

[\(/\)](#)[What is Docker? \(http://www.docker.com/whatisdocker/\)](http://www.docker.com/whatisdocker/)[Use Cases \(http://www.docker.com/resources/usecases/\)](http://www.docker.com/resources/usecases/)[Try It! \(http://www.docker.com/tryit/\)](http://www.docker.com/tryit/)[Browse \(https://registry.hub.docker.com\)](https://registry.hub.docker.com)[Log In \(https://hub.docker.com/account/login\)](https://hub.docker.com/account/login)[Sign Up \(https://hub.docker.com/account/signup\)](https://hub.docker.com/account/signup)[About \(/\)](#)[Installation \(/installation/ubuntu/linux/\)](/installation/ubuntu/linux/)[User Guide \(/userguide/\)](/userguide/)[Docker Hub \(/docker-hub/\)](/docker-hub/)

## Working with Docker Hub

[Examples \(Examples of Docker apps\)](#)[Articles \(/articles/basics/\)](/articles/basics/)[Reference \(/reference/commandline/cli/\)](/reference/commandline/cli/)

Version v1 (Latest)

So far you've learned how to use the command line to run Docker on your local host. You've learned how to pull down images (</userguide/usingdocker/>) to build containers from existing images and you've learned how to create your own images (</userguide/dockerimages/>).

Next, you're going to learn how to use the Docker Hub (<https://hub.docker.com>) to simplify and enhance your Docker workflows.

The Docker Hub (<https://hub.docker.com>) is a public registry maintained by Docker, Inc. It contains over 15,000 images you can download and use to build containers. It also provides authentication, work group structure, workflow tools like webhooks and build triggers, and privacy tools like private repositories for storing images you don't want to share publicly.

## Docker commands and Docker Hub

Docker itself provides access to Docker Hub services via the `docker search`, `pull`, `login`, and `push` commands. This page will show you how these commands work.

### Account creation and login

Typically, you'll want to start by creating an account on Docker Hub (if you haven't already) and logging in. You can create your account directly on Docker Hub (<https://hub.docker.com/account/signup/>), or by running:

```
$ sudo docker login
```

This will prompt you for a user name, which will become the public namespace for your public repositories. If your user name is available, Docker will prompt you to enter a password and your e-mail address. It will then automatically log you in. You can now commit and push your own images up to your repos on Docker Hub.

**Note:** Your authentication credentials will be stored in the `.dockercfg` authentication file in your home directory.

## Searching for images

You can search the Docker Hub (<https://hub.docker.com>) registry via its search interface or by using the command line interface. Searching can find images by image name, user name, or description:

```
$ sudo docker search centos
NAME                DESCRIPTION                               STARS   OFFICIAL   TRUSTED
centos              Official CentOS 6 Image as of 12 April 2014  88
tianon/centos       CentOS 5 and 6, created using rinse instea...  21
...
```

There you can see two example results: `centos` and `tianon/centos`. The second result shows that it comes from the public repository of a user, named `tianon/`, while the first result, `centos`, doesn't explicitly list a repository which means that it comes from the trusted top-level namespace. The `/` character separates a user's repository from the image name.

Once you've found the image you want, you can download it with `docker pull <imagename>`:

```
$ sudo docker pull centos
Pulling repository centos
0b443ba03958: Download complete
539c0211cd76: Download complete
511136ea3c5a: Download complete
7064731afe90: Download complete

Status: Downloaded newer image for centos
```

You now have an image from which you can run containers.

## Contributing to Docker Hub

Anyone can pull public images from the Docker Hub (<https://hub.docker.com>) registry, but if you would like to share your own images, then you must register first, as we saw in the first section of the Docker User Guide (</userguide/dockerhub/>).

## Pushing a repository to Docker Hub

In order to push a repository to its registry, you need to have named an image or committed your container to a named image as we saw here (/userguide/dockerimages).

Now you can push this repository to the registry designated by its name or tag.

```
$ sudo docker push yourname/newimage
```

The image will then be uploaded and available for use by your team-mates and/or the community.

## Features of Docker Hub

Let's take a closer look at some of the features of Docker Hub. You can find more information here (<http://docs.docker.com/docker-hub/>).

- Private repositories
- Organizations and teams
- Automated Builds
- Webhooks

### Private Repositories

Sometimes you have images you don't want to make public and share with everyone. So Docker Hub allows you to have private repositories. You can sign up for a plan here (<https://registry.hub.docker.com/plans/>).

### Organizations and teams

One of the useful aspects of private repositories is that you can share them only with members of your organization or team. Docker Hub lets you create organizations where you can collaborate with your colleagues and manage private repositories. You can learn how to create and manage an organization here (<https://registry.hub.docker.com/account/organizations/>).

### Automated Builds

Automated Builds automate the building and updating of images from GitHub (<https://www.github.com>) or BitBucket (<http://bitbucket.com>), directly on Docker Hub. It works by adding a commit hook to your selected GitHub or BitBucket repository, triggering a build and update when you push a commit.

#### To setup an Automated Build

1. Create a Docker Hub account (<https://hub.docker.com/>) and login.
2. Link your GitHub or BitBucket account through the "Link Accounts" (<https://registry.hub.docker.com/account/accounts/>) menu.
3. Configure an Automated Build (<https://registry.hub.docker.com/builds/add/>).
4. Pick a GitHub or BitBucket project that has a `Dockerfile` that you want to build.
5. Pick the branch you want to build (the default is the `master` branch).
6. Give the Automated Build a name.
7. Assign an optional Docker tag to the Build.
8. Specify where the `Dockerfile` is located. The default is `/`.

Once the Automated Build is configured it will automatically trigger a build and, in a few minutes, you should see your new Automated Build on the Docker Hub (<https://hub.docker.com>) Registry. It will stay in sync with your GitHub and BitBucket repository until you deactivate the Automated Build.

If you want to see the status of your Automated Builds, you can go to your Automated Builds page (<https://registry.hub.docker.com/builds/>) on the Docker Hub, and it will show you the status of your builds and their build history.

Once you've created an Automated Build you can deactivate or delete it. You cannot, however, push to an Automated Build with the `docker push` command. You can only manage it by committing code to your GitHub or BitBucket repository.

You can create multiple Automated Builds per repository and configure them to point to specific `Dockerfile`'s or Git branches.

### Build Triggers

Automated Builds can also be triggered via a URL on Docker Hub. This allows you to rebuild an Automated build image on demand.

### Webhooks

Webhooks are attached to your repositories and allow you to trigger an event when an image or updated image is pushed to the repository. With a webhook you can specify a target URL and a JSON payload that will be delivered when the image is pushed.

See the Docker Hub documentation for more information on webhooks (<http://docs.docker.com/docker-hub/repos/#webhooks>)

## Next steps

Go and use Docker!

## Community

Events

(<https://www.docker.com/community/events/>)

Friends' Posts

(<http://posts.docker.com/>)

Meetups

(<https://www.docker.com/community/meetups/>)

Governance

(<https://www.docker.com/community/governance/>)

Forums

(<http://forums.docker.com/>)

IRC

(<http://botbot.me/freenode/docker/>)

GitHub

(<https://github.com/docker/docker>)

Stackoverflow

(<http://stackoverflow.com/search?q=docker>)

Swag

(<http://www.cafepress.com/docker>)

## Enterprise

Support

(<https://www.docker.com/enterprise/support/>)

Education

(<https://www.docker.com/enterprise/education/>)

Services

(<https://www.docker.com/enterprise/services/>)

Partners

(<https://www.docker.com/enterprise/partners/>)

Find a Partner

(<https://www.docker.com/enterprise/partners/find/>)

Partner Program

(<https://www.docker.com/enterprise/partners/learn/>)

Learn More

(<https://www.docker.com/enterprise/partners/learn/>)

## Resources

Documentation

(<https://docs.docker.com/>)

Help

(<https://use.edocker.com/resources/help/>)

Use Cases

(<https://use.edocker.com/resources/usecases/>)

Online Tutorial

(<http://www.docker.com/tutorial>)

How To Buy

(<https://www.docker.com/enterprise/partners/find/>)

Status

(<http://status.docker.com/>)

Security

(<https://www.docker.com/resources/security/>)

## Company

About Us

(<https://www.docker.com/company/aboutus/>)

Team

(<https://www.docker.com/company/team/>)

News

(<https://www.docker.com/company/news/>)

Press

(<http://www.docker.com/company/press/>)

Careers

(<https://www.docker.com/company/careers/>)

Contact

(<https://www.docker.com/company/contact/>)

## Connect

Subscribe to our newsletter

(<https://www.docker.com/company/aboutus/>)

Enter your email



Blog

(<http://blog.docker.com/>)

Twitter

(<http://twitter.com/docker>)

Google+

(<https://plus.google.com/+docker>)

Facebook

(<https://www.facebook.com/docker>)

YouTube

(<http://www.youtube.com/docker>)

inc-1)

(<https://angel.co/docker>)

(<https://www.linkedin.com/company/docker>)

(<https://www.slideshare.net/Docker>)

(<https://www.github.com/docker>)

(<https://www.docker.com/081468566714>)

© 2014-2015 Docker, Inc.

Terms ([http://www.docker.com/legal/terms\\_of\\_service](http://www.docker.com/legal/terms_of_service)) · Privacy

([http://www.docker.com/legal/privacy\\_policy](http://www.docker.com/legal/privacy_policy)) · Trademarks

([http://www.docker.com/legal/trademark\\_guidelines](http://www.docker.com/legal/trademark_guidelines))