

Create/Build a Singularity Container Image

The documented method to build a Singularity container image requires using sudo privilege. In this how-to document, we outline how to work with this constraint. Obviously if you have sudo somewhere that you can use to build a Singularity container image, your problem is solved. For those who don't, here are some of the options:

1. Install [Oracle Virtual Box](#) on your desktop or laptop. You can then install your own Linux virtual machine with the Singularity rpm, giving yourself root/sudo privilege on that VM. This will allow you to build an image. The solution works for both Windows, Mac and Linux desktop/laptop.
 - a. Even easier than using Virtual Box directly: Use [Vagrant](#). Vagrant uses Virtual Box to spin up VMs, but it's automated and very fast. The advantage to Vagrant is that the OS install does not need to be done – Vagrant downloads preinstalled images, so you can run "vagrant up" and then ssh into your VM in seconds, and use sudo there.
2. Similarly, one can install Docker on your desktop or laptop. Then install / pull a Linux Docker container and install Singularity rpm in the container. The rest is the same as above.
3. Build what you need in a Docker image and upload to Docker Hub, and on a SLAC Linux host, use command "singularity pull your_docker_image" to convert and save it as a Singularity image.

If you want to do none of the above, you have an option to build a Singularity container image without root on a SLAC CentOS 7 machine. This method is a bit of hacking so it may not work in the future when Singularity makes significant changes. The following are the steps to do this, assuming you are on a SLAC CentOS 7 public login node (centos7.slac.stanford.edu), your current directory is /tmp, and you want to build a CentOS based Singularity container image:

- Download and save a base CentOS image from Docker with the following command. This command downloads the image to /tmp/centos_centos7.7.1908.sif. The image contains a very basic set of CentOS 7 operating system. It does include yum and rpm commands so that one can install more RPMs.

```
singularity pull docker://centos:centos7.7.1908
```

- Extract the squashfs image from the saved .sif file. A .sif has several objects inside, which you can exam by using "singularity sif list /tmp/centos_centos7.7.1908.sif" command. The squashfs image we need is usually the 3rd object. Extract and save it with command

```
singularity sif dump 3 /tmp/centos_centos7.7.1908.sif > /tmp/centos_centos7.7.1908.sqsh
```

- Unpack CentOS 7 operating system files from the squashfs image. This will create a sub-directory squashfs-root and put all the files there. All files under this directory are own by you, not root.

```
/sbin/unsquashfs -no-xattrs /tmp/centos_centos7.7.1908.sqsh
```

- Singularity can use the files in /tmp/squashfs-root as an "image" (Singularity sandbox image). But one needs to change a few things in order to install additional RPMs via yum in the image. So do this:

- Fix a few directory permission issues:

```
cd /tmp/squashfs-root; find usr bin lib* etc root var -type d -exec chmod u+w {} \;
```

- By default, yum checks if the user is root, and will refuse to install anything if the user is not. So edit /tmp/squashfs-root/usr/share/yum-cli/yumcommands.py and comment out the relevant code (in red)

```
def checkRootUID(base):
    """Verify that the program is being run by the root user.

    :param base: a :class:`yum.Yumbase` object.
    :raises: :class:`cli.CliError`
    """
    # if base.conf.uid != 0:
    #     base.logger.critical(_('You need to be root to perform this command.'))
    #     raise cli.CliError
```

- After the above changes, you can start using the image, installing RPMs or do other customization, such as

```
singularity shell /tmp/squashfs-root
```

```
singularity exec -w --no-home /tmp/squashfs-root yum install git
```

- When you are done, you can create a squashfs image, which can be use by Singularity at other places.

```
cd /tmp/squashfs-root; /sbin/mksquashfs . /tmp/my-container.img
```

```
singularity shell /tmp/my-container.img
```

Note that this image can also be used by Shifter at NERSC. But since this method doesn't follow NERSC's normal procedure to load Shifter image, you need to ask NERSC to load the image to their image pool manually.



Related articles

- [Create/Build a Singularity Container Image](#)