

Running Docker images using Shifter

- [What is Shifter ?](#)
- [Authentication](#)
- [Upload an image to Shifter from Dockerhub using the registry.c4science.ch \(preferred\)](#)
- [Upload an image directly from Dockerhub to Shifter](#)
- [Run the image on the clusters](#)
- [Interactive Shell \(Bash\)](#)
- [Using GPUs](#)
- [Account on the c4science registry](#)
- [Related articles](#)

When you want to run Docker containers on the clusters.

What is Shifter ?

Shifter allows to use Docker images in a shared computing environment in userland. It converts the docker image to the squashfs format that can be mounted and chrooted into as a user.

The SCITAS filesystems are also mounted inside the image so you can easily access your data. Please note that if your image has some files in /home there will not appear in the image and will be hidden.

Authentication

You can use public images without being authenticated (shifterimg login) in the following cases:

Image source	Public image	Login required
Dockerhub	Yes	No
Dockerhub	No	Yes, Dockerhub
C4science registry	Yes	No
C4science registry	No	Yes, c4science registry

When using authentication, it's relative to the source you want to use (Dockerhub and c4science registry supported) and the authentication is saved per cluster.

You need to ask for an account on the c4science registry, see [Account request](#).

NOTE: By default, shifterimg is pulling from c4science registry, you have to specify [registry-1.docker.io/library/alpine:latest](#) for Dockerhub

Upload an image to Shifter from Dockerhub using the registry.c4science.ch (preferred)



Prerequisite

You need to have Docker installed on your machine

1. Get a docker image from Dockerhub for instance (From local computer)

```
$ docker pull alpine:latest
$ docker images
```

2. Set up your machine (From local computer)

Login on the registry from your local Docker installation (you need an account, see [Account request](#))

```
$ docker login registry.c4science.ch
Username (username): username
Password:
Login Succeeded
```

3. Upload a Docker image to the registry (From local computer)

- On the web interface, create a Project on the registry (private or public)

- Tag the image you want to upload on your local machine and push it to the registry
NOTE: Do not use the `` character in the tag name, only letters, numbers and underscore

```
$ docker tag alpine:latest registry.c4science.ch/yourproject/alpine:latest
$ docker push registry.c4science.ch/yourproject/alpine:latest
```

4. Pull an image on Shifter and specify a user or group ACL (From cluster login node)

- From each cluster frontend (i.e.: fidis.epfl.ch), login to the registry, pull the image and check it's was pulled with success.

```
$ shifterimg login
  default username: <username>
  default password:
$ shifterimg pull yourproject/alpine:latest
$ shifterimg images
tcm      docker      READY      9797e5e798    2018-03-15T16:00:59 yourproject/alpine:latest
```

- You can specify one or multiple (separated by a comma) LDAP username and/or group so the image is only available to those people

```
$ id
$ shifterimg --group scitas-ge --user aubort,user2 pull yourproject/alpin
```

- To update the user/group ACL you can re-run the pull command
- The images are unique for each cluster (deneb, fidis, helvetios, izar)
- To view the full info about the images (warning: JSON):

```
$ shifterimg -v images
Message: {
  "list": [
    {
      "ENTRY": null,
      "ENV": [
        "PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"
      ],
      "WORKDIR": "MISSING",
      "groupACL": [],
      "id": "9797e5e798a034d53525968de25bd25c913e7bb17c6d068ebc778cb33e3ff6e5",
      "itype": "docker",
      "last_pull": 1536842228.15727,
      "status": "READY",
      "status_message": "",
      "system": "fdata2-int.fidis",
      "tag": [
        "scitas/alpine:latest"
      ],
      "userACL": []
    },
    [...]
  ]
}
```

Upload an image directly from Dockerhub to Shifter

Optional (needed for private images only)

```
$ shifterimg login registry-1.docker.io
```

Get the image from Dockerhub by specifying the image account (optional for official image) and name

```

$ shifterimg pull registry-1.docker.io/alpine:latest
# OR
$ shifterimg pull registry-1.docker.io/library/alpine:latest

$ shifterimg images
fidis      docker    READY    4a28aef4f8    2020-10-16T14:52:41    registry-1.docker.io/alpine:latest
fidis      docker    READY    d0877b767c    2020-09-16T15:26:31    registry-1.docker.io/library/busybox:latest

$ srun --pty -p debug -N 1 shifter --image registry-1.docker.io/alpine:latest /bin/sh
~ $ cat /etc/alpine-release
3.12.0

```

Run the image on the clusters

You can submit the following Slurm script with the *sbatch* command

```

#!/bin/bash -l

#SBATCH --nodes 1
#SBATCH --ntasks 1
#SBATCH --cpus-per-task 1
#SBATCH --mem 1024

srun shifter --image yourproject/alpine ls /etc

```

Interactive Shell (Bash)

To have an interactive shell within your image, simply use this:

```
$ srun --pty shifter --image yourproject/alpine bash
```

Using GPUs

On Deneb shifter runtime is installed on the GPU nodes. You need prior access to the GPUs nodes, see [FAQ](#)

```

[aubort@deneb1 ~]$ srun --gres gpu:1 --partition gpu --qos gpu shifter --image library/debian:stable-slim
nvidia-smi -L
GPU 0: Tesla K40m (UUID: GPU-21730043-7144-85e7-d251-7834adb2d1ee)

[aubort@deneb1 ~]$ srun --gres gpu:1 --partition gpu --qos gpu shifter --image library/nvidia-cuda:9.1-runtime
/home/aubort/gpu/cuda-samples/bin/x86_64/linux/release/simpleCUFFT
[simpleCUFFT] is starting...
GPU Device 0: "Tesla K40m" with compute capability 3.5

Temporary buffer size 448 bytes
Transforming signal cufftExecC2C
Launching ComplexPointwiseMulAndScale<<< >>>
Transforming signal back cufftExecC2C

```

FEEDBACK is welcome as this feature is experimental.

Account on the c4science registry

- [Request an account](#)
- Change your password on <https://registry.c4science.ch>

Related articles

- [FAQ](#)

- [Using the clusters](#)
- [Running Docker images using Shifter](#)
- [Licensed Software on the Clusters](#)
- [Running R on SCITAS machines](#)