Singularity

.sif i cache

```
singularity pull hello-world.sif shub://vsoch/hello-world
```

```
singularity cache list [-v]
```

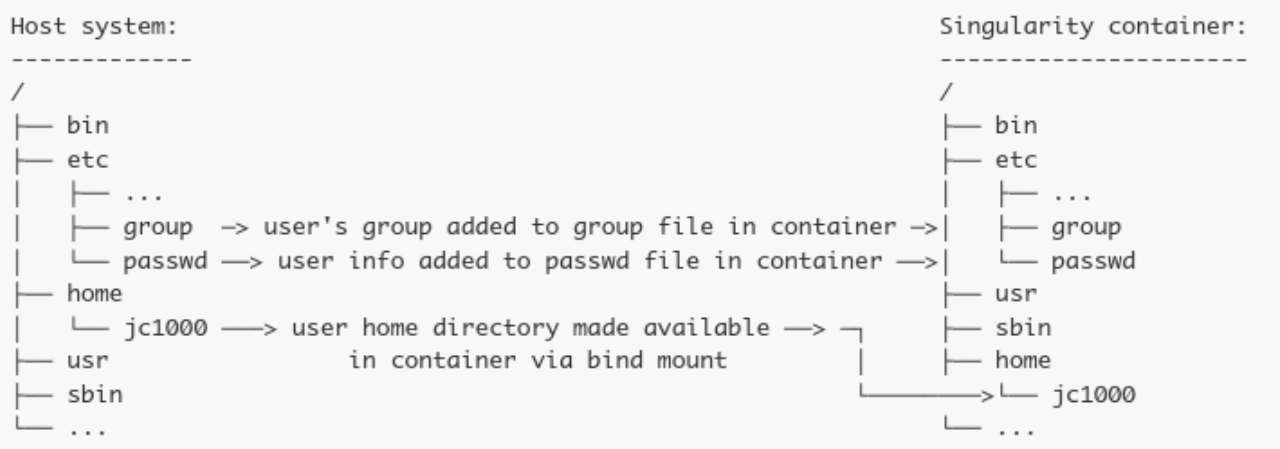
containers

```
singularity exec hello-world.sif /bin/echo Hello World!
```

```
singularity shell hello-world.sif
```

```
singularity shell hello-world.sif
Singularity> whoami
```

- If you have any familiarity with Linux system administration, you may be aware that in Linux, users and their Unix groups are configured in the /etc/passwd and /etc/group files respectively. In order for the shell within the container to know of my user, the relevant user information needs to be available within these files within the container.
- Assuming this feature is enabled on your system, when the container is started, Singularity appends the relevant user and group lines from the host system to the /etc/passwd and /etc/group files within the container [1].
- Singularity also binds some directories from the host system where you are running the singularity command into the container that you're starting. Note that this bind process isn't copying files into the running container, it is simply making an existing directory on the host system visible and accessible within the container environment. If you write files to this directory within the running container, when the container shuts down, those changes will persist in the relevant location on the host system.
- There is a default configuration of which files and directories are bound into the container but ultimate control of how things are set up on the system where you're running Singularity is determined by the system administrator. As a result, this section provides an overview but you may find that things are a little different on the system that you're running on.




```
singularity pull python-3.8.2.sif docker://python:3.8.2-slim-buster
```

- Singularity can also start containers from Docker images, opening up access to a huge number of existing container images available on Docker Hub and other registries.

- While Singularity doesn't support running Docker images directly, it can pull them from Docker Hub and convert them into a suitable format for running via Singularity. When you pull a Docker image, Singularity pulls the slices or layers that make up the Docker image and converts them into a single-file Singularity SIF image.
- Note how we see singularity saying that it's "Converting OCI blobs to SIF format". We then see the layers of the Docker image being downloaded and unpacked and written into a single SIF file. Once the process is complete, we should see the python-3.8.2.sif image file in the current directory.

/via: https://imperialcollegelondon.github.io/2020-07-13-Containers-Online/10-singularity-containers/index.html