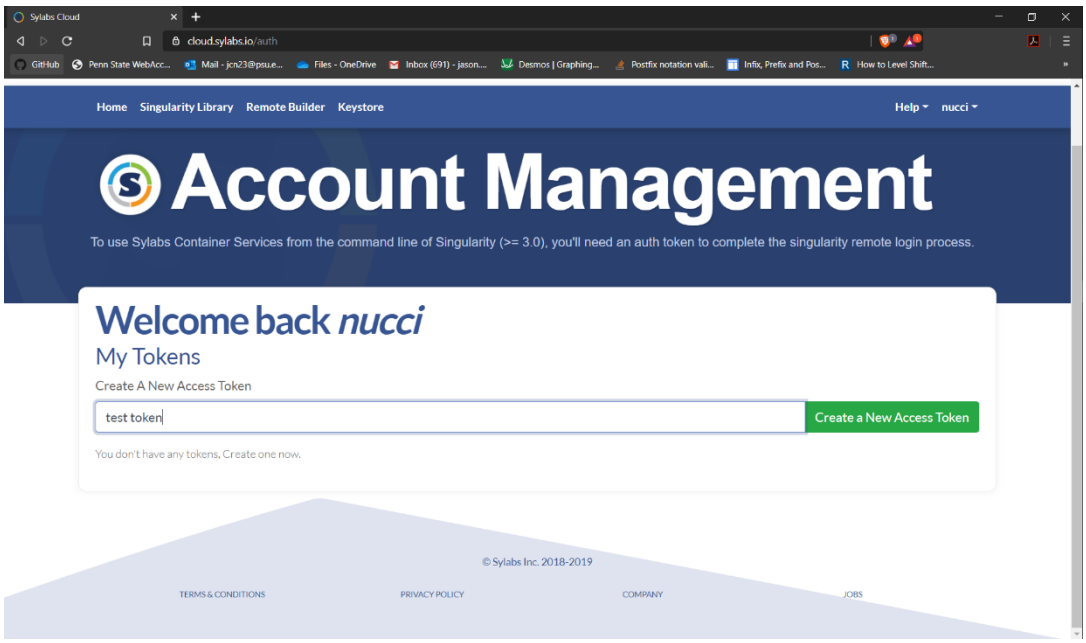
Pushing Singularity Images from the Command Line to the Cloud

Assuming that you already have Singularity installed on your Linux machine and you have already made an account with https://cloud.sylabs.io, this is a step-by-step guide on how to push a Singularity image that you have made on your local machine up to the cloud.

Setting up access on your computer:

1. First, go to https://cloud.sylabs.io and login
2. After logging in, click on your login id and select "Access Tokens"

3. After selecting "Access Tokens," enter a name for your new token and then select "Create a New Access Token"



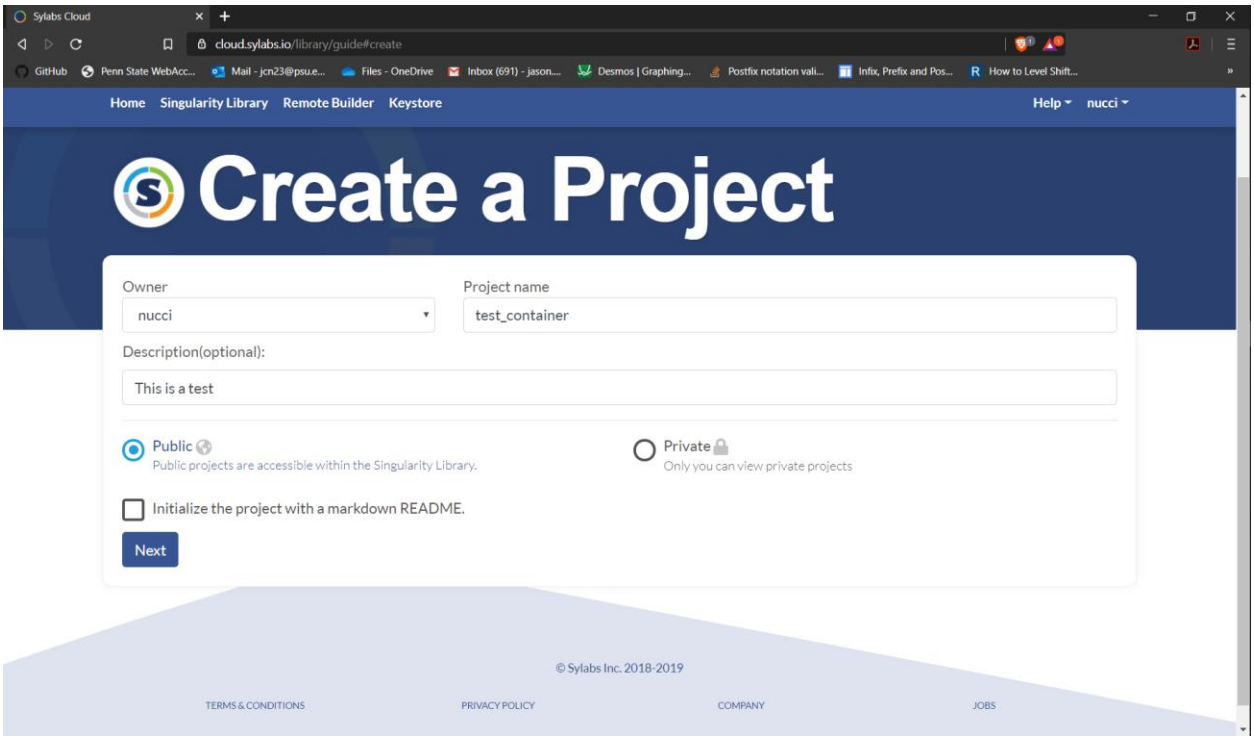
4. After creating your new token, click “Copy token to Clipboard”
5. Once you have copied the token, open Terminal and enter the command:

```
$ singularity remote login
```

After entering the command, you will be prompted to enter your access token. Right-click, select “Paste,” and then hit Enter. You are now ready to start pushing images to the cloud!

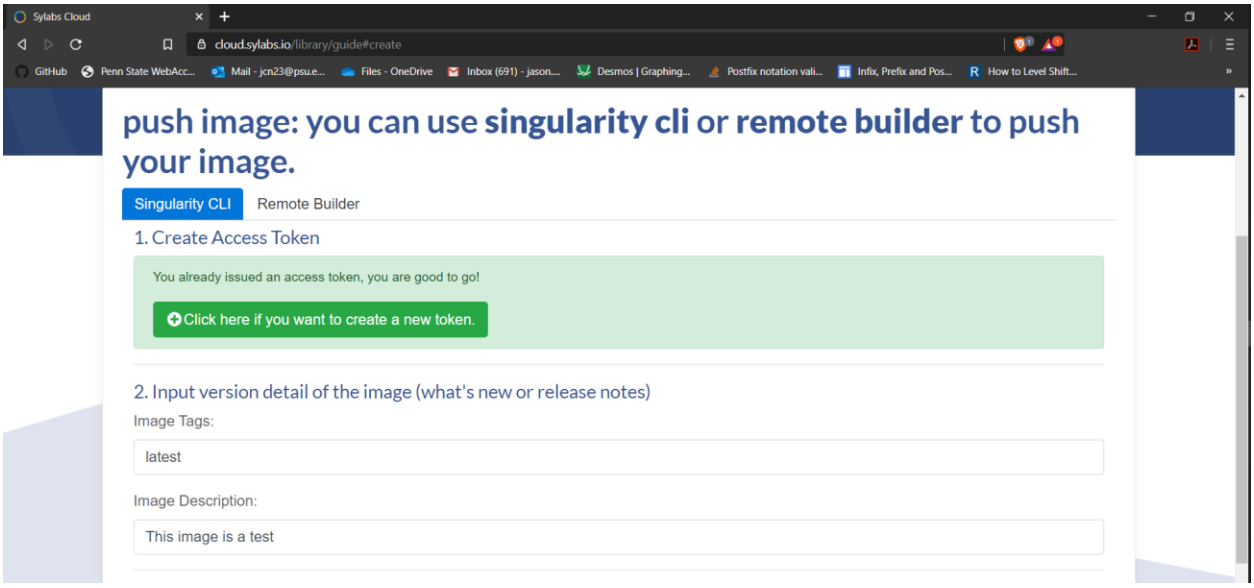
Creating a new project

1. Go to https://cloud.sylabs.io and selecting “Browse Library”
2. Now, select “Create a new Project”
3. You will be taken to the “Create a Project” page. Proceed through the following steps:
    a. Specify an owner of the project (usually yourself)
    b. Specify a name for the project (be specific)
    c. Write a short description about the container
    d. Select whether you want the container to be public or private
    e. Check or uncheck the “Initialize the project with a markdown README” (You should check this box if more than one user is going to be using this container)



4. You will now be taken to the “push image” page. Proceed through the following steps:
    a. You will be prompted to create an access token. You can ignore this step if you already made one earlier

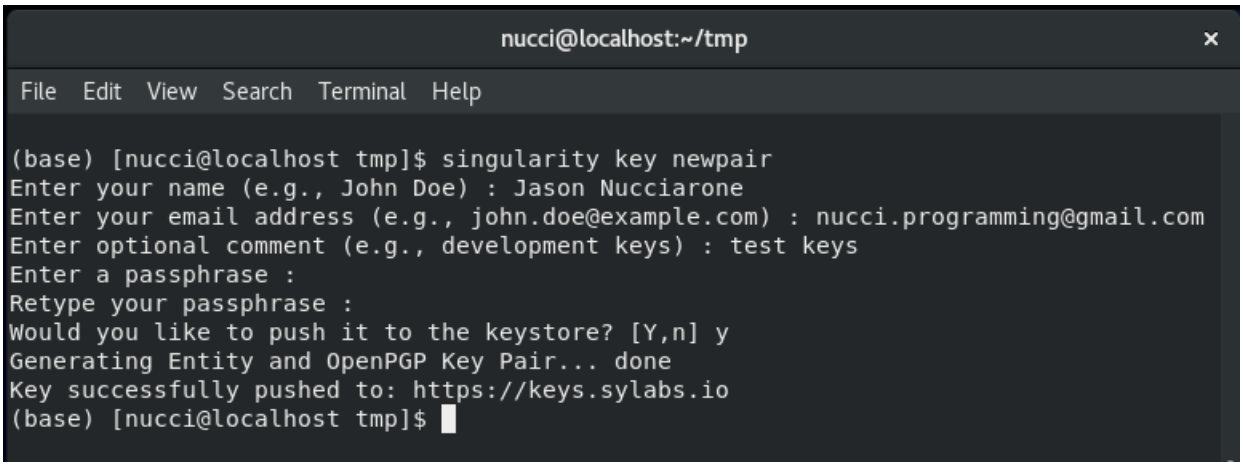
b. Specify a tag for you image (latest, v1.0.0, etc.)
c. Write a short description about the image (Specify what it does, what you changed, how you configured it, etc.)



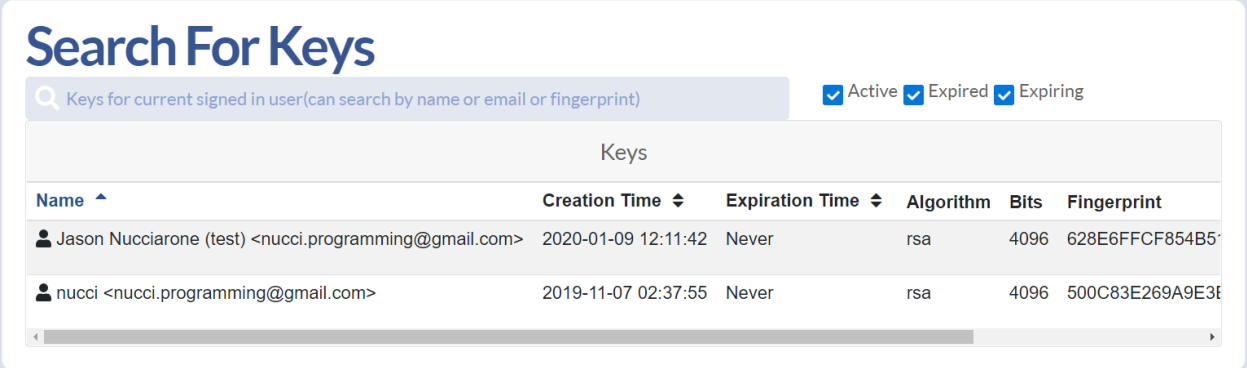
Pushing image from the Command Line

1. Before you can push your image up to the cloud you need to sign the image first. This process is relatively easy, however, for a first-time user it can be difficult. Use the following steps to create a key and sign your image:
    a. First, enter the following command:
    ```
    $ singularity key newpair
    ```
    b. You will now be prompted to create a new key. Think of a key as your way of personalizing your image; it lets other users know that you created it. Use the following steps to create your own key:
        i. Enter your name (i.e. John Doe)
        ii. Enter your email address
        iii. Enter an optional comment for the key (test, devel, release, etc.)
        iv. Create a passphrase for your key (make it something that is easier to remember as you will not be able to change it later)

v. Push your key to the keystore (Stores your keys on the cloud so that you can access them from multiple machines. Should you choose not to push the key to the store then it will only be available on your local machine)

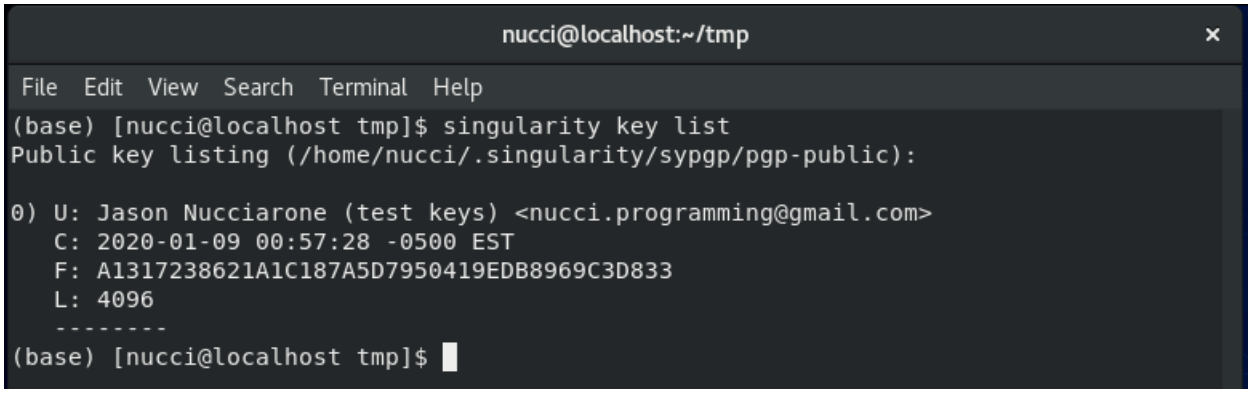


vi. Should you ever want to see what keys you have pushed up to the cloud, don't go to https://keys.sylabs.io. You will receive a 404 HTTP error. Instead, go to https://cloud.sylabs.io/keystore



vii. Should you ever want to see the keys that you have stored locally, simply enter the command:

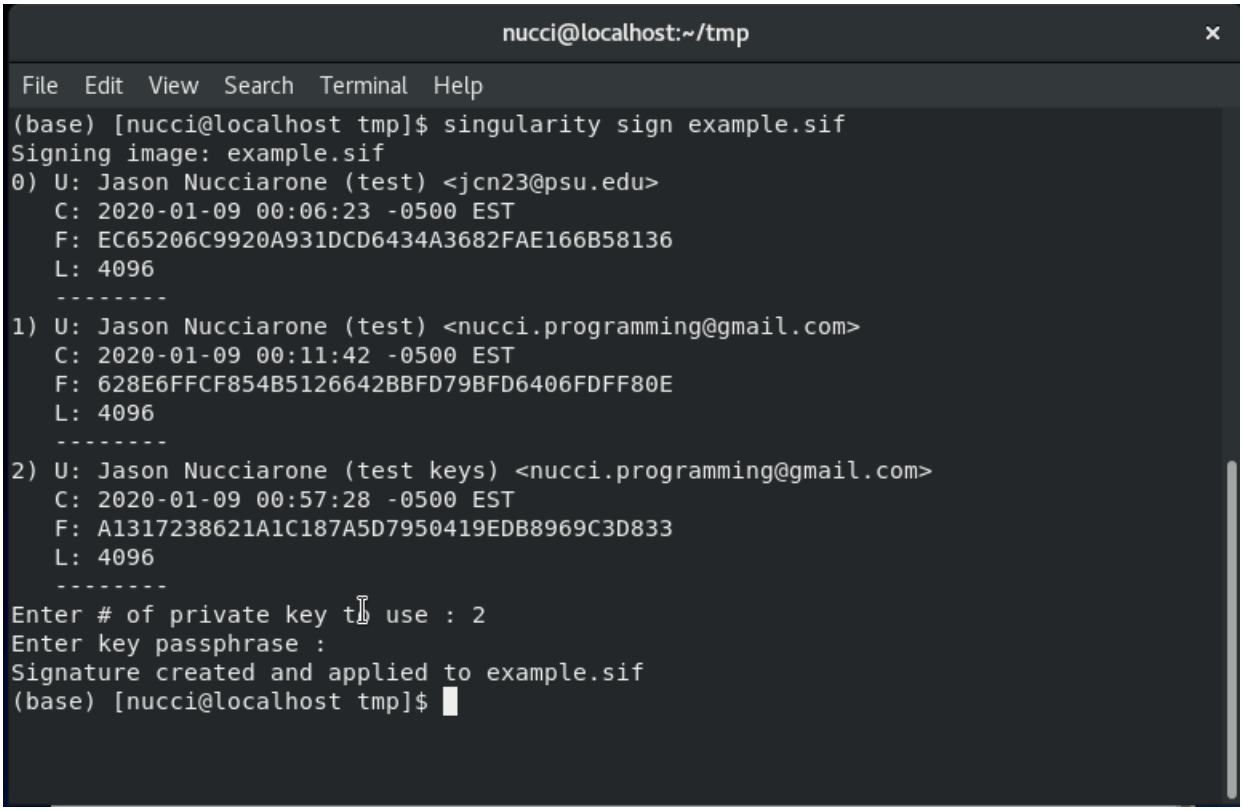
```
$ singularity key list
```



c. Now that you have created your new key, you will now be able to sign your images using the following command:

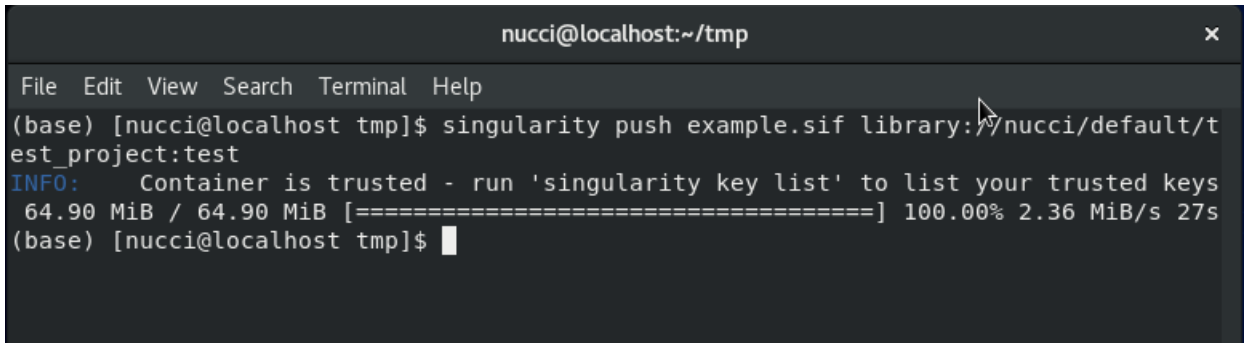
```
$ singularity sign myImage.sif
```

d. You will be prompted to select which key you want to use. Simply insert the integer value and hit Enter. After that, enter the passphrase for the key and hit Enter again. Congratulations! You signed your container!



2. Now that we have gotten signing out of the way, it is time to push your image to the cloud. Use the following command to push your image to the cloud:

```
$ singularity push myImage.sif library://path/to/project
```



And here is what the final project should look like in the end!