Registries

With Rancher, you can add credentials to access private registries from DockerHub, Quay.io, or any address that you have a private registry. By having the ability to access your private registries, it enables Rancher to use your private images. In each environment, you can only use one credential per registry address. This makes it a simple request to launch images from private addresses. If you have added multiple credentials for the same address, Rancher will always use the most recently added one.

Rancher supports different registries for each enviroment.

> **Note:** Currently, registries are only supported for **Cattle** and **Kubernetes** environments.

Adding Registries

On the **Infrastructure** -> **Registries** page, click on **Add Registry**.

For all registries, you'll need to provide the **e-mail address**, **username**, and **password**. For a **Custom** registry, you'll need to also provide the **registry address**. Click on **Create**.

> **Note:** For the `Address` in the custom registry, please do not pre-fix with `http://` or `https://` as we are expecting just the IP or Hostname.

If you add a credential for an address that already exists, Rancher will start using the new credentials.

Insecure Registries

In order to access an insecure registry, you'll need to configure your Docker daemon on your host(s). `DOMAIN` and `PORT` are the domain and port where the private registry is hosted.

> **Note:** Whenever you restart docker on the host, you may encounter issues with Network Agent being stuck in *Starting* state. To workaround the issue, please reboot the host.

```
# Edit the config file "/etc/default/docker"
$ sudo vi /etc/default/docker
# Add this line at the end of file. If there are already options, make
$ DOCKER_OPTS="$DOCKER_OPTS --insecure-registry=${DOMAIN}:${PORT}"
# Restart the docker service
$ sudo service docker restart
```

Self Signed Certificates

In order to use a self signed certificate with a registry, you'll need to configure your Docker daemon on your host(s). `DOMAIN` and `PORT` are the domain and port where the private registry is hosted.

```
# Download the certificate from the domain
$ openssl s_client -showcerts -connect ${DOMAIN}:${PORT} </dev/null 2>
# Copy the certificate to the appropriate directories
$ sudo cp ca.crt /etc/docker/certs.d/${DOMAIN}/ca.crt
# Append the certificate to a file
$ cat ca.crt | sudo tee -a /etc/ssl/certs/ca-certificates.crt
# Restart the docker service to have the changes take affect
$ sudo service docker restart
```

Using Registries

As soon as the registry is created, you will be able to use these private registries when launching services and containers. The syntax for the image name is the same as what you would use for the `docker run` command.

```
[registry-name]/[namespace]/[imagename]:[version]
```

By default, we are assuming that you are trying to pull images from `DockerHub`.

Editing Registries

All options for a registry are accessible through the dropdown menu on the right hand side of the listed registry.

For any **Active** registry, you can **Deactivate** the registry, which would prohibit access to the registry. No new containers can be launched with any images in that registry.

For any **Deactivated** registry, you have two options. You can **Activate** the registry, which will allow containers to access images from those registries. Any members of your environment will be able to activate your credential without needing to re-input the password. If you don't want anyone using your credential, you should **Delete** the registry, which will remove the credentials from the environment.

You can **Edit** any registry, which allows you to change the credentials to the registry address. You will not be able to change the registry address. The password is not saved in the "Edit" page, so you will need to re-input it in order to save any changes.

> **Note:** If a registry is invalid (i.e. inactive, removed, or overridden due to a newer credential), any service using a private registry image will continue to run. Since the image has already been pulled onto the host, there will be no restrictions on usage of the image regardless of registry permissions. Therefore, any scaling up of services or additional containers using the image will be able to run. Rancher does not check if the credentials are still valid when running containers as we assume that you've already given the host permissions to access the image.

Copyright © 2017 Rancher Labs. All Rights Reserved.

Edit this page