

Saving and Loading Images

Sunday, 4 August 2019 8:09 PM

The Docker CLI allows us to export and import Docker images and container layers using export/import or save/load Docker commands

The difference between save/load and export/import is that the first one works with images including metadata, but the export/import combination uses only container layers and doesn't include any image metadata information such as name, tags, and so on. In most cases, the save/load combination is more relevant and works properly for images without special needs

The `docker save` command packs the layers and metadata of all the chains required to build the image. You can then load this *saved* images chain into another Docker instance and create containers from these images.

```
njain50@ubuntu:~/dockerdemo$ docker pull redis
Using default tag: latest
latest: Pulling from library/redis
f5d23c7fed46: Pull complete
831c20fd50cb: Pull complete
bc2a0f25caa5: Pull complete
745ac314a007: Pull complete
6deeca231441: Pull complete
6291e84f5373: Pull complete
Digest: sha256:854715f5cd1b64d2f62ec219a7b7baceae149453e4d29a8f72cecb5ac51c4ad
Status: Downloaded newer image for redis:latest
njain50@ubuntu:~/dockerdemo$ docker save redis -o redis.tar
```

```

njain50@ubuntu: ~/dockerdemo
njain50@ubuntu:~/dockerdemo$ ls -lrt
total 99200
-rw-rw-r-- 1 njain50 njain50      141 Aug  3 08:25 Dockerfile
-rw----- 1 njain50 njain50 101574656 Aug  4 05:53 redis.tar
njain50@ubuntu:~/dockerdemo$
```

Now delete image in cache first:

```

njain50@ubuntu:~/dockerdemo$ docker rmi 857c4ab5f029
Untagged: redis:latest
Untagged: redis@sha256:854715f5cd1b64d2f62ec219a7b7baceae149453e4d29a8f72cecb5ac51c4ad
Deleted: sha256:857c4ab5f0291ecbb4de238be9d5f9676e63dcc9608f70c8acc3748fe9689911
Deleted: sha256:cf8131ebc8cf48e212a6cba652c19328eb997fa360e59dfc1d5ae4e9841e52d6
Deleted: sha256:ad2aeaa9a0026ba9194c4143de8846e93cea2a8851ac1c30b669c0c1040c4798
Deleted: sha256:e7a18a4c63c68b5c3848d87b970aea938032e78a14093b794e3bc8cfac4b3ab7
Deleted: sha256:2de5fabe69e135fd6c8e3ac5d5537d8943b9e964ec3b542eabc3b97ae810a4a2
Deleted: sha256:64c3e67d2d7fdeb252803ce9ed76375c756327bf88cc072b22c0fd1e24a9af2e
Deleted: sha256:d8a33133e477d367977987129313d9072e0ec80894ed4c52c2d88186f354c29a
njain50@ubuntu:~/dockerdemo$
```

Now when we load image from tar, we can again generate same image from it:

```

njain50@ubuntu: ~/dockerdemo
njain50@ubuntu:~/dockerdemo$ docker images
REPOSITORY          TAG             IMAGE ID        CREATED         SIZE
njain50@ubuntu:~/dockerdemo$ docker ps -a
CONTAINER ID       IMAGE          COMMAND         CREATED         STATUS          PORTS          NAMES
njain50@ubuntu:~/dockerdemo$ docker load -i redis.tar
d8a33133e477: Loading layer [=====>] 72.47MB/72.47MB
0a7b72b28690: Loading layer [=====>] 338.4kB/338.4kB
4b66eec92ef1: Loading layer [=====>] 4.079MB/4.079MB
f448714fe70e: Loading layer [=====>] 24.65MB/24.65MB
8ffc0dea4ded: Loading layer [=====>] 1.536kB/1.536kB
6bbfe33be022: Loading layer [=====>] 3.584kB/3.584kB
Loaded image: redis:latest
njain50@ubuntu:~/dockerdemo$ docker images
REPOSITORY          TAG             IMAGE ID        CREATED         SIZE
redis               latest          857c4ab5f029    2 days ago      98.2MB
njain50@ubuntu:~/dockerdemo$
```