Running Docker images using Shifter

- Running a Docker image with Shifter - Step by step
- Interactive Shell (Bash)
- Using GPUs
- Related articles

When you want to run Docker containers on the Fidis/Gacrux cluster

**Warning**
We are still in beta phase. The installation will soon be improved with:

- automatic account creation on the registry

The present documentation will be updated once the registry has been modified.

Running a Docker image with Shifter - Step by step

**Prerequisite**
You need to have Docker installed on your machine

1. **Get a docker image from dockerhub for instance**

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

2. **Account on the c4science registry**

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

3. **Set up your machine**

   - Login on the registry from your local Docker installation

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

4. **Upload a Docker image to the registry**

   - 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
   ```

5. **Pull an image on Shifter and specify a user or group ACL**

   - 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
5. 

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

- 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": "9797e5e798a034d53525968de25bd25c913e7bb17c6d068ebc778cb33e3f6e5",
            "itype": "docker",
            "last_pull": 1536842228.15727,
            "status": "READY",
            "status_message": "",
            "system": "fdata2-int.fidis",
            "tag": [
                "scitas/alpine:latest"
            ],
            "userACL": []
        },
        [...]
    ]
}

6. Run the image

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

```bash
#!/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:

```bash
$ 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.

Related articles

- Running Docker images using Shifter
- FAQ
- Licensed Software on the Clusters
- How to use Tensorflow on the GPU nodes
- Using the clusters