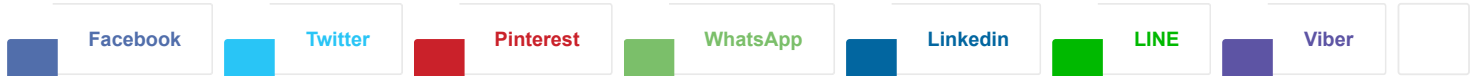


Setup Red Hat Quay Registry on CentOS / RHEL / Ubuntu

By [Josphat Mutai](#) - April 1, 2020 2721 0



You can support us by downloading this article as PDF from the Link below.

[Download the guide as PDF](#)

Red Hat Quay is an enterprise-quality container registry rebranded after the acquisition of CoreOS Quay Enterprise by Red Hat. Red Hat Quay registry is used to build and store containers, which are later deployed to the servers across your enterprise container platforms such as Kubernetes, OpenShift Container Platform e.t.c.

Check the Operator guide for installation on OpenShift:

[Install Project Quay Registry on OpenShift With Operator](#)

Features of Quay Registry

The top features of Quay include:

- High availability design
- Support for Geo-replication
- Support for Docker v2, schema 2 (multiarch)
- Best integration with Continuous integration pipelines
- Support for a custom log rotation
- Support for various Authentication, access methods, and storage backends
- Automated scanning for Security vulnerabilities

This guide will discuss the steps used to setup a single instance Quay Registry. This setup is for POC purposes and is not intended for use as a production install. For Highly available Quay registry, consult official Red Hat documentation.

Quay Components

The three core components of Simple Quay setup are:

- **Database:** Used by Red Hat Quay as its primary metadata storage (not for image storage).
- **Redis:** Used as a key/value store for providing real-time events.
- **Quay** (container registry): Runs the quay container as a service, consisting of several components in the pod.

Setup Hardware Minimum requirements

This setup requires a physical or virtual machine with the following minimum hardware requirements.

- Memory: **4 GB**
- CPUs: **2**
- Disk Space: **30GB**
- At least 10GB of disk space for docker storage (to run 3 containers)
- At least 10GB of disk space for Quay local storage (CEPH or other local storage might require more memory)

Below are the steps to follow during setup.

Step 1: Install Docker Engine

Docker is used to run Quay containers, install it on your system using our guides below.

[How To Install Docker on RHEL 7 Server / Workstation](#)

[How to install Docker CE on Ubuntu / Debian / Fedora / CentOS](#)

Step 2: Install and Deploy a Database

You can choose to run a database server in a container or on VM. For my setup, the MariaDB database is installed on the Virtual Machine that will host Quay.

```
--- CentOS / RHEL ---  
$ sudo yum -y install mariadb-server mariadb  
  
-- Ubuntu ----  
$ sudo apt install mariadb-server
```

Once installed, start and enable the server to start at boot.

```
sudo systemctl enable --now mariadb
```

Secure your Database server installation.

```
mysql_secure_installation
```

Now create a database for Quay.

```
CREATE DATABASE quay_registry;  
GRANT ALL ON quay_registry.* TO quay_registry@'%' IDENTIFIED BY "Ain  
FLUSH PRIVILEGES;  
QUIT;
```

Step 3: Install and Configure Redis

We'll also install Redis server on the Virtual machine and not in a container.

On CentOS / RHEL

```
sudo rpm -ivh https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm  
sudo rpm -ivh https://rpms.remirepo.net/enterprise/remi-release-7.rpm  
sudo yum install -y redis --enablerepo=remi
```

For RHEL 7, enable below repository:

```
sudo subscription-manager repos --enable=rhel-7-server-optional-rpm
```

Ubuntu:

```
sudo apt-get install redis-server
```

Start and enable redis service after installation.

```
sudo systemctl enable --now redis
```

Since everything is running in the same server, configure Redis to listen on 127.0.0.1 and Server Private IP address.

```
sudo vim /etc/redis.conf
```

Set the bind address to:

```
bind 127.0.0.1 ::1 10.10.1.15
```

Restart service after making the change.

```
sudo systemctl restart redis
```

Confirm that the redis service is started.

```
$ systemctl status redis  
$ ss -tunelp | grep 6379
```

Step 4: Generating Red Hat Quay Configuration

Before running the Red Hat Quay service as a container, you need to use that same quay container to create the configuration file (**config.yaml**) needed to deploy Red Hat Quay. The UI admin password is passed as a configuration argument to the Quay container.

```
$ sudo docker run -p 443:8443 quay.io/redhat/quay config Heer4giivoeCoh4
```

Replace **Heer4giivoeCoh4** with the admin password. Once the container is started, access Quay dashboard on **https://<IP|Hostname>:443**.

The login credentials are:

Username: quayconfig

Password: Heer4giivoeCoh4

Choose new Registry setup on the next screen.

Enter database connection settings for the schema to be created.

Create superuser account for administering Red Hat Quay.

Also provide redis connection information.

Set server hostname and whether to enable TLS/SSL.

Set Registry storage accordingly.

Validate Quay Registry settings and save.

Download configuration that will be used to bootstrap Quay registry servers.

Step 5: Deploy Red Hat Quay Registry

You have configuration file downloaded to your local machine, upload it to the server where Quay will run.

```
scp quay-config.tar.gz user@serverip:
```

You'll use the same quay container you used to create the configuration file to deploy the Red Hat Quay service on the nodes in your cluster. Since this is a basic setup, we're doing the setup on a single node. For high availability, you probably want three or more nodes (for example, quay01, quay02, and quay03)

Create data and configuration directories.

```
mkdir -p /data/quay/config  
mkdir -p /data/quay/storage
```


Copy config file and extract it.

```
cp quay-config.tar.gz /data/quay/config/  
cd /data/quay/config/  
tar xvf quay-config.tar.gz
```

Finally, start the Quay container.

```
docker run --restart=always -p 443:8443 -p 80:8080 \  
--sysctl net.core.somaxconn=4096 \  
-v /data/quay/config:/conf/stack:Z \  
-v /data/quay/storage:/datastorage:Z \  
-d quay.io/redhat/quay:v3.0.3
```

Allow http and https ports in the firewall.

```
sudo firewall-cmd --permanent --zone=trusted --add-port=80/tcp  
sudo firewall-cmd --permanent --zone=trusted --add-port=443/tcp  
sudo firewall-cmd --reload
```

You can now access Red Hat Quay Registry dashboard over **https** protocol on a configured DNS name.

For more reading on setup and more configurations, visit [Red Hat Quay documentation](#) page.