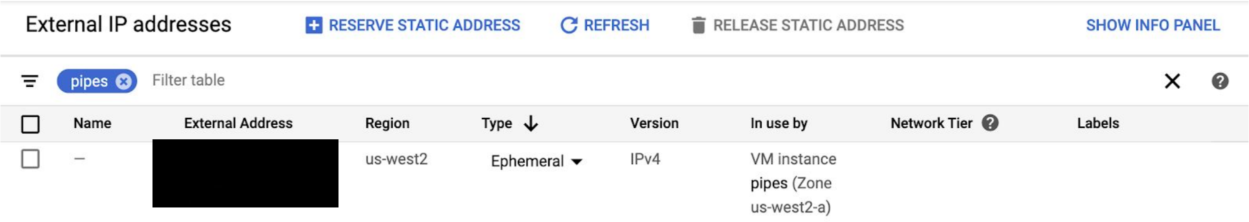
Name	Zone	Recommendation	In use by	Internal IP	External IP	Connect
pipes	us-west2-a			[redacted]	[redacted]	SSH

Step 2: Make external IP address Static

a) Go to the Menu button in the top right corner ➡ **VPC network** ➡ **External IP addresses**

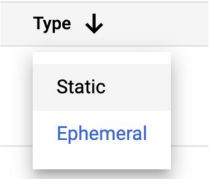


b) Search your VM's name in the search bar

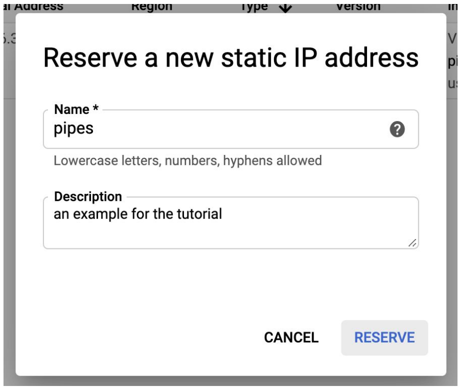


☐	Name	External Address	Region	Type ↓	Version	In use by	Network Tier	Labels
☐	–	[redacted]	us-west2	Ephemeral ▾	IPv4	VM instance pipes (Zone us-west2-a)		

c) Change **Type** from **Ephemeral** to **Static**



d) Reserve a new static IP address as instructed and add a description if you please



Reserve a new static IP address

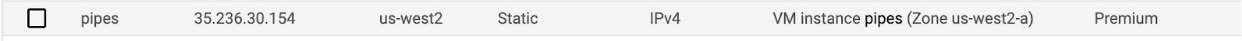
Name *
pipes

Lowercase letters, numbers, hyphens allowed

Description
an example for the tutorial

e) Hit **Reserve**

f) You should now see your VM at the top of the queue of External IP Addresses



□	pipes	35.236.30.154	us-west2	Static	IPv4	VM instance pipes (Zone us-west2-a)	Premium

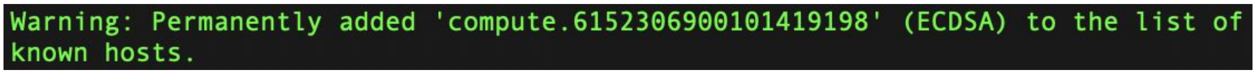
Step 3: Install google-cloud-sdk on your local machine

a) Install google-cloud-sdk
   i) Follow this link to install whichever version is best matched with your **local machine**:
   ii) https://cloud.google.com/sdk/docs/downloads-interactive

Step 4: Start up your VM from terminal (or equivalent)

a) Run this command:
   iii) gcloud beta compute ssh --zone "us-west2-a" "username@pipes" --project "breakthrough-listen-sandbox" -- -L 5555:localhost:5555
      1) Change your time zone to the one you specified when you created your VM
      2) Change your username to the email address on which you were given access to Breakthrough Listen's Sandbox (or anything—I'm not really sure this matters)
      3) Change the virtual machine name ("pipes" in this case) to whatever you called your VM
      4) Change the port ("5555" in this case) to whichever port you want to use on your VM and whichever port you want to use on local machine (in order of left to right)
      5) Run your altered command

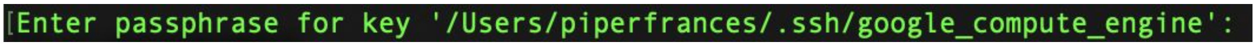
You should see this:



```
Warning: Permanently added 'compute.6152306900101419198' (ECDSA) to the list of
known hosts.
```

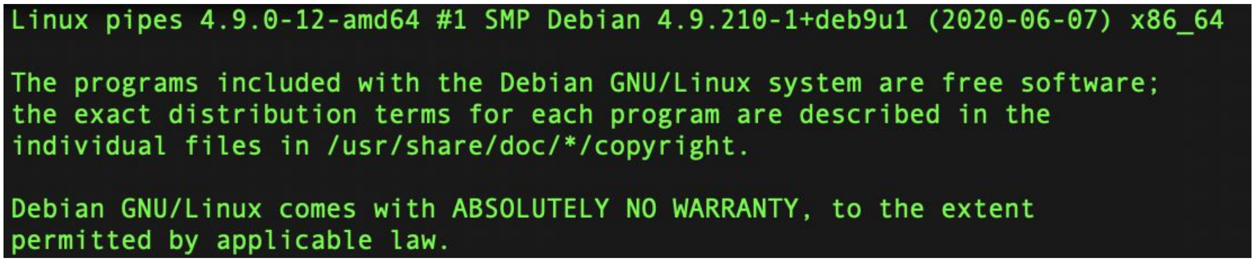
b) If you have never done this before, it may ask you to enter a password. Do so and remember your password!

c) If you have done this before, enter your password



```
[Enter passphrase for key '/Users/piperfrances/.ssh/google_compute_engine':
```

Now you should see this:



```
Linux pipes 4.9.0-12-amd64 #1 SMP Debian 4.9.210-1+deb9u1 (2020-06-07) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
```

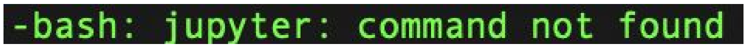
d) Run this command:

   i) jupyter notebook --port=5555

      1) If you altered your port in the previous ssh tunneling command, change "5555" to whatever port you used there (for your VM)

      2) Run the command

e) If you see this:



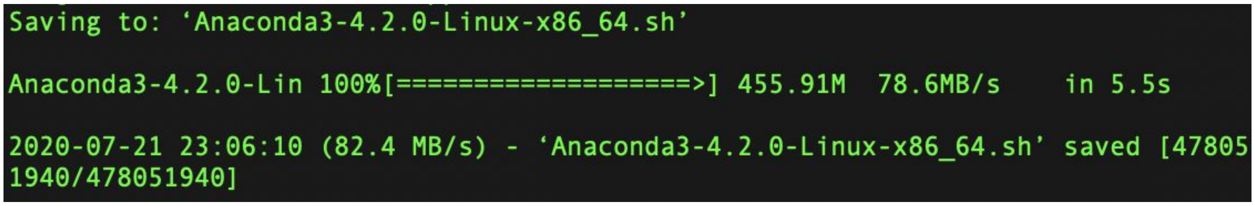
```
-bash: jupyter: command not found
```

[run the following commands to install anaconda3 and jupyter]

   i) Then run this command:

      1) wget http://repo.continuum.io/archive/Anaconda3-4.0.0-Linux-x86_64.sh

      2) You should see this progress bar as Anaconda installs:

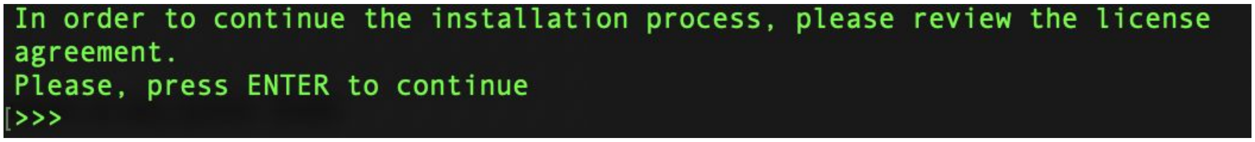


```
Saving to: 'Anaconda3-4.2.0-Linux-x86_64.sh'

Anaconda3-4.2.0-Lin 100%[===================>] 455.91M  78.6MB/s    in 5.5s

2020-07-21 23:06:10 (82.4 MB/s) - 'Anaconda3-4.2.0-Linux-x86_64.sh' saved [47805
1940/478051940]
```

   ii) Now run this command:

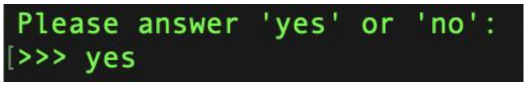
      1) bash Anaconda3-4.0.0-Linux-x86_64.sh

      2) Hit Enter



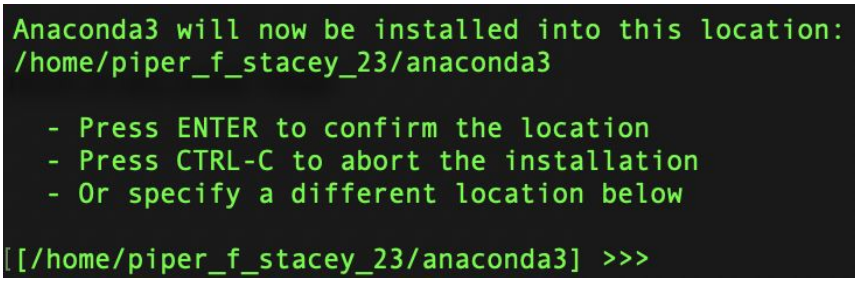
```
In order to continue the installation process, please review the license
agreement.
Please, press ENTER to continue
[>>>
```

      3) Continue through the license and then type "yes" at the end:



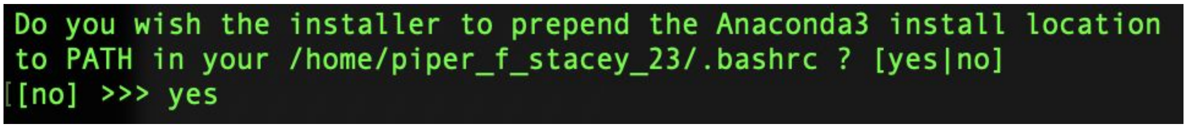
```
Please answer 'yes' or 'no':
[>>> yes
```

4) When you see this, I recommend Pressing Enter but you may specify a different location for Anaconda3



```
Anaconda3 will now be installed into this location:
/home/piper_f_stacey_23/anaconda3

  - Press ENTER to confirm the location
  - Press CTRL-C to abort the installation
  - Or specify a different location below

[[/home/piper_f_stacey_23/anaconda3] >>>
```

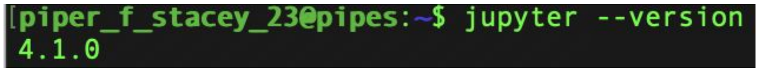
5) Now, allow the installer to prepend Anaconda3 to your .bashrc file:



```
Do you wish the installer to prepend the Anaconda3 install location
to PATH in your /home/piper_f_stacey_23/.bashrc ? [yes|no]
[[no] >>> yes
```

6) Now, **exit** out of your connection to your VM and **close that terminal window**

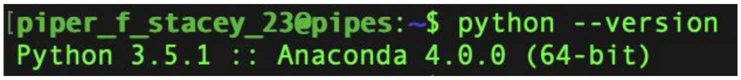
7) Reopen terminal and ssh into your VM once more. Now check the version of Jupyter that you have installed to check that it is there:



```
[piper_f_stacey_23@pipes:~$ jupyter --version
4.1.0
```

8) Success!

f) Now you should have all the packages you need. Feel free to install any more at any point. Sudo apt install works best to install further packages and install pip.



```
[piper_f_stacey_23@pipes:~$ python --version
Python 3.5.1 :: Anaconda 4.0.0 (64-bit)
```

g) Now, finally, run this command:

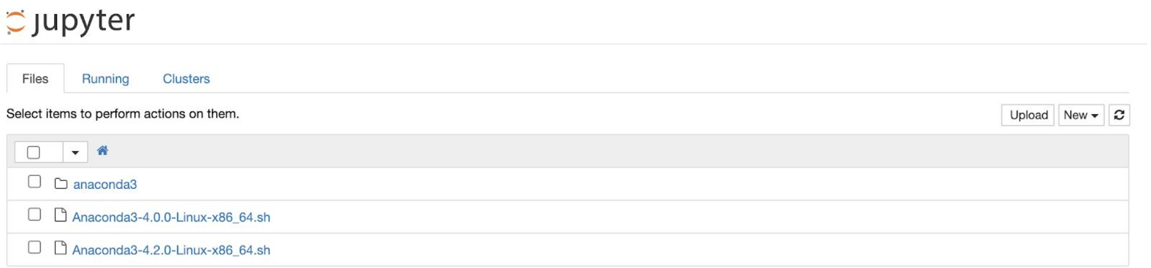
i) jupyter notebook --port=5555

1) If you altered your port in the previous ssh tunneling command, change “5555” to whatever port you used there (for your VM)

2) Run the command

h) Either you will see links for you to copy and paste in your terminal window or you can go to http://localhost:5555/ where the “5555” is the port you input as your local port during the ssh tunneling step

Hopefully you now see this:



Select items to perform actions on them.

- anaconda3
- Anaconda3-4.0.0-Linux-x86_64.sh
- Anaconda3-4.2.0-Linux-x86_64.sh

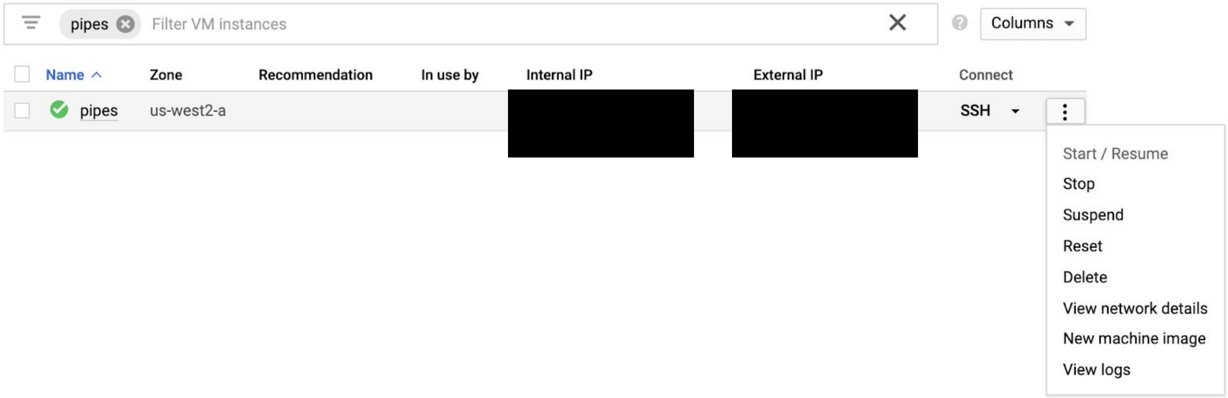
And you’re in!

Step 5: Install Git

a) Run this command:
   - i) sudo apt install git
   - ii) Press ‘y’ when instructed to do so
   - iii) Git should install

* Tip: link your personal git repository to what you have on GCP as another way to back up the scripts and files you have there

* Tip: Within the GCP website, click stop to stop being billed for this instance.



Name	Zone	Recommendation	In use by	Internal IP	External IP	Connect
pipes	us-west2-a			[redacted]	[redacted]	SSH